ckb-next

v0.2.8 at branch master

Generated by Doxygen 1.8.6

Sat Jan 20 2018 16:59:32

Contents

1 README 1

2 Building ckb 9

3 cbk Improvements Roadmap 11

4 DAEMON 13

5 Todo List 21

6 Deprecated List 23

7 Namespace Index 25

7.1 Namespace List . . . 25

8 Hierarchical Index 27

8.1 Class Hierarchy . . . 27

9 Data Structure Index 31

9.1 Data Structures . . . 31

10 File Index 35

10.1 File List . . . 35

11 Namespace Documentation 39

11.1 Ui Namespace Reference . . . 39

12 Data Structure Documentation 41

12.1 AnimAddDialog Class Reference . . . 41

12.1.1 Detailed Description . . . 42

12.1.2 Constructor & Destructor Documentation . . . 42

12.1.2.1 AnimAddDialog . . . 43

12.1.2.2 ~AnimAddDialog . . . 43

12.1.3 Member Function Documentation . . . 43

12.1.3.1 chosenPreset . . . 43

12.1.3.2 chosenScript . . . 44

12.1.3.3 on_animBox_activated . . . 44
12.1.3.4 on_presetBox_activated . . . 45
12.1.3.5 on_previewBox_clicked . . . 46
12.1.4 Field Documentation . . . 46
12.1.4.1 _keys . . . 46
12.1.4.2 _light . . . 46
12.1.4.3 scripts . . . 46
12.1.4.4 showPreview . . . 47
12.1.4.5 ui . . . 47
12.2 Ui::AnimAddDialog Class Reference . . . 47
12.2.1 Detailed Description . . . 48
12.2.2 Member Function Documentation . . . 48
12.2.2.1 retranslateUi . . . 48
12.2.2.2 setupUi . . . 49
12.2.3 Field Documentation . . . 50
12.2.3.1 animBox . . . 50
12.2.3.2 buttonBox . . . 51
12.2.3.3 gridLayout . . . 51
12.2.3.4 horizontalLayout . . . 51
12.2.3.5 horizontalSpacer . . . 51
12.2.3.6 horizontalSpacer_2 . . . 51
12.2.3.7 label . . . 51
12.2.3.8 label_2 . . . 51
12.2.3.9 presetBox . . . 51
12.2.3.10 previewBox . . . 51
12.2.3.11 verticalSpacer_2 . . . 52
12.2.3.12 verticalSpacer_3 . . . 52
12.3 AnimScript Class Reference . . . 52
12.3.1 Detailed Description . . . 56
12.3.2 Data Structure Documentation . . . 56
12.3.2.1 struct AnimScript::Param . . . 56
12.3.3 Member Typedef Documentation . . . 57
12.3.3.1 PresetValue . . . 57
12.3.4 Constructor & Destructor Documentation . . . 57
12.3.4.1 ~AnimScript . . . 57
12.3.4.2 AnimScript . . . 57
12.3.4.3 AnimScript . . . 58
12.3.5 Member Function Documentation . . . 58
12.3.5.1 advance . . . 58
12.3.5.2 author . . . 58

12.3.5.3 begin . . . 59
12.3.5.4 colors . . . 60
12.3.5.5 copy . . . 60
12.3.5.6 copyright . . . 61
12.3.5.7 count . . . 61
12.3.5.8 description . . . 62
12.3.5.9 end . . . 62
12.3.5.10 frame . . . 63
12.3.5.11 guid . . . 64
12.3.5.12 hasFrame . . . 64
12.3.5.13 hasKeypress . . . 65
12.3.5.14 hasParam . . . 65
12.3.5.15 init . . . 66
12.3.5.16 keypress . . . 67
12.3.5.17 license . . . 68
12.3.5.18 list . . . 68
12.3.5.19 load . . . 69
12.3.5.20 name . . . 72
12.3.5.21 param . . . 72
12.3.5.22 parameters . . . 73
12.3.5.23 paramIterator . . . 74
12.3.5.24 path . . . 74
12.3.5.25 preset . . . 75
12.3.5.26 presets . . . 75
12.3.5.27 printParams . . . 75
12.3.5.28 readProcess . . . 76
12.3.5.29 retrigger . . . 77
12.3.5.30 scan . . . 78
12.3.5.31 setDuration . . . 79
12.3.5.32 stop . . . 79
12.3.5.33 version . . . 80
12.3.5.34 year . . . 80
12.3.6 Field Documentation . . . 81
12.3.6.1 _colorBuffer . . . 81
12.3.6.2 _colors . . . 81
12.3.6.3 _info . . . 81
12.3.6.4 _keys . . . 81
12.3.6.5 _map . . . 81
12.3.6.6 _paramValues . . . 81
12.3.6.7 _path . . . 81

12.3.6.8 _presets . . . 82
12.3.6.9 _presetValues . . . 82
12.3.6.10 durationMsec . . . 82
12.3.6.11 firstFrame . . . 82
12.3.6.12 inFrame . . . 82
12.3.6.13 initialized . . . 82
12.3.6.14 KP_NAME . . . 82
12.3.6.15 KP_NONE . . . 82
12.3.6.16 KP_POSITION . . . 82
12.3.6.17 lastFrame . . . 83
12.3.6.18 minX . . . 83
12.3.6.19 minY . . . 83
12.3.6.20 process . . . 83
12.3.6.21 readAnyFrame . . . 83
12.3.6.22 readFrame . . . 83
12.3.6.23 repeatMsec . . . 83
12.3.6.24 scripts . . . 83
12.3.6.25 stopped . . . 83
12.4 Ui::AnimSettingDialog Class Reference . . . 84
12.4.1 Detailed Description . . . 86
12.4.2 Member Function Documentation . . . 86
12.4.2.1 retranslateUi . . . 86
12.4.2.2 setupUi . . . 87
12.4.3 Field Documentation . . . 91
12.4.3.1 animName . . . 91
12.4.3.2 buttonBox . . . 91
12.4.3.3 delayBox . . . 91
12.4.3.4 gridLayout . . . 92
12.4.3.5 gridLayout_2 . . . 92
12.4.3.6 horizontalSpacer . . . 92
12.4.3.7 horizontalSpacer_2 . . . 92
12.4.3.8 horizontalSpacer_3 . . . 92
12.4.3.9 horizontalSpacer_4 . . . 92
12.4.3.10 horizontalSpacer_5 . . . 92
12.4.3.11 horizontalSpacer_6 . . . 92
12.4.3.12 infoTab . . . 92
12.4.3.13 kpDelayBox . . . 93
12.4.3.14 kpModeStopBox . . . 93
12.4.3.15 kpReleaseBox . . . 93
12.4.3.16 kpRepeatBox . . . 93

12.4.3.17 kpRepeatLabel . . . 93
12.4.3.18 kpRepeatLabel_2 . . . 93
12.4.3.19 label . . . 93
12.4.3.20 label_10 . . . 93
12.4.3.21 label_11 . . . 93
12.4.3.22 label_12 . . . 94
12.4.3.23 label_13 . . . 94
12.4.3.24 label_2 . . . 94
12.4.3.25 label_3 . . . 94
12.4.3.26 label_4 . . . 94
12.4.3.27 label_5 . . . 94
12.4.3.28 label_6 . . . 94
12.4.3.29 label_7 . . . 94
12.4.3.30 label_8 . . . 94
12.4.3.31 label_9 . . . 95
12.4.3.32 line . . . 95
12.4.3.33 line_2 . . . 95
12.4.3.34 repeatBox . . . 95
12.4.3.35 repeatLabel . . . 95
12.4.3.36 repeatLabel_2 . . . 95
12.4.3.37 sAuthorLabel . . . 95
12.4.3.38 sDescLabel . . . 95
12.4.3.39 settingsGrid . . . 95
12.4.3.40 settingTab . . . 96
12.4.3.41 sLicenseLabel . . . 96
12.4.3.42 sNameLabel . . . 96
12.4.3.43 sVerLabel . . . 96
12.4.3.44 sYearLabel . . . 96
12.4.3.45 tabWidget . . . 96
12.4.3.46 timeGrid . . . 96
12.4.3.47 timeTab . . . 96
12.4.3.48 verticalSpacer . . . 96
12.4.3.49 verticalSpacer_2 . . . 97
12.4.3.50 verticalSpacer_3 . . . 97
12.5 AnimSettingDialog Class Reference . . . 97
12.5.1 Detailed Description . . . 100
12.5.2 Constructor & Destructor Documentation . . . 100
12.5.2.1 AnimSettingDialog . . . 100
12.5.2.2 ~AnimSettingDialog . . . 105
12.5.3 Member Function Documentation . . . 105

12.5.3.1 angleDialChanged . . . 106
12.5.3.2 angleSpinnerChanged . . . 106
12.5.3.3 hLine . . . 107
12.5.3.4 name . . . 108
12.5.3.5 newDuration . . . 108
12.5.3.6 on_delayBox_valueChanged . . . 109
12.5.3.7 on_kpDelayBox_valueChanged . . . 110
12.5.3.8 on_kpReleaseBox_stateChanged . . . 110
12.5.3.9 on_kpRepeatBox_valueChanged . . . 111
12.5.3.10 on_repeatBox_valueChanged . . . 112
12.5.3.11 updateParam . . . 112
12.5.3.12 updateStops . . . 115
12.5.4 Field Documentation . . . 116
12.5.4.1 _anim . . . 116
12.5.4.2 angleDialMapper . . . 116
12.5.4.3 angleSpinnerMapper . . . 116
12.5.4.4 angleSpinners . . . 117
12.5.4.5 hasRepeat . . . 117
12.5.4.6 kpStopCheck . . . 117
12.5.4.7 lastDuration . . . 117
12.5.4.8 settingWidgets . . . 117
12.5.4.9 stopCheck . . . 117
12.5.4.10 ui . . . 117
12.5.4.11 updateMapper . . . 117
12.6 AutoRun Class Reference . . . 118
12.6.1 Detailed Description . . . 118
12.6.2 Member Function Documentation . . . 118
12.6.2.1 available . . . 118
12.6.2.2 disable . . . 119
12.6.2.3 enable . . . 119
12.6.2.4 isEnabled . . . 120
12.6.2.5 once . . . 121
12.7 CkbSettings Class Reference . . . 122
12.7.1 Detailed Description . . . 124
12.7.2 Constructor & Destructor Documentation . . . 124
12.7.2.1 CkbSettings . . . 124
12.7.2.2 CkbSettings . . . 124
12.7.2.3 CkbSettings . . . 125
12.7.2.4 ~CkbSettings . . . 125
12.7.3 Member Function Documentation . . . 125

12.7.3.1 beginGroup . . . 125
12.7.3.2 childGroups . . . 125
12.7.3.3 childKeys . . . 126
12.7.3.4 cleanUp . . . 127
12.7.3.5 contains . . . 128
12.7.3.6 containsGroup . . . 128
12.7.3.7 endGroup . . . 129
12.7.3.8 get . . . 129
12.7.3.9 isBusy . . . 130
12.7.3.10 pwd . . . 131
12.7.3.11 pwd . . . 132
12.7.3.12 remove . . . 133
12.7.3.13 set . . . 133
12.7.3.14 setValue . . . 135
12.7.3.15 value . . . 135
12.7.4 Field Documentation . . . 136
12.7.4.1 backing . . . 136
12.7.4.2 groups . . . 136
12.7.4.3 removeCache . . . 137
12.7.4.4 writeCache . . . 137
12.8 CkbSettingsWriter Class Reference . . . 137
12.8.1 Detailed Description . . . 138
12.8.2 Constructor & Destructor Documentation . . . 138
12.8.2.1 CkbSettingsWriter . . . 138
12.8.2.2 ∼CkbSettingsWriter . . . 139
12.8.3 Member Function Documentation . . . 139
12.8.3.1 run . . . 139
12.8.4 Field Documentation . . . 139
12.8.4.1 _backing . . . 139
12.8.4.2 _removals . . . 139
12.8.4.3 _updates . . . 139
12.9 ColorButton Class Reference . . . 139
12.9.1 Detailed Description . . . 142
12.9.2 Constructor & Destructor Documentation . . . 142
12.9.2.1 ColorButton . . . 142
12.9.3 Member Function Documentation . . . 142
12.9.3.1 allowAlpha . . . 142
12.9.3.2 allowAlpha . . . 143
12.9.3.3 bigIcons . . . 143
12.9.3.4 bigIcons . . . 144

12.9.3.5 color . . . 144
12.9.3.6 color . . . 145
12.9.3.7 colorChanged . . . 146
12.9.3.8 pickColor . . . 146
12.9.3.9 setLabel . . . 147
12.9.3.10 setLabel . . . 147
12.9.3.11 updateImage . . . 147
12.9.4 Field Documentation . . . 148
12.9.4.1 _alpha . . . 148
12.9.4.2 _bigIcons . . . 148
12.9.4.3 _color . . . 148
12.9.4.4 _setLabel . . . 148
12.10 ColorMap Class Reference . . . 149
12.10.1 Detailed Description . . . 150
12.10.2 Constructor & Destructor Documentation . . . 150
12.10.2.1 ColorMap . . . 150
12.10.2.2 ~ColorMap . . . 150
12.10.2.3 ColorMap . . . 150
12.10.3 Member Function Documentation . . . 151
12.10.3.1 alloc . . . 151
12.10.3.2 clear . . . 151
12.10.3.3 colorForName . . . 152
12.10.3.4 colorForName . . . 153
12.10.3.5 colors . . . 153
12.10.3.6 colors . . . 154
12.10.3.7 count . . . 154
12.10.3.8 deAlloc . . . 154
12.10.3.9 init . . . 155
12.10.3.10 keyNames . . . 156
12.10.3.11 operator= . . . 156
12.10.3.12 operator== . . . 157
12.10.4 Field Documentation . . . 157
12.10.4.1 _colors . . . 157
12.10.4.2 _count . . . 157
12.10.4.3 _keyNames . . . 157
12.10.4.4 _mapCount . . . 157
12.11 Ui::ExtraSettingsWidget Class Reference . . . 158
12.11.1 Detailed Description . . . 160
12.11.2 Member Function Documentation . . . 160
12.11.2.1 retranslateUi . . . 160

12.11.2.2 setupUi . . . 161
12.11.3 Field Documentation . . . 165
12.11.3.1 animCountLabel . . . 165
12.11.3.2 animPathLabel . . . 165
12.11.3.3 animScanButton . . . 165
12.11.3.4 brightnessBox . . . 165
12.11.3.5 buttonBox . . . 166
12.11.3.6 delayBox . . . 166
12.11.3.7 ditherBox . . . 166
12.11.3.8 fpsBox . . . 166
12.11.3.9 fpsWarnLabel . . . 166
12.11.3.10 gridLayout . . . 166
12.11.3.11 horizontalLayout . . . 166
12.11.3.12 horizontalLayout_2 . . . 166
12.11.3.13 horizontalSpacer . . . 166
12.11.3.14 horizontalSpacer_2 . . . 167
12.11.3.15 horizontalSpacer_3 . . . 167
12.11.3.16 label . . . 167
12.11.3.17 label_10 . . . 167
12.11.3.18 label_11 . . . 167
12.11.3.19 label_2 . . . 167
12.11.3.20 label_3 . . . 167
12.11.3.21 label_5 . . . 167
12.11.3.22 label_6 . . . 167
12.11.3.23 line . . . 168
12.11.3.24 line_2 . . . 168
12.11.3.25 line_3 . . . 168
12.11.3.26 mAccelBox . . . 168
12.11.3.27 osxLabel . . . 168
12.11.3.28 osxLine . . . 168
12.11.3.29 sAccelBox . . . 168
12.11.3.30 sSpeedBox . . . 168
12.11.3.31 sSpeedWidget . . . 168
12.11.3.32 trayBox . . . 169
12.11.3.33 verticalSpacer . . . 169
12.11.3.34 verticalSpacer_10 . . . 169
12.11.3.35 verticalSpacer_11 . . . 169
12.11.3.36 verticalSpacer_2 . . . 169
12.11.3.37 verticalSpacer_3 . . . 169
12.11.3.38 verticalSpacer_4 . . . 169

12.11.3.39 verticalSpacer_5 . . . 169
12.11.3.40 verticalSpacer_6 . . . 169
12.11.3.41 verticalSpacer_7 . . . 170
12.11.3.42 verticalSpacer_8 . . . 170
12.11.3.43 verticalSpacer_9 . . . 170
12.12 ExtraSettingsWidget Class Reference . . . 170
12.12.1 Detailed Description . . . 172
12.12.2 Constructor & Destructor Documentation . . . 172
12.12.2.1 ExtraSettingsWidget . . . 172
12.12.2.2 ~ExtraSettingsWidget . . . 173
12.12.3 Member Function Documentation . . . 173
12.12.3.1 on_animScanButton_clicked . . . 173
12.12.3.2 on_brightnessBox_clicked . . . 174
12.12.3.3 on_delayBox_clicked . . . 175
12.12.3.4 on_ditherBox_clicked . . . 175
12.12.3.5 on_fpsBox_valueChanged . . . 175
12.12.3.6 on_mAccelBox_clicked . . . 176
12.12.3.7 on_sAccelBox_clicked . . . 176
12.12.3.8 on_sSpeedBox_valueChanged . . . 177
12.12.3.9 on_trayBox_clicked . . . 177
12.12.3.10 pollUpdates . . . 177
12.12.4 Field Documentation . . . 178
12.12.4.1 ui . . . 178
12.13 KbFirmware::FW Struct Reference . . . 179
12.13.1 Detailed Description . . . 179
12.13.2 Constructor & Destructor Documentation . . . 179
12.13.2.1 FW . . . 179
12.13.3 Field Documentation . . . 180
12.13.3.1 ckbVersion . . . 180
12.13.3.2 fileName . . . 180
12.13.3.3 fwVersion . . . 180
12.13.3.4 hash . . . 180
12.13.3.5 url . . . 180
12.14 FwUpgradeDialog Class Reference . . . 180
12.14.1 Detailed Description . . . 182
12.14.2 Constructor & Destructor Documentation . . . 182
12.14.2.1 FwUpgradeDialog . . . 182
12.14.2.2 ~FwUpgradeDialog . . . 183
12.14.3 Member Function Documentation . . . 183
12.14.3.1 cleanBlob . . . 183

12.14.3.2 closeEvent . . . 184
12.14.3.3 exec . . . 184
12.14.3.4 fwUpdateFinished . . . 185
12.14.3.5 fwUpdateProgress . . . 186
12.14.3.6 on_actionButton_clicked . . . 186
12.14.3.7 on_cancelButton_clicked . . . 187
12.14.3.8 removeDev . . . 187
12.14.3.9 saveBlob . . . 188
12.14.4 Field Documentation . . . 188
12.14.4.1 blob . . . 188
12.14.4.2 evLoop . . . 188
12.14.4.3 exitSuccess . . . 188
12.14.4.4 kb . . . 188
12.14.4.5 savePath . . . 189
12.14.4.6 ui . . . 189
12.15 Ui::FwUpgradeDialog Class Reference . . . 189
12.15.1 Detailed Description . . . 190
12.15.2 Member Function Documentation . . . 191
12.15.2.1 retranslateUi . . . 191
12.15.2.2 setupUi . . . 191
12.15.3 Field Documentation . . . 193
12.15.3.1 actionButton . . . 193
12.15.3.2 cancelButton . . . 193
12.15.3.3 curLabel . . . 193
12.15.3.4 devLabel . . . 194
12.15.3.5 gridLayout . . . 194
12.15.3.6 gridLayout_2 . . . 194
12.15.3.7 horizontalSpacer . . . 194
12.15.3.8 horizontalSpacer_2 . . . 194
12.15.3.9 horizontalSpacer_3 . . . 194
12.15.3.10 horizontalSpacer_4 . . . 194
12.15.3.11 label_2 . . . 194
12.15.3.12 label_4 . . . 194
12.15.3.13 newLabel . . . 195
12.15.3.14 progressBar . . . 195
12.15.3.15 statusLabel . . . 195
12.16 GradientButton Class Reference . . . 195
12.16.1 Detailed Description . . . 196
12.16.2 Constructor & Destructor Documentation . . . 197
12.16.2.1 GradientButton . . . 197

12.16.3 Member Function Documentation . . . 197
12.16.3.1 fromString . . . 197
12.16.3.2 gradientChanged . . . 198
12.16.3.3 pickGradient . . . 198
12.16.3.4 toString . . . 199
12.16.3.5 updateImage . . . 200
12.16.4 Field Documentation . . . 201
12.16.4.1 _alpha . . . 201
12.16.4.2 _stops . . . 201
12.17 Ui::GradientDialog Class Reference . . . 201
12.17.1 Detailed Description . . . 203
12.17.2 Member Function Documentation . . . 203
12.17.2.1 retranslateUi . . . 203
12.17.2.2 setupUi . . . 204
12.17.3 Field Documentation . . . 206
12.17.3.1 buttonBox . . . 206
12.17.3.2 gridLayout . . . 206
12.17.3.3 gridLayout_2 . . . 207
12.17.3.4 gridLayout_3 . . . 207
12.17.3.5 groupBox_2 . . . 207
12.17.3.6 horizontalLayout . . . 207
12.17.3.7 horizontalLayout_2 . . . 207
12.17.3.8 horizontalSpacer . . . 207
12.17.3.9 horizontalSpacer_2 . . . 207
12.17.3.10 horizontalSpacer_3 . . . 207
12.17.3.11 label . . . 207
12.17.3.12 label_2 . . . 208
12.17.3.13 presetDelete . . . 208
12.17.3.14 presetList . . . 208
12.17.3.15 presetName . . . 208
12.17.3.16 presetSave . . . 208
12.17.3.17 stopBox . . . 208
12.17.3.18 stopColor . . . 208
12.17.3.19 stopOpacity . . . 208
12.17.3.20 stopPos . . . 209
12.17.3.21 verticalSpacer . . . 209
12.17.3.22 verticalSpacer_2 . . . 209
12.17.3.23 widget . . . 209
12.18 GradientDialog Class Reference . . . 209
12.18.1 Detailed Description . . . 212

12.18.2 Constructor & Destructor Documentation . . . 212
12.18.2.1 GradientDialog . . . 212
12.18.2.2 ~GradientDialog . . . 213
12.18.3 Member Function Documentation . . . 213
12.18.3.1 addPreset . . . 213
12.18.3.2 colorChanged . . . 214
12.18.3.3 currentChanged . . . 214
12.18.3.4 getGradient . . . 215
12.18.3.5 makeIcon . . . 216
12.18.3.6 on_presetDelete_clicked . . . 217
12.18.3.7 on_presetList_currentItemChanged . . . 217
12.18.3.8 on_presetName_textEdited . . . 218
12.18.3.9 on_presetSave_clicked . . . 218
12.18.3.10 on_stopOpacity_valueChanged . . . 219
12.18.3.11 on_stopPos_valueChanged . . . 219
12.18.3.12 setPreset . . . 220
12.18.3.13 updatePresets . . . 221
12.18.4 Field Documentation . . . 221
12.18.4.1 currentPreset . . . 221
12.18.4.2 presets . . . 222
12.18.4.3 ui . . . 222
12.19 GradientDialogWidget Class Reference . . . 222
12.19.1 Detailed Description . . . 225
12.19.2 Constructor & Destructor Documentation . . . 225
12.19.2.1 GradientDialogWidget . . . 225
12.19.3 Member Function Documentation . . . 225
12.19.3.1 colorAt . . . 225
12.19.3.2 current . . . 226
12.19.3.3 currentChanged . . . 226
12.19.3.4 fillRect . . . 226
12.19.3.5 makeStops . . . 227
12.19.3.6 mouseMoveEvent . . . 228
12.19.3.7 mousePressEvent . . . 229
12.19.3.8 mouseReleaseEvent . . . 230
12.19.3.9 moveCurrent . . . 230
12.19.3.10 paintEvent . . . 231
12.19.3.11 selectedStop . . . 232
12.19.3.12 selectionColors . . . 233
12.19.3.13 setCurrentColor . . . 234
12.19.3.14 setStops . . . 234

12.19.3.15 stopCount . . . 235
12.19.3.16 stopRect . . . 236
12.19.3.17 stops . . . 236
12.19.4 Field Documentation . . . 237
12.19.4.1 _colors . . . 237
12.19.4.2 _current . . . 237
12.19.4.3 _stops . . . 237
12.19.4.4 selected . . . 237
12.19.4.5 selectedOffset . . . 237
12.19.4.6 selectedPos . . . 237
12.20 JlCompress Class Reference . . . 238
12.20.1 Detailed Description . . . 239
12.20.2 Member Function Documentation . . . 239
12.20.2.1 compressDir . . . 239
12.20.2.2 compressFile . . . 240
12.20.2.3 compressFile . . . 241
12.20.2.4 compressFiles . . . 243
12.20.2.5 compressSubDir . . . 245
12.20.2.6 extractDir . . . 247
12.20.2.7 extractFile . . . 249
12.20.2.8 extractFile . . . 251
12.20.2.9 extractFiles . . . 252
12.20.2.10 getFileList . . . 253
12.20.2.11 removeFile . . . 255
12.21 Kb Class Reference . . . 256
12.21.1 Detailed Description . . . 261
12.21.2 Constructor & Destructor Documentation . . . 261
12.21.2.1 ~Kb . . . 261
12.21.2.2 Kb . . . 262
12.21.3 Member Function Documentation . . . 264
12.21.3.1 appendProfile . . . 264
12.21.3.2 autoSave . . . 264
12.21.3.3 currentBind . . . 265
12.21.3.4 currentLight . . . 265
12.21.3.5 currentMode . . . 266
12.21.3.6 currentPerf . . . 266
12.21.3.7 currentProfile . . . 267
12.21.3.8 deleteHw . . . 267
12.21.3.9 deletePrevious . . . 268
12.21.3.10 dither . . . 268

12.21.3.11 dither . . . 269
12.21.3.12 find . . . 269
12.21.3.13 frameRate . . . 270
12.21.3.14 frameRate . . . 270
12.21.3.15 frameUpdate . . . 271
12.21.3.16 fwUpdate . . . 272
12.21.3.17 fwUpdateFinished . . . 273
12.21.3.18 fwUpdateProgress . . . 273
12.21.3.19 getKeyMap . . . 274
12.21.3.20 getMacroNumber . . . 274
12.21.3.21 getMacroPath . . . 275
12.21.3.22 hwProfile . . . 275
12.21.3.23 hwProfile . . . 275
12.21.3.24 hwSave . . . 276
12.21.3.25 indexOf . . . 277
12.21.3.26 infoUpdated . . . 278
12.21.3.27 isKeyboard . . . 278
12.21.3.28 isMouse . . . 279
12.21.3.29 isOpen . . . 280
12.21.3.30 layout . . . 280
12.21.3.31 layout . . . 281
12.21.3.32 load . . . 281
12.21.3.33 macroDelay . . . 282
12.21.3.34 macroDelay . . . 282
12.21.3.35 matches . . . 283
12.21.3.36 modeChanged . . . 283
12.21.3.37 model . . . 284
12.21.3.38 modeRenamed . . . 284
12.21.3.39 mouseAccel . . . 284
12.21.3.40 mouseAccel . . . 285
12.21.3.41 needsSave . . . 285
12.21.3.42 newMode . . . 285
12.21.3.43 newMode . . . 286
12.21.3.44 newProfile . . . 287
12.21.3.45 newProfile . . . 287
12.21.3.46 profileAdded . . . 288
12.21.3.47 profileChanged . . . 288
12.21.3.48 profileRenamed . . . 289
12.21.3.49 profiles . . . 289
12.21.3.50 profiles . . . 290

12.21.3.51 readNotify . . . 290
12.21.3.52 run . . . 294
12.21.3.53 save . . . 295
12.21.3.54 scrollSpeed . . . 296
12.21.3.55 scrollSpeed . . . 296
12.21.3.56 setCurrentMode . . . 297
12.21.3.57 setCurrentMode . . . 297
12.21.3.58 setCurrentMode . . . 298
12.21.3.59 setCurrentProfile . . . 299
12.21.3.60 updateLayout . . . 300
12.21.3.61 writeProfileHeader . . . 301
12.21.4 Friends And Related Function Documentation . . . 302
12.21.4.1 KbManager . . . 302
12.21.5 Field Documentation . . . 302
12.21.5.1 _currentMode . . . 302
12.21.5.2 _currentProfile . . . 302
12.21.5.3 _delay . . . 302
12.21.5.4 _dither . . . 302
12.21.5.5 _frameRate . . . 302
12.21.5.6 _hwProfile . . . 303
12.21.5.7 _layout . . . 303
12.21.5.8 _model . . . 303
12.21.5.9 _mouseAccel . . . 303
12.21.5.10 _needsSave . . . 303
12.21.5.11 _profiles . . . 303
12.21.5.12 _scrollSpeed . . . 303
12.21.5.13 cmd . . . 303
12.21.5.14 cmdpath . . . 303
12.21.5.15 devpath . . . 304
12.21.5.16 features . . . 304
12.21.5.17 firmware . . . 304
12.21.5.18 fwUpdPath . . . 304
12.21.5.19 hwLoading . . . 304
12.21.5.20 HWMODE_MAX . . . 304
12.21.5.21 hwModeCount . . . 304
12.21.5.22 iState . . . 304
12.21.5.23 lastAutoSave . . . 304
12.21.5.24 macroNumber . . . 305
12.21.5.25 macroPath . . . 305
12.21.5.26 monochrome . . . 305

12.21.5.27notifyNumber . . . 305
12.21.5.28notifyPath . . . 305
12.21.5.29pollrate . . . 305
12.21.5.30prefsPath . . . 305
12.21.5.31prevMode . . . 305
12.21.5.32prevProfile . . . 305
12.21.5.33usbModel . . . 306
12.21.5.34usbSerial . . . 306
12.22KbAnim Class Reference . . . 306
12.22.1 Detailed Description . . . 309
12.22.2 Member Enumeration Documentation . . . 309
12.22.2.1 Mode . . . 309
12.22.3 Constructor & Destructor Documentation . . . 310
12.22.3.1 KbAnim . . . 310
12.22.3.2 KbAnim . . . 311
12.22.3.3 KbAnim . . . 312
12.22.4 Member Function Documentation . . . 312
12.22.4.1 blend . . . 312
12.22.4.2 catchUp . . . 314
12.22.4.3 commitParams . . . 315
12.22.4.4 effectiveParams . . . 316
12.22.4.5 guid . . . 316
12.22.4.6 hasParameter . . . 317
12.22.4.7 isActive . . . 317
12.22.4.8 isRunning . . . 317
12.22.4.9 keypress . . . 318
12.22.4.10keys . . . 319
12.22.4.11keys . . . 320
12.22.4.12map . . . 320
12.22.4.13map . . . 321
12.22.4.14mode . . . 321
12.22.4.15mode . . . 321
12.22.4.16name . . . 322
12.22.4.17name . . . 322
12.22.4.18needsSave . . . 322
12.22.4.19newId . . . 323
12.22.4.20opacity . . . 323
12.22.4.21opacity . . . 323
12.22.4.22parameter . . . 324
12.22.4.23parameter . . . 324

12.22.4.24 reInit . . . 325
12.22.4.25 resetParams . . . 325
12.22.4.26 save . . . 326
12.22.4.27 script . . . 327
12.22.4.28 scriptName . . . 327
12.22.4.29 stop . . . 328
12.22.4.30 trigger . . . 329
12.22.4.31 updateParams . . . 330
12.22.5 Field Documentation . . . 331
12.22.5.1 _guid . . . 331
12.22.5.2 _isActive . . . 331
12.22.5.3 _isActiveKp . . . 331
12.22.5.4 _keys . . . 331
12.22.5.5 _map . . . 331
12.22.5.6 _mode . . . 331
12.22.5.7 _name . . . 331
12.22.5.8 _needsSave . . . 332
12.22.5.9 _opacity . . . 332
12.22.5.10 _parameters . . . 332
12.22.5.11 _script . . . 332
12.22.5.12 _scriptGuid . . . 332
12.22.5.13 _scriptName . . . 332
12.22.5.14 _tempParameters . . . 332
12.22.5.15 kpRepeatMsec . . . 332
12.22.5.16 kpRepeatTime . . . 332
12.22.5.17 kpStopTime . . . 333
12.22.5.18 repeatKey . . . 333
12.22.5.19 repeatMsec . . . 333
12.22.5.20 repeatTime . . . 333
12.22.5.21 stopTime . . . 333
12.23 Ui::KbAnimWidget Class Reference . . . 333
12.23.1 Detailed Description . . . 335
12.23.2 Member Function Documentation . . . 335
12.23.2.1 retranslateUi . . . 335
12.23.2.2 setupUi . . . 336
12.23.3 Field Documentation . . . 339
12.23.3.1 aCopyLabel . . . 339
12.23.3.2 aMissingLabel . . . 339
12.23.3.3 aMissingPage . . . 339
12.23.3.4 aNameLabel . . . 339

12.23.3.5 animList . . . 339
12.23.3.6 animPage . . . 340
12.23.3.7 aVerLabel . . . 340
12.23.3.8 blendBox . . . 340
12.23.3.9 deleteButton . . . 340
12.23.3.10 gridLayout . . . 340
12.23.3.11 gridLayout_2 . . . 340
12.23.3.12 gridLayout_3 . . . 340
12.23.3.13 gridLayout_4 . . . 340
12.23.3.14 horizontalLayout . . . 340
12.23.3.15 horizontalSpacer . . . 341
12.23.3.16 horizontalSpacer_2 . . . 341
12.23.3.17 horizontalSpacer_3 . . . 341
12.23.3.18 horizontalSpacer_4 . . . 341
12.23.3.19 horizontalSpacer_5 . . . 341
12.23.3.20 keyButton . . . 341
12.23.3.21 label_5 . . . 341
12.23.3.22 label_6 . . . 341
12.23.3.23 label_7 . . . 341
12.23.3.24 nameBox . . . 342
12.23.3.25 noAnimLabel . . . 342
12.23.3.26 noSelPage . . . 342
12.23.3.27 opacityBox . . . 342
12.23.3.28 propertyButton . . . 342
12.23.3.29 selectionStack . . . 342
12.23.3.30 verticalLayout . . . 342
12.23.3.31 verticalSpacer . . . 342
12.23.3.32 verticalSpacer_2 . . . 342
12.23.3.33 verticalSpacer_3 . . . 343
12.24 KbAnimWidget Class Reference . . . 343
12.24.1 Detailed Description . . . 345
12.24.2 Constructor & Destructor Documentation . . . 345
12.24.2.1 KbAnimWidget . . . 345
12.24.2.2 ~KbAnimWidget . . . 346
12.24.3 Member Function Documentation . . . 346
12.24.3.1 addAnim . . . 346
12.24.3.2 animChanged . . . 347
12.24.3.3 clearSelection . . . 348
12.24.3.4 didUpdateSelection . . . 348
12.24.3.5 duplicateAnim . . . 349

12.24.3.6 on_animList_currentItemChanged . . . 350
12.24.3.7 on_animList_customContextMenuRequested . . . 350
12.24.3.8 on_animList_itemChanged . . . 351
12.24.3.9 on_blendBox_activated . . . 351
12.24.3.10 on_deleteButton_clicked . . . 352
12.24.3.11 on_keyButton_clicked . . . 353
12.24.3.12 on_nameBox_textEdited . . . 353
12.24.3.13 on_opacityBox_valueChanged . . . 354
12.24.3.14 on_propertyButton_clicked . . . 354
12.24.3.15 refreshList . . . 355
12.24.3.16 reorderAnims . . . 356
12.24.3.17 setCurrent . . . 357
12.24.3.18 setLight . . . 358
12.24.3.19 setSelectedKeys . . . 359
12.24.4 Field Documentation . . . 360
12.24.4.1 animations . . . 360
12.24.4.2 current . . . 360
12.24.4.3 light . . . 360
12.24.4.4 noReorder . . . 360
12.24.4.5 selectedKeys . . . 360
12.24.4.6 ui . . . 360
12.25 KbBind Class Reference . . . 361
12.25.1 Detailed Description . . . 364
12.25.2 Constructor & Destructor Documentation . . . 364
12.25.2.1 KbBind . . . 364
12.25.2.2 KbBind . . . 364
12.25.3 Member Function Documentation . . . 365
12.25.3.1 action . . . 365
12.25.3.2 bindAction . . . 366
12.25.3.3 defaultAction . . . 366
12.25.3.4 defaultAction . . . 367
12.25.3.5 devParent . . . 367
12.25.3.6 didLoad . . . 368
12.25.3.7 friendlyActionName . . . 368
12.25.3.8 friendlyName . . . 369
12.25.3.9 getMacroNumber . . . 369
12.25.3.10 getMacroPath . . . 370
12.25.3.11 globalRemap . . . 371
12.25.3.12 handleNotificationChannel . . . 371
12.25.3.13 isISO . . . 372

12.25.3.14 isKeyboard . . . 373
12.25.3.15 isMouse . . . 373
12.25.3.16 keyEvent . . . 374
12.25.3.17 layoutChanged . . . 375
12.25.3.18 light . . . 376
12.25.3.19 load . . . 377
12.25.3.20 loadGlobalRemap . . . 378
12.25.3.21 map . . . 378
12.25.3.22 map . . . 379
12.25.3.23 modeParent . . . 379
12.25.3.24 needsSave . . . 380
12.25.3.25 noAction . . . 380
12.25.3.26 noAction . . . 380
12.25.3.27 perf . . . 381
12.25.3.28 resetAction . . . 382
12.25.3.29 resetAction . . . 382
12.25.3.30 save . . . 383
12.25.3.31 saveGlobalRemap . . . 384
12.25.3.32 setAction . . . 385
12.25.3.33 setAction . . . 386
12.25.3.34 setGlobalRemap . . . 386
12.25.3.35 setNeedsUpdate . . . 387
12.25.3.36 update . . . 387
12.25.3.37 updated . . . 389
12.25.3.38 winLock . . . 389
12.25.3.39 winLock . . . 390
12.25.4 Friends And Related Function Documentation . . . 390
12.25.4.1 KeyAction . . . 390
12.25.5 Field Documentation . . . 390
12.25.5.1 _bind . . . 390
12.25.5.2 _devParent . . . 390
12.25.5.3 _globalRemap . . . 390
12.25.5.4 _map . . . 390
12.25.5.5 _needsSave . . . 391
12.25.5.6 _needsUpdate . . . 391
12.25.5.7 _winLock . . . 391
12.25.5.8 globalRemapTime . . . 391
12.25.5.9 lastCmd . . . 391
12.25.5.10 lastGlobalRemapTime . . . 391
12.26 Ui::KbBindWidget Class Reference . . . 391

12.26.1 Detailed Description . . . 393
12.26.2 Member Function Documentation . . . 393
12.26.2.1 retranslateUi . . . 393
12.26.2.2 setupUi . . . 394
12.26.3 Field Documentation . . . 396
12.26.3.1 copyButton . . . 396
12.26.3.2 gridLayout . . . 396
12.26.3.3 gridLayout_2 . . . 396
12.26.3.4 horizontalLayout_2 . . . 396
12.26.3.5 horizontalSpacer . . . 396
12.26.3.6 horizontalSpacer_2 . . . 396
12.26.3.7 keyWidget . . . 396
12.26.3.8 label . . . 397
12.26.3.9 label_2 . . . 397
12.26.3.10line . . . 397
12.26.3.11line_2 . . . 397
12.26.3.12rbWidget . . . 397
12.26.3.13resetButton . . . 397
12.26.3.14selectLabel . . . 397
12.26.3.15verticalLayout . . . 397
12.26.3.16verticalSpacer . . . 397
12.26.3.17widget . . . 398
12.27KbBindWidget Class Reference . . . 398
12.27.1 Detailed Description . . . 399
12.27.2 Constructor & Destructor Documentation . . . 400
12.27.2.1 KbBindWidget . . . 400
12.27.2.2 ∼KbBindWidget . . . 400
12.27.3 Member Function Documentation . . . 401
12.27.3.1 newLayout . . . 401
12.27.3.2 newSelection . . . 402
12.27.3.3 on_copyButton_clicked . . . 403
12.27.3.4 on_resetButton_clicked . . . 404
12.27.3.5 setBind . . . 405
12.27.3.6 updateBind . . . 406
12.27.3.7 updateSelDisplay . . . 407
12.27.4 Field Documentation . . . 408
12.27.4.1 bind . . . 408
12.27.4.2 currentSelection . . . 408
12.27.4.3 profile . . . 409
12.27.4.4 ui . . . 409

12.28KbFirmware Class Reference . . . 409
12.28.1 Detailed Description . . . 412
12.28.2 Member Enumeration Documentation . . . 412
12.28.2.1 anonymous enum . . . 412
12.28.3 Constructor & Destructor Documentation . . . 412
12.28.3.1 KbFirmware . . . 412
12.28.3.2 ~KbFirmware . . . 412
12.28.4 Member Function Documentation . . . 412
12.28.4.1 _checkUpdates . . . 412
12.28.4.2 _fileForBoard . . . 413
12.28.4.3 _latestForBoard . . . 414
12.28.4.4 checkUpdates . . . 415
12.28.4.5 dataForBoard . . . 415
12.28.4.6 downloaded . . . 416
12.28.4.7 downloadFinished . . . 416
12.28.4.8 hasDownloaded . . . 417
12.28.4.9 processDownload . . . 417
12.28.4.10versionForBoard . . . 419
12.28.5 Field Documentation . . . 420
12.28.5.1 fwTable . . . 420
12.28.5.2 fwTableHash . . . 420
12.28.5.3 hasGPG . . . 420
12.28.5.4 instance . . . 420
12.28.5.5 lastCheck . . . 420
12.28.5.6 lastFinished . . . 420
12.28.5.7 networkManager . . . 420
12.28.5.8 tableDownload . . . 420
12.29KbLight Class Reference . . . 421
12.29.1 Detailed Description . . . 423
12.29.2 Member Typedef Documentation . . . 424
12.29.2.1 AnimList . . . 424
12.29.3 Constructor & Destructor Documentation . . . 424
12.29.3.1 KbLight . . . 424
12.29.3.2 KbLight . . . 424
12.29.3.3 ~KbLight . . . 425
12.29.4 Member Function Documentation . . . 425
12.29.4.1 addAnim . . . 425
12.29.4.2 animKeypress . . . 426
12.29.4.3 animList . . . 427
12.29.4.4 animList . . . 427

12.29.4.5 base . . . 427
12.29.4.6 close . . . 428
12.29.4.7 color . . . 429
12.29.4.8 color . . . 430
12.29.4.9 colorMap . . . 431
12.29.4.10 didLoad . . . 431
12.29.4.11 dimming . . . 431
12.29.4.12 dimming . . . 432
12.29.4.13 duplicateAnim . . . 432
12.29.4.14 findAnim . . . 433
12.29.4.15 findAnimIdx . . . 434
12.29.4.16 frameDisplayed . . . 434
12.29.4.17 frameUpdate . . . 435
12.29.4.18 isStarted . . . 436
12.29.4.19 load . . . 437
12.29.4.20 map . . . 438
12.29.4.21 map . . . 439
12.29.4.22 needsSave . . . 440
12.29.4.23 open . . . 440
12.29.4.24 previewAnim . . . 441
12.29.4.25 printRGB . . . 442
12.29.4.26 rebuildBaseMap . . . 443
12.29.4.27 resetIndicators . . . 443
12.29.4.28 restartAnimation . . . 444
12.29.4.29 save . . . 445
12.29.4.30 setIndicator . . . 446
12.29.4.31 shareDimming . . . 446
12.29.4.32 shareDimming . . . 447
12.29.4.33 stopPreview . . . 447
12.29.4.34 updated . . . 448
12.29.5 Field Documentation . . . 448
12.29.5.1 _animList . . . 448
12.29.5.2 _animMap . . . 448
12.29.5.3 _colorMap . . . 449
12.29.5.4 _dimming . . . 449
12.29.5.5 _indicatorList . . . 449
12.29.5.6 _indicatorMap . . . 449
12.29.5.7 _lastFrameAnimMap . . . 449
12.29.5.8 _lastFrameDimming . . . 449
12.29.5.9 _lastFrameIndicatorMap . . . 449

12.29.5.10 _map . . . 449
12.29.5.11 _needsMapRefresh . . . 449
12.29.5.12 _needsSave . . . 450
12.29.5.13 _previewAnim . . . 450
12.29.5.14 _qColorMap . . . 450
12.29.5.15 _start . . . 450
12.29.5.16 lastFrameSignal . . . 450
12.29.5.17 MAX_DIM . . . 450
12.30 Ui::KbLightWidget Class Reference . . . 450
12.30.1 Detailed Description . . . 452
12.30.2 Member Function Documentation . . . 452
12.30.2.1 retranslateUi . . . 452
12.30.2.2 setupUi . . . 453
12.30.3 Field Documentation . . . 456
12.30.3.1 animButton . . . 456
12.30.3.2 animWidget . . . 456
12.30.3.3 bgButton . . . 456
12.30.3.4 brightnessBox . . . 456
12.30.3.5 gridLayout . . . 456
12.30.3.6 gridLayout_2 . . . 456
12.30.3.7 gridLayout_4 . . . 456
12.30.3.8 horizontalLayout . . . 457
12.30.3.9 horizontalLayout_2 . . . 457
12.30.3.10 horizontalLayout_3 . . . 457
12.30.3.11 horizontalSpacer_3 . . . 457
12.30.3.12 horizontalSpacer_4 . . . 457
12.30.3.13 horizontalSpacer_5 . . . 457
12.30.3.14 horizontalSpacer_6 . . . 457
12.30.3.15 keyWidget . . . 457
12.30.3.16 label . . . 458
12.30.3.17 label_2 . . . 458
12.30.3.18 label_3 . . . 458
12.30.3.19 line . . . 458
12.30.3.20 line_2 . . . 458
12.30.3.21 selLabel . . . 458
12.30.3.22 showAnimBox . . . 458
12.30.3.23 verticalLayout . . . 458
12.30.3.24 verticalSpacer_2 . . . 458
12.30.3.25 verticalSpacer_3 . . . 459
12.30.3.26 widget . . . 459

12.31 KbLightWidget Class Reference . . . 459
12.31.1 Detailed Description . . . 461
12.31.2 Constructor & Destructor Documentation . . . 461
12.31.2.1 KbLightWidget . . . 461
12.31.2.2 ~KbLightWidget . . . 462
12.31.3 Member Function Documentation . . . 462
12.31.3.1 changeAnim . . . 462
12.31.3.2 changeAnimKeys . . . 462
12.31.3.3 changeColor . . . 463
12.31.3.4 newSelection . . . 464
12.31.3.5 on_animButton_clicked . . . 465
12.31.3.6 on_bgButton_clicked . . . 466
12.31.3.7 on_brightnessBox_activated . . . 466
12.31.3.8 on_showAnimBox_clicked . . . 467
12.31.3.9 setLight . . . 467
12.31.3.10 setMonochrome . . . 469
12.31.3.11 toggleSidelight . . . 469
12.31.3.12 updateLight . . . 470
12.31.4 Field Documentation . . . 471
12.31.4.1 currentSelection . . . 471
12.31.4.2 light . . . 471
12.31.4.3 ui . . . 471
12.32 KbManager Class Reference . . . 472
12.32.1 Detailed Description . . . 474
12.32.2 Constructor & Destructor Documentation . . . 474
12.32.2.1 KbManager . . . 474
12.32.3 Member Function Documentation . . . 475
12.32.3.1 ckbDaemonVersion . . . 475
12.32.3.2 ckbDaemonVersionF . . . 475
12.32.3.3 ckbGuiVersion . . . 476
12.32.3.4 ckbGuiVersionF . . . 476
12.32.3.5 devices . . . 477
12.32.3.6 eventTimer . . . 477
12.32.3.7 fps . . . 477
12.32.3.8 init . . . 478
12.32.3.9 kbConnected . . . 479
12.32.3.10 kbDisconnected . . . 479
12.32.3.11 kbManager . . . 479
12.32.3.12 parseVersionString . . . 480
12.32.3.13 scanKeyboards . . . 480

12.32.3.14 scanTimer . . . 482
12.32.3.15 stop . . . 482
12.32.3.16 versionUpdated . . . 483
12.32.4 Field Documentation . . . 483
12.32.4.1 _daemonVersion . . . 483
12.32.4.2 _devices . . . 483
12.32.4.3 _eventTimer . . . 483
12.32.4.4 _guiVersion . . . 484
12.32.4.5 _kbManager . . . 484
12.32.4.6 _scanTimer . . . 484
12.33 KbMode Class Reference . . . 484
12.33.1 Detailed Description . . . 487
12.33.2 Constructor & Destructor Documentation . . . 487
12.33.2.1 KbMode . . . 487
12.33.2.2 KbMode . . . 488
12.33.2.3 KbMode . . . 488
12.33.2.4 ~KbMode . . . 489
12.33.3 Member Function Documentation . . . 489
12.33.3.1 bind . . . 489
12.33.3.2 doUpdate . . . 490
12.33.3.3 id . . . 490
12.33.3.4 id . . . 491
12.33.3.5 keyMap . . . 491
12.33.3.6 light . . . 492
12.33.3.7 name . . . 493
12.33.3.8 name . . . 493
12.33.3.9 needsSave . . . 493
12.33.3.10 newId . . . 494
12.33.3.11 perf . . . 495
12.33.3.12 save . . . 495
12.33.3.13 setNeedsSave . . . 496
12.33.3.14 setNeedsUpdate . . . 497
12.33.3.15 updated . . . 497
12.33.4 Field Documentation . . . 498
12.33.4.1 _bind . . . 498
12.33.4.2 _id . . . 498
12.33.4.3 _light . . . 498
12.33.4.4 _name . . . 498
12.33.4.5 _needsSave . . . 498
12.33.4.6 _perf . . . 498

12.34 KbPerf Class Reference . . . 498
12.34.1 Detailed Description . . . 502
12.34.2 Member Enumeration Documentation . . . 502
12.34.2.1 height . . . 502
12.34.2.2 i_hw . . . 503
12.34.2.3 indicator . . . 503
12.34.3 Constructor & Destructor Documentation . . . 503
12.34.3.1 KbPerf . . . 503
12.34.3.2 KbPerf . . . 504
12.34.4 Member Function Documentation . . . 505
12.34.4.1 _curDpi . . . 505
12.34.4.2 angleSnap . . . 505
12.34.4.3 angleSnap . . . 506
12.34.4.4 applyIndicators . . . 506
12.34.4.5 baseDpiIdx . . . 507
12.34.4.6 baseDpiIdx . . . 508
12.34.4.7 bind . . . 508
12.34.4.8 didLoad . . . 509
12.34.4.9 dpi . . . 509
12.34.4.10 dpi . . . 510
12.34.4.11 dpi . . . 511
12.34.4.12 dpiColor . . . 511
12.34.4.13 dpiColor . . . 512
12.34.4.14 dpiCycleDown . . . 512
12.34.4.15 dpiCycleUp . . . 513
12.34.4.16 dpiDown . . . 513
12.34.4.17 dpiEnabled . . . 514
12.34.4.18 dpiEnabled . . . 514
12.34.4.19 dpiIndicator . . . 515
12.34.4.20 dpiIndicator . . . 515
12.34.4.21 dpiUp . . . 515
12.34.4.22 getIndicator . . . 516
12.34.4.23 lOpacity . . . 516
12.34.4.24 lOpacity . . . 517
12.34.4.25 liftHeight . . . 517
12.34.4.26 liftHeight . . . 517
12.34.4.27 light . . . 518
12.34.4.28 lightIndicator . . . 518
12.34.4.29 load . . . 519
12.34.4.30 modeParent . . . 521

12.34.4.31 needsSave . . . 521
12.34.4.32 operator= . . . 522
12.34.4.33 popDpi . . . 522
12.34.4.34 pushDpi . . . 523
12.34.4.35 pushDpi . . . 524
12.34.4.36 pushSniper . . . 525
12.34.4.37 save . . . 525
12.34.4.38 setIndicator . . . 526
12.34.4.39 setNeedsUpdate . . . 527
12.34.4.40 settingsUpdated . . . 527
12.34.4.41 sniperDpi . . . 528
12.34.4.42 sniperDpi . . . 528
12.34.4.43 sniperDpi . . . 528
12.34.4.44 update . . . 529
12.34.5 Field Documentation . . . 530
12.34.5.1 _angleSnap . . . 530
12.34.5.2 _dpiIndicator . . . 530
12.34.5.3 _iOpacity . . . 531
12.34.5.4 _liftHeight . . . 531
12.34.5.5 _needsSave . . . 531
12.34.5.6 _needsUpdate . . . 531
12.34.5.7 DPI_COUNT . . . 531
12.34.5.8 DPI_MAX . . . 531
12.34.5.9 DPI_MIN . . . 531
12.34.5.10 dpiBaseIdx . . . 531
12.34.5.11 dpiClr . . . 531
12.34.5.12 dpiCurX . . . 532
12.34.5.13 dpiCurY . . . 532
12.34.5.14 dpiOn . . . 532
12.34.5.15 dpiX . . . 532
12.34.5.16 dpiY . . . 532
12.34.5.17 HW_I_COUNT . . . 532
12.34.5.18 hwIType . . . 532
12.34.5.19 I_COUNT . . . 532
12.34.5.20 iColor . . . 532
12.34.5.21 iEnable . . . 533
12.34.5.22 light100Color . . . 533
12.34.5.23 muteNAColor . . . 533
12.34.5.24 OTHER . . . 533
12.34.5.25 pushedDpis . . . 533

12.34.5.26 runningPushIdx . . . 533
12.34.5.27 SNIPER . . . 533
12.35 KbProfile Class Reference . . . 533
12.35.1 Detailed Description . . . 536
12.35.2 Member Typedef Documentation . . . 536
12.35.2.1 ModeList . . . 536
12.35.3 Constructor & Destructor Documentation . . . 536
12.35.3.1 KbProfile . . . 536
12.35.3.2 KbProfile . . . 537
12.35.3.3 KbProfile . . . 537
12.35.4 Member Function Documentation . . . 538
12.35.4.1 append . . . 538
12.35.4.2 currentMode . . . 538
12.35.4.3 currentMode . . . 539
12.35.4.4 find . . . 539
12.35.4.5 id . . . 540
12.35.4.6 id . . . 540
12.35.4.7 indexOf . . . 541
12.35.4.8 insert . . . 541
12.35.4.9 keyMap . . . 542
12.35.4.10 keyMap . . . 542
12.35.4.11 modeCount . . . 543
12.35.4.12 modes . . . 543
12.35.4.13 modes . . . 544
12.35.4.14 move . . . 544
12.35.4.15 name . . . 545
12.35.4.16 name . . . 545
12.35.4.17 needsSave . . . 546
12.35.4.18 newId . . . 546
12.35.4.19 removeAll . . . 547
12.35.4.20 save . . . 548
12.35.4.21 setNeedsSave . . . 549
12.35.4.22 setNeedsUpdate . . . 549
12.35.5 Field Documentation . . . 550
12.35.5.1 _currentMode . . . 550
12.35.5.2 _id . . . 550
12.35.5.3 _keyMap . . . 550
12.35.5.4 _modes . . . 550
12.35.5.5 _name . . . 551
12.35.5.6 _needsSave . . . 551

12.36 KbProfileDialog Class Reference . . . 551
12.36.1 Detailed Description . . . 552
12.36.2 Constructor & Destructor Documentation . . . 553
12.36.2.1 KbProfileDialog . . . 553
12.36.2.2 ~KbProfileDialog . . . 553
12.36.3 Member Function Documentation . . . 553
12.36.3.1 addNewProfileItem . . . 553
12.36.3.2 on_profileList_currentItemChanged . . . 554
12.36.3.3 on_profileList_customContextMenuRequested . . . 554
12.36.3.4 on_profileList_itemChanged . . . 556
12.36.3.5 on_profileList_itemClicked . . . 557
12.36.3.6 profileList_reordered . . . 558
12.36.3.7 repopulate . . . 558
12.36.4 Field Documentation . . . 559
12.36.4.1 device . . . 559
12.36.4.2 GUID . . . 560
12.36.4.3 NEW_FLAG . . . 560
12.36.4.4 ui . . . 560
12.37 Ui::KbProfileDialog Class Reference . . . 560
12.37.1 Detailed Description . . . 562
12.37.2 Member Function Documentation . . . 562
12.37.2.1 retranslateUi . . . 562
12.37.2.2 setupUi . . . 563
12.37.3 Field Documentation . . . 564
12.37.3.1 buttonBox . . . 564
12.37.3.2 gridLayout . . . 564
12.37.3.3 horizontalSpacer . . . 564
12.37.3.4 horizontalSpacer_2 . . . 564
12.37.3.5 profileList . . . 564
12.38 KbWidget Class Reference . . . 565
12.38.1 Detailed Description . . . 567
12.38.2 Constructor & Destructor Documentation . . . 567
12.38.2.1 KbWidget . . . 567
12.38.2.2 ~KbWidget . . . 569
12.38.3 Member Function Documentation . . . 569
12.38.3.1 addNewModeItem . . . 569
12.38.3.2 devUpdate . . . 570
12.38.3.3 modeChanged . . . 570
12.38.3.4 modeIcon . . . 572
12.38.3.5 modesList_reordered . . . 573

12.38.3.6 modeUpdate . . . 574
12.38.3.7 name . . . 574
12.38.3.8 on_fwUpdButton_clicked . . . 575
12.38.3.9 on_hwSaveButton_clicked . . . 576
12.38.3.10 on_modesList_currentItemChanged . . . 577
12.38.3.11 on_modesList_customContextMenuRequested . . . 578
12.38.3.12 on_modesList_itemChanged . . . 580
12.38.3.13 on_modesList_itemClicked . . . 580
12.38.3.14 on_profileBox_activated . . . 581
12.38.3.15 on_tabWidget_currentChanged . . . 581
12.38.3.16 profileChanged . . . 582
12.38.3.17 showFirstTab . . . 583
12.38.3.18 showFwUpdate . . . 583
12.38.3.19 showLastTab . . . 584
12.38.3.20 updateFwButton . . . 584
12.38.3.21 updateProfileList . . . 585
12.38.4 Field Documentation . . . 586
12.38.4.1 currentMode . . . 586
12.38.4.2 device . . . 586
12.38.4.3 GUID . . . 587
12.38.4.4 hasShownNewFW . . . 587
12.38.4.5 lastAutoSave . . . 587
12.38.4.6 NEW_FLAG . . . 587
12.38.4.7 ui . . . 587
12.39 Ui::KbWidget Class Reference . . . 587
12.39.1 Detailed Description . . . 589
12.39.2 Member Function Documentation . . . 589
12.39.2.1 retranslateUi . . . 589
12.39.2.2 setupUi . . . 590
12.39.3 Field Documentation . . . 593
12.39.3.1 bindTab . . . 593
12.39.3.2 bindWidget . . . 593
12.39.3.3 devTab . . . 593
12.39.3.4 fwLabel . . . 593
12.39.3.5 fwUpdButton . . . 594
12.39.3.6 fwUpdLabel . . . 594
12.39.3.7 fwUpdLayout . . . 594
12.39.3.8 gridLayout . . . 594
12.39.3.9 gridLayout_2 . . . 594
12.39.3.10 gridLayout_3 . . . 594

12.39.3.11 gridLayout_4 . . . 594
12.39.3.12 gridLayout_5 . . . 594
12.39.3.13 gridLayout_7 . . . 594
12.39.3.14 horizontalSpacer_2 . . . 595
12.39.3.15 horizontalSpacer_3 . . . 595
12.39.3.16 hwSaveButton . . . 595
12.39.3.17 kPerfTab . . . 595
12.39.3.18 kPerfWidget . . . 595
12.39.3.19 label_3 . . . 595
12.39.3.20 label_4 . . . 595
12.39.3.21 label_9 . . . 595
12.39.3.22 lightTab . . . 595
12.39.3.23 lightWidget . . . 596
12.39.3.24 modesList . . . 596
12.39.3.25 mPerfTab . . . 596
12.39.3.26 mPerfWidget . . . 596
12.39.3.27 pollLabel . . . 596
12.39.3.28 pollLabel2 . . . 596
12.39.3.29 profileBox . . . 596
12.39.3.30 serialLabel . . . 596
12.39.3.31 tabWidget . . . 596
12.39.3.32 verticalSpacer . . . 597
12.40 Key Struct Reference . . . 597
12.40.1 Detailed Description . . . 598
12.40.2 Member Function Documentation . . . 598
12.40.2.1 friendlyName . . . 598
12.40.2.2 operator bool . . . 598
12.40.2.3 operator! . . . 599
12.40.2.4 storageName . . . 599
12.40.3 Field Documentation . . . 599
12.40.3.1 _friendlyName . . . 599
12.40.3.2 _storageName . . . 599
12.40.3.3 hasLed . . . 599
12.40.3.4 hasScan . . . 599
12.40.3.5 height . . . 600
12.40.3.6 name . . . 600
12.40.3.7 width . . . 600
12.40.3.8 x . . . 600
12.40.3.9 y . . . 600
12.41 KeyAction Class Reference . . . 600

12.41.1 Detailed Description . . . 604
12.41.2 Member Enumeration Documentation . . . 604
12.41.2.1 Type . . . 604
12.41.3 Constructor & Destructor Documentation . . . 605
12.41.3.1 KeyAction . . . 605
12.41.3.2 KeyAction . . . 605
12.41.3.3 ~KeyAction . . . 605
12.41.3.4 KeyAction . . . 605
12.41.4 Member Function Documentation . . . 605
12.41.4.1 adjustDisplay . . . 605
12.41.4.2 animAction . . . 606
12.41.4.3 animInfo . . . 607
12.41.4.4 defaultAction . . . 607
12.41.4.5 dpiAction . . . 608
12.41.4.6 dpiInfo . . . 609
12.41.4.7 driverName . . . 609
12.41.4.8 friendlyName . . . 610
12.41.4.9 isAnim . . . 611
12.41.4.10 isDPI . . . 612
12.41.4.11 isMacro . . . 612
12.41.4.12 isMedia . . . 612
12.41.4.13 isMouse . . . 613
12.41.4.14 isNormal . . . 613
12.41.4.15 isProgram . . . 614
12.41.4.16 isSpecial . . . 614
12.41.4.17 isUnbound . . . 615
12.41.4.18 isValidMacro . . . 615
12.41.4.19 keyEvent . . . 616
12.41.4.20 keyRelease . . . 620
12.41.4.21 lightAction . . . 620
12.41.4.22 lockAction . . . 620
12.41.4.23 macroAction . . . 620
12.41.4.24 macroContent . . . 621
12.41.4.25 macroDisplay . . . 622
12.41.4.26 macroFullLine . . . 622
12.41.4.27 macroLine . . . 623
12.41.4.28 macroTiming . . . 624
12.41.4.29 modeAction . . . 624
12.41.4.30 noAction . . . 625
12.41.4.31 operator QString . . . 625

12.41.4.32operator= . . . 625
12.41.4.33programAction . . . 625
12.41.4.34programInfo . . . 625
12.41.4.35specialInfo . . . 626
12.41.4.36type . . . 627
12.41.4.37value . . . 627
12.41.5 Field Documentation . . . 627
12.41.5.1 _value . . . 627
12.41.5.2 DPI_CUSTOM . . . 628
12.41.5.3 DPI_CYCLE_DOWN . . . 628
12.41.5.4 DPI_CYCLE_UP . . . 628
12.41.5.5 DPI_DOWN . . . 628
12.41.5.6 DPI_SNIPER . . . 628
12.41.5.7 DPI_UP . . . 628
12.41.5.8 LIGHT_DOWN . . . 628
12.41.5.9 LIGHT_DOWN_WRAP . . . 628
12.41.5.10LIGHT_UP . . . 628
12.41.5.11LIGHT_UP_WRAP . . . 629
12.41.5.12LOCK_OFF . . . 629
12.41.5.13LOCK_ON . . . 629
12.41.5.14LOCK_TOGGLE . . . 629
12.41.5.15MODE_NEXT . . . 629
12.41.5.16MODE_NEXT_WRAP . . . 629
12.41.5.17MODE_PREV . . . 629
12.41.5.18MODE_PREV_WRAP . . . 629
12.41.5.19preProgram . . . 629
12.41.5.20PROGRAM_PR_INDEF . . . 630
12.41.5.21PROGRAM_PR_KPSTOP . . . 630
12.41.5.22PROGRAM_PR_KRSTOP . . . 630
12.41.5.23PROGRAM_PR_MULTI . . . 630
12.41.5.24PROGRAM_RE_INDEF . . . 630
12.41.5.25PROGRAM_RE_KPSTOP . . . 630
12.41.5.26PROGRAM_RE_MULTI . . . 630
12.41.5.27relProgram . . . 630
12.41.5.28sniperValue . . . 630
12.42KeyMap Class Reference . . . 631
12.42.1 Detailed Description . . . 633
12.42.2 Member Enumeration Documentation . . . 633
12.42.2.1 Layout . . . 633
12.42.2.2 Model . . . 634

12.42.3 Constructor & Destructor Documentation . . . 634
12.42.3.1 KeyMap . . . 634
12.42.3.2 KeyMap . . . 635
12.42.4 Member Function Documentation . . . 635
12.42.4.1 byPosition . . . 635
12.42.4.2 contains . . . 636
12.42.4.3 count . . . 636
12.42.4.4 friendlyName . . . 637
12.42.4.5 fromName . . . 638
12.42.4.6 fromStorage . . . 639
12.42.4.7 getLayout . . . 640
12.42.4.8 getLayout . . . 640
12.42.4.9 getModel . . . 641
12.42.4.10 getModel . . . 642
12.42.4.11 height . . . 642
12.42.4.12 isISO . . . 643
12.42.4.13 isISO . . . 643
12.42.4.14 isJP . . . 644
12.42.4.15 isJP . . . 644
12.42.4.16 isKeyboard . . . 645
12.42.4.17 isKeyboard . . . 645
12.42.4.18 isMouse . . . 646
12.42.4.19 isMouse . . . 646
12.42.4.20 isPTBR . . . 647
12.42.4.21 isPTBR . . . 647
12.42.4.22 key . . . 648
12.42.4.23 keys . . . 649
12.42.4.24 layout . . . 649
12.42.4.25 layoutNames . . . 649
12.42.4.26 locale . . . 650
12.42.4.27 map . . . 651
12.42.4.28 model . . . 651
12.42.4.29 modelHeight . . . 652
12.42.4.30 modelWidth . . . 652
12.42.4.31 name . . . 652
12.42.4.32 operator const QHash< QString, Key > & . . . 653
12.42.4.33 operator[] . . . 653
12.42.4.34 positions . . . 653
12.42.4.35 strLayout . . . 654
12.42.4.36 strModel . . . 654

12.42.4.37 toStorage . . . 655
12.42.4.38 width . . . 655
12.42.5 Field Documentation . . . 656
12.42.5.1 _keys . . . 656
12.42.5.2 keyHeight . . . 656
12.42.5.3 keyLayout . . . 656
12.42.5.4 keyModel . . . 656
12.42.5.5 keyWidth . . . 656
12.43 KeyWidget Class Reference . . . 656
12.43.1 Detailed Description . . . 659
12.43.2 Member Typedef Documentation . . . 659
12.43.2.1 BindMap . . . 659
12.43.3 Member Enumeration Documentation . . . 660
12.43.3.1 anonymous enum . . . 660
12.43.4 Constructor & Destructor Documentation . . . 660
12.43.4.1 KeyWidget . . . 660
12.43.5 Member Function Documentation . . . 660
12.43.5.1 bindMap . . . 660
12.43.5.2 bindMap . . . 661
12.43.5.3 clearAnimation . . . 661
12.43.5.4 clearSelection . . . 661
12.43.5.5 colorMap . . . 662
12.43.5.6 colorMap . . . 662
12.43.5.7 displayColorMap . . . 662
12.43.5.8 drawInfo . . . 663
12.43.5.9 map . . . 663
12.43.5.10 map . . . 664
12.43.5.11 monochrome . . . 664
12.43.5.12 monochrome . . . 665
12.43.5.13 mouseMoveEvent . . . 665
12.43.5.14 mousePressEvent . . . 666
12.43.5.15 mouseReleaseEvent . . . 667
12.43.5.16 paintEvent . . . 668
12.43.5.17 rgbMode . . . 673
12.43.5.18 rgbMode . . . 674
12.43.5.19 selectAll . . . 674
12.43.5.20 selectionChanged . . . 675
12.43.5.21 setAnimation . . . 675
12.43.5.22 setAnimationToSelection . . . 676
12.43.5.23 setSelection . . . 676

12.43.5.24 sidelightToggled . . . 677
12.43.6 Field Documentation . . . 677
12.43.6.1 _bindMap . . . 677
12.43.6.2 _colorMap . . . 678
12.43.6.3 _displayColorMap . . . 678
12.43.6.4 _indicators . . . 678
12.43.6.5 _monochrome . . . 678
12.43.6.6 _rgbMode . . . 678
12.43.6.7 animation . . . 678
12.43.6.8 keyMap . . . 678
12.43.6.9 mouseCurrentX . . . 678
12.43.6.10 mouseCurrentY . . . 678
12.43.6.11 mouseDownMode . . . 679
12.43.6.12 mouseDownX . . . 679
12.43.6.13 mouseDownY . . . 679
12.43.6.14 newSelection . . . 679
12.43.6.15 selection . . . 679
12.44 Ui::KPerfWidget Class Reference . . . 679
12.44.1 Detailed Description . . . 683
12.44.2 Member Function Documentation . . . 683
12.44.2.1 retranslateUi . . . 683
12.44.2.2 setupUi . . . 684
12.44.3 Field Documentation . . . 691
12.44.3.1 capsBox . . . 691
12.44.3.2 capsColorOff . . . 691
12.44.3.3 capsColorOn . . . 691
12.44.3.4 copyButton . . . 691
12.44.3.5 gridLayout . . . 692
12.44.3.6 horizontalLayout . . . 692
12.44.3.7 horizontalLayout_2 . . . 692
12.44.3.8 horizontalSpacer . . . 692
12.44.3.9 horizontalSpacer_2 . . . 692
12.44.3.10 horizontalSpacer_3 . . . 692
12.44.3.11 horizontalSpacer_4 . . . 692
12.44.3.12 horizontalSpacer_5 . . . 692
12.44.3.13 horizontalSpacer_6 . . . 692
12.44.3.14 horizontalSpacer_7 . . . 693
12.44.3.15 horizontalSpacer_8 . . . 693
12.44.3.16 intensityBox . . . 693
12.44.3.17 k95Label1 . . . 693

12.44.3.18 k95Label2 . . . 693
12.44.3.19 k95Label3 . . . 693
12.44.3.20 k95Label4 . . . 693
12.44.3.21 k95Label5 . . . 693
12.44.3.22 k95Label6 . . . 693
12.44.3.23 k95Line . . . 694
12.44.3.24 k95Spacer . . . 694
12.44.3.25 label . . . 694
12.44.3.26 label_10 . . . 694
12.44.3.27 label_11 . . . 694
12.44.3.28 label_12 . . . 694
12.44.3.29 label_14 . . . 694
12.44.3.30 label_15 . . . 694
12.44.3.31 label_16 . . . 694
12.44.3.32 label_17 . . . 695
12.44.3.33 label_18 . . . 695
12.44.3.34 label_19 . . . 695
12.44.3.35 label_2 . . . 695
12.44.3.36 label_20 . . . 695
12.44.3.37 label_21 . . . 695
12.44.3.38 label_22 . . . 695
12.44.3.39 label_23 . . . 695
12.44.3.40 label_24 . . . 695
12.44.3.41 label_3 . . . 696
12.44.3.42 label_4 . . . 696
12.44.3.43 label_5 . . . 696
12.44.3.44 label_6 . . . 696
12.44.3.45 label_7 . . . 696
12.44.3.46 label_8 . . . 696
12.44.3.47 lightBox . . . 696
12.44.3.48 lightColor1 . . . 696
12.44.3.49 lightColor2 . . . 696
12.44.3.50 lightColor3 . . . 697
12.44.3.51 line . . . 697
12.44.3.52 line_2 . . . 697
12.44.3.53 line_3 . . . 697
12.44.3.54 lockBox . . . 697
12.44.3.55 lockColorOff . . . 697
12.44.3.56 lockColorOn . . . 697
12.44.3.57 macroBox . . . 697

12.44.3.58 macroColorOff . . . 697
12.44.3.59 macroColorOn . . . 698
12.44.3.60 modeBox . . . 698
12.44.3.61 modeColorOff . . . 698
12.44.3.62 modeColorOn . . . 698
12.44.3.63 muteBox . . . 698
12.44.3.64 muteColorNA . . . 698
12.44.3.65 muteColorOff . . . 698
12.44.3.66 muteColorOn . . . 698
12.44.3.67 numBox . . . 698
12.44.3.68 numColorOff . . . 699
12.44.3.69 numColorOn . . . 699
12.44.3.70 scrollBox . . . 699
12.44.3.71 scrollColorOff . . . 699
12.44.3.72 scrollColorOn . . . 699
12.44.3.73 verticalSpacer . . . 699
12.44.3.74 verticalSpacer_2 . . . 699
12.44.3.75 verticalSpacer_3 . . . 699
12.44.3.76 verticalSpacer_4 . . . 699
12.45 KPerfWidget Class Reference . . . 700
12.45.1 Detailed Description . . . 702
12.45.2 Data Structure Documentation . . . 702
12.45.2.1 struct KPerfWidget::IndicatorUi . . . 702
12.45.3 Member Typedef Documentation . . . 703
12.45.3.1 i_hw . . . 704
12.45.3.2 indicator . . . 704
12.45.4 Member Enumeration Documentation . . . 704
12.45.4.1 HwMode . . . 704
12.45.5 Constructor & Destructor Documentation . . . 704
12.45.5.1 KPerfWidget . . . 704
12.45.5.2 ~KPerfWidget . . . 706
12.45.6 Member Function Documentation . . . 706
12.45.6.1 mode2Raw . . . 706
12.45.6.2 on_copyButton_clicked . . . 707
12.45.6.3 on_intensityBox_valueChanged . . . 708
12.45.6.4 raw2Mode . . . 708
12.45.6.5 setPerf . . . 709
12.45.6.6 uiUpdated . . . 710
12.45.7 Field Documentation . . . 711
12.45.7.1 HW_I_COUNT . . . 711

12.45.7.2 I_COUNT . . . 712
12.45.7.3 indicators . . . 712
12.45.7.4 k95Widgets . . . 712
12.45.7.5 perf . . . 712
12.45.7.6 profile . . . 712
12.45.7.7 ui . . . 712
12.45.7.8 updateMapper . . . 712
12.46 Ui::LayoutDialog Class Reference . . . 712
12.46.1 Detailed Description . . . 714
12.46.2 Member Function Documentation . . . 714
12.46.2.1 retranslateUi . . . 714
12.46.2.2 setupUi . . . 715
12.46.3 Field Documentation . . . 716
12.46.3.1 buttonBox . . . 716
12.46.3.2 comboBox . . . 716
12.46.3.3 gridLayout . . . 716
12.46.3.4 horizontalSpacer . . . 716
12.46.3.5 horizontalSpacer_2 . . . 716
12.46.3.6 label . . . 717
12.46.3.7 label_2 . . . 717
12.46.3.8 verticalSpacer . . . 717
12.46.3.9 verticalSpacer_2 . . . 717
12.46.3.10 verticalSpacer_3 . . . 717
12.46.3.11 verticalSpacer_4 . . . 717
12.47 LayoutDialog Class Reference . . . 717
12.47.1 Detailed Description . . . 719
12.47.2 Constructor & Destructor Documentation . . . 719
12.47.2.1 LayoutDialog . . . 719
12.47.2.2 ~LayoutDialog . . . 719
12.47.3 Member Function Documentation . . . 720
12.47.3.1 selected . . . 720
12.47.4 Field Documentation . . . 720
12.47.4.1 ui . . . 720
12.48 MacroReader Class Reference . . . 720
12.48.1 Detailed Description . . . 722
12.48.2 Constructor & Destructor Documentation . . . 723
12.48.2.1 MacroReader . . . 723
12.48.2.2 MacroReader . . . 723
12.48.2.3 ~MacroReader . . . 723
12.48.3 Member Function Documentation . . . 723

12.48.3.1 startWorkInAThread . . . 723
12.48.4 Field Documentation . . . 724
12.48.4.1 macroReaderThread . . . 724
12.49 MacroReaderThread Class Reference . . . 724
12.49.1 Detailed Description . . . 726
12.49.2 Constructor & Destructor Documentation . . . 726
12.49.2.1 MacroReaderThread . . . 726
12.49.3 Member Function Documentation . . . 727
12.49.3.1 readMacro . . . 727
12.49.3.2 run . . . 727
12.49.4 Field Documentation . . . 728
12.49.4.1 macroBox . . . 728
12.49.4.2 macroNumber . . . 728
12.49.4.3 macroPath . . . 728
12.49.4.4 macroText . . . 728
12.50 Ui::MainWindow Class Reference . . . 728
12.50.1 Detailed Description . . . 730
12.50.2 Member Function Documentation . . . 730
12.50.2.1 retranslateUi . . . 730
12.50.2.2 setupUi . . . 730
12.50.3 Field Documentation . . . 731
12.50.3.1 actionAbout . . . 731
12.50.3.2 actionExit . . . 731
12.50.3.3 centralWidget . . . 732
12.50.3.4 gridLayout . . . 732
12.50.3.5 tabWidget . . . 732
12.51 MainWindow Class Reference . . . 732
12.51.1 Detailed Description . . . 735
12.51.2 Constructor & Destructor Documentation . . . 735
12.51.2.1 MainWindow . . . 735
12.51.2.2 ~MainWindow . . . 737
12.51.3 Member Function Documentation . . . 737
12.51.3.1 addDevice . . . 737
12.51.3.2 checkFwUpdates . . . 738
12.51.3.3 cleanup . . . 739
12.51.3.4 closeEvent . . . 739
12.51.3.5 iconClicked . . . 740
12.51.3.6 quitApp . . . 741
12.51.3.7 removeDevice . . . 741
12.51.3.8 showFwUpdateNotification . . . 742

12.51.3.9 showWindow . . . 743
12.51.3.10 stateChange . . . 743
12.51.3.11 timerTick . . . 744
12.51.3.12 toggleTrayIcon . . . 745
12.51.3.13 updateVersion . . . 745
12.51.4 Field Documentation . . . 746
12.51.4.1 closeAction . . . 746
12.51.4.2 kbWidgets . . . 746
12.51.4.3 mainWindow . . . 746
12.51.4.4 restoreAction . . . 747
12.51.4.5 settingsWidget . . . 747
12.51.4.6 trayIcon . . . 747
12.51.4.7 trayIconMenu . . . 747
12.51.4.8 ui . . . 747
12.52 Ui::ModeSelectDialog Class Reference . . . 747
12.52.1 Detailed Description . . . 749
12.52.2 Member Function Documentation . . . 749
12.52.2.1 retranslateUi . . . 749
12.52.2.2 setupUi . . . 749
12.52.3 Field Documentation . . . 750
12.52.3.1 buttonBox . . . 751
12.52.3.2 gridLayout . . . 751
12.52.3.3 label . . . 751
12.52.3.4 modeList . . . 751
12.52.3.5 selAllButton . . . 751
12.52.3.6 selNoneButton . . . 751
12.53 ModeSelectDialog Class Reference . . . 751
12.53.1 Detailed Description . . . 753
12.53.2 Constructor & Destructor Documentation . . . 753
12.53.2.1 ModeSelectDialog . . . 753
12.53.2.2 ~ModeSelectDialog . . . 754
12.53.3 Member Function Documentation . . . 754
12.53.3.1 on_selAllButton_clicked . . . 754
12.53.3.2 on_selNoneButton_clicked . . . 754
12.53.3.3 selection . . . 754
12.53.4 Field Documentation . . . 755
12.53.4.1 _modeList . . . 755
12.53.4.2 ui . . . 755
12.54 Ui::MPerfWidget Class Reference . . . 755
12.54.1 Detailed Description . . . 758

12.54.2 Member Function Documentation . . . 758
12.54.2.1 retranslateUi . . . 758
12.54.2.2 setupUi . . . 760
12.54.3 Field Documentation . . . 767
12.54.3.1 aSnapBox . . . 767
12.54.3.2 copyButton . . . 767
12.54.3.3 eBox1 . . . 768
12.54.3.4 eBox2 . . . 768
12.54.3.5 eBox3 . . . 768
12.54.3.6 eBox4 . . . 768
12.54.3.7 eBox5 . . . 768
12.54.3.8 gridLayout . . . 768
12.54.3.9 horizontalLayout . . . 768
12.54.3.10 horizontalLayout_2 . . . 768
12.54.3.11 horizontalLayout_4 . . . 768
12.54.3.12 horizontalSpacer . . . 769
12.54.3.13 horizontalSpacer_2 . . . 769
12.54.3.14 horizontalSpacer_3 . . . 769
12.54.3.15 horizontalSpacer_4 . . . 769
12.54.3.16 horizontalSpacer_5 . . . 769
12.54.3.17 horizontalSpacer_6 . . . 769
12.54.3.18 horizontalSpacer_7 . . . 769
12.54.3.19 iButton0 . . . 769
12.54.3.20 iButton1 . . . 769
12.54.3.21 iButton2 . . . 770
12.54.3.22 iButton3 . . . 770
12.54.3.23 iButton4 . . . 770
12.54.3.24 iButton5 . . . 770
12.54.3.25 iButtonO . . . 770
12.54.3.26 indicBox . . . 770
12.54.3.27 label . . . 770
12.54.3.28 label_11 . . . 770
12.54.3.29 label_12 . . . 770
12.54.3.30 label_13 . . . 771
12.54.3.31 label_14 . . . 771
12.54.3.32 label_15 . . . 771
12.54.3.33 label_2 . . . 771
12.54.3.34 label_25 . . . 771
12.54.3.35 label_3 . . . 771
12.54.3.36 label_4 . . . 771

12.54.3.37 label_5 . . . 771
12.54.3.38 label_6 . . . 771
12.54.3.39 label_7 . . . 772
12.54.3.40 label_8 . . . 772
12.54.3.41 llHeightBox . . . 772
12.54.3.42 line . . . 772
12.54.3.43 line_2 . . . 772
12.54.3.44 line_3 . . . 772
12.54.3.45 spinBox . . . 772
12.54.3.46 verticalLayout . . . 772
12.54.3.47 verticalSpacer . . . 772
12.54.3.48 verticalSpacer_2 . . . 773
12.54.3.49 verticalSpacer_3 . . . 773
12.54.3.50 xBox0 . . . 773
12.54.3.51 xBox1 . . . 773
12.54.3.52 xBox2 . . . 773
12.54.3.53 xBox3 . . . 773
12.54.3.54 xBox4 . . . 773
12.54.3.55 xBox5 . . . 773
12.54.3.56 xSlider0 . . . 773
12.54.3.57 xSlider1 . . . 774
12.54.3.58 xSlider2 . . . 774
12.54.3.59 xSlider3 . . . 774
12.54.3.60 xSlider4 . . . 774
12.54.3.61 xSlider5 . . . 774
12.54.3.62 xyBox . . . 774
12.54.3.63 yBox0 . . . 774
12.54.3.64 yBox1 . . . 774
12.54.3.65 yBox2 . . . 774
12.54.3.66 yBox3 . . . 775
12.54.3.67 yBox4 . . . 775
12.54.3.68 yBox5 . . . 775
12.54.3.69 ySlider0 . . . 775
12.54.3.70 ySlider1 . . . 775
12.54.3.71 ySlider2 . . . 775
12.54.3.72 ySlider3 . . . 775
12.54.3.73 ySlider4 . . . 775
12.54.3.74 ySlider5 . . . 775
12.55 MPerfWidget Class Reference . . . 776
12.55.1 Detailed Description . . . 778

12.55.2 Data Structure Documentation . . . 778
12.55.2.1 struct MPerfWidget::DpiUi . . . 778
12.55.3 Constructor & Destructor Documentation . . . 779
12.55.3.1 MPerfWidget . . . 779
12.55.3.2 ~MPerfWidget . . . 781
12.55.4 Member Function Documentation . . . 781
12.55.4.1 boxXChanged . . . 781
12.55.4.2 boxYChanged . . . 782
12.55.4.3 colorChanged . . . 783
12.55.4.4 colorClicked . . . 784
12.55.4.5 enableChanged . . . 784
12.55.4.6 on_aSnapBox_clicked . . . 785
12.55.4.7 on_copyButton_clicked . . . 785
12.55.4.8 on_indicBox_clicked . . . 786
12.55.4.9 on_lHeightBox_activated . . . 786
12.55.4.10 on_spinBox_valueChanged . . . 787
12.55.4.11 on_xyBox_clicked . . . 787
12.55.4.12 setPerf . . . 788
12.55.4.13 sliderXMoved . . . 789
12.55.4.14 sliderYMoved . . . 790
12.55.5 Field Documentation . . . 791
12.55.5.1 _xyLink . . . 791
12.55.5.2 boxXMapper . . . 791
12.55.5.3 boxYMapper . . . 791
12.55.5.4 buttonMapper1 . . . 791
12.55.5.5 buttonMapper2 . . . 792
12.55.5.6 colorLink . . . 792
12.55.5.7 DPI_COUNT . . . 792
12.55.5.8 DPI_MAX . . . 792
12.55.5.9 DPI_MIN . . . 792
12.55.5.10 enableMapper . . . 792
12.55.5.11 isSetting . . . 792
12.55.5.12 perf . . . 792
12.55.5.13 profile . . . 792
12.55.5.14 sliderXMapper . . . 792
12.55.5.15 sliderYMapper . . . 793
12.55.5.16 stages . . . 793
12.55.5.17 ui . . . 793
12.56 GradientDialog::Preset Struct Reference . . . 793
12.56.1 Detailed Description . . . 794

12.56.2 Constructor & Destructor Documentation . . . 794
12.56.2.1 Preset . . . 794
12.56.2.2 Preset . . . 794
12.56.2.3 Preset . . . 794
12.56.3 Field Documentation . . . 794
12.56.3.1 builtIn . . . 794
12.56.3.2 gradient . . . 794
12.56.3.3 name . . . 795
12.57 QDialog Class Reference . . . 795
12.58 QIODevice Class Reference . . . 796
12.59 QIODevice_descriptor Struct Reference . . . 797
12.59.1 Detailed Description . . . 797
12.59.2 Constructor & Destructor Documentation . . . 797
12.59.2.1 QIODevice_descriptor . . . 797
12.59.3 Field Documentation . . . 797
12.59.3.1 pos . . . 797
12.60 QListWidget Class Reference . . . 798
12.61 QMainWindow Class Reference . . . 799
12.62 QObject Class Reference . . . 800
12.63 QPushButton Class Reference . . . 801
12.64 QThread Class Reference . . . 802
12.65 QuaAdler32 Class Reference . . . 803
12.65.1 Detailed Description . . . 804
12.65.2 Constructor & Destructor Documentation . . . 805
12.65.2.1 QuaAdler32 . . . 805
12.65.3 Member Function Documentation . . . 805
12.65.3.1 calculate . . . 805
12.65.3.2 reset . . . 805
12.65.3.3 update . . . 806
12.65.3.4 value . . . 806
12.65.4 Field Documentation . . . 806
12.65.4.1 checksum . . . 806
12.66 QuaChecksum32 Class Reference . . . 806
12.66.1 Detailed Description . . . 808
12.66.2 Member Function Documentation . . . 808
12.66.2.1 calculate . . . 808
12.66.2.2 reset . . . 808
12.66.2.3 update . . . 808
12.66.2.4 value . . . 808
12.67 QuaCrc32 Class Reference . . . 809

12.67.1 Detailed Description . . . 810
12.67.2 Constructor & Destructor Documentation . . . 811
12.67.2.1 QuaCrc32 . . . 811
12.67.3 Member Function Documentation . . . 811
12.67.3.1 calculate . . . 811
12.67.3.2 reset . . . 811
12.67.3.3 update . . . 812
12.67.3.4 value . . . 812
12.67.4 Field Documentation . . . 812
12.67.4.1 checksum . . . 812
12.68 QuaGzipFile Class Reference . . . 812
12.68.1 Detailed Description . . . 815
12.68.2 Constructor & Destructor Documentation . . . 815
12.68.2.1 QuaGzipFile . . . 815
12.68.2.2 QuaGzipFile . . . 815
12.68.2.3 QuaGzipFile . . . 816
12.68.2.4 ~QuaGzipFile . . . 816
12.68.2.5 QuaGzipFile . . . 816
12.68.3 Member Function Documentation . . . 816
12.68.3.1 close . . . 816
12.68.3.2 flush . . . 817
12.68.3.3 getFileName . . . 817
12.68.3.4 isSequential . . . 817
12.68.3.5 open . . . 817
12.68.3.6 open . . . 818
12.68.3.7 operator= . . . 818
12.68.3.8 readData . . . 818
12.68.3.9 setFileName . . . 818
12.68.3.10 writeData . . . 819
12.68.4 Field Documentation . . . 819
12.68.4.1 d . . . 819
12.69 QuaZIODevice Class Reference . . . 819
12.69.1 Detailed Description . . . 822
12.69.2 Constructor & Destructor Documentation . . . 822
12.69.2.1 QuaZIODevice . . . 822
12.69.2.2 ~QuaZIODevice . . . 822
12.69.3 Member Function Documentation . . . 822
12.69.3.1 close . . . 822
12.69.3.2 flush . . . 823
12.69.3.3 getIoDevice . . . 824

12.69.3.4 isSequential . . . 824
12.69.3.5 open . . . 824
12.69.3.6 readData . . . 825
12.69.3.7 writeData . . . 826
12.69.4 Field Documentation . . . 826
12.69.4.1 d . . . 826
12.70 QuaZip Class Reference . . . 827
12.70.1 Detailed Description . . . 829
12.70.2 Member Enumeration Documentation . . . 830
12.70.2.1 CaseSensitivity . . . 830
12.70.2.2 Constants . . . 830
12.70.2.3 Mode . . . 830
12.70.3 Constructor & Destructor Documentation . . . 831
12.70.3.1 QuaZip . . . 831
12.70.3.2 QuaZip . . . 831
12.70.3.3 QuaZip . . . 831
12.70.3.4 QuaZip . . . 831
12.70.3.5 ~QuaZip . . . 831
12.70.4 Member Function Documentation . . . 832
12.70.4.1 close . . . 832
12.70.4.2 convertCaseSensitivity . . . 833
12.70.4.3 getComment . . . 834
12.70.4.4 getCommentCodec . . . 834
12.70.4.5 getCurrentFileInfo . . . 835
12.70.4.6 getCurrentFileInfo . . . 836
12.70.4.7 getCurrentFileName . . . 837
12.70.4.8 getEntriesCount . . . 838
12.70.4.9 getFileInfoList . . . 839
12.70.4.10 getFileInfoList64 . . . 840
12.70.4.11 getFileNameCodec . . . 840
12.70.4.12 getFileNameList . . . 841
12.70.4.13 getIoDevice . . . 841
12.70.4.14 getMode . . . 841
12.70.4.15 getUnzFile . . . 842
12.70.4.16 getZipError . . . 843
12.70.4.17 getZipFile . . . 844
12.70.4.18 getZipName . . . 844
12.70.4.19 goToFirstFile . . . 845
12.70.4.20 goToNextFile . . . 846
12.70.4.21 hasCurrentFile . . . 847

12.70.4.22 isAutoClose . . . 847
12.70.4.23 isDataDescriptorWritingEnabled . . . 847
12.70.4.24 isOpen . . . 848
12.70.4.25 isZip64Enabled . . . 848
12.70.4.26 open . . . 849
12.70.4.27 operator= . . . 852
12.70.4.28 setAutoClose . . . 852
12.70.4.29 setComment . . . 853
12.70.4.30 setCommentCodec . . . 853
12.70.4.31 setCommentCodec . . . 853
12.70.4.32 setCurrentFile . . . 854
12.70.4.33 setDataDescriptorWritingEnabled . . . 855
12.70.4.34 setDefaultFileNameCodec . . . 856
12.70.4.35 setDefaultFileNameCodec . . . 857
12.70.4.36 setFileNameCodec . . . 857
12.70.4.37 setFileNameCodec . . . 857
12.70.4.38 setIoDevice . . . 857
12.70.4.39 setZip64Enabled . . . 858
12.70.4.40 setZipName . . . 858
12.70.5 Friends And Related Function Documentation . . . 859
12.70.5.1 QuaZipPrivate . . . 859
12.70.6 Field Documentation . . . 859
12.70.6.1 p . . . 859
12.71 QuaZipDir Class Reference . . . 859
12.71.1 Detailed Description . . . 862
12.71.2 Constructor & Destructor Documentation . . . 862
12.71.2.1 QuaZipDir . . . 862
12.71.2.2 QuaZipDir . . . 862
12.71.2.3 ~QuaZipDir . . . 863
12.71.3 Member Function Documentation . . . 863
12.71.3.1 caseSensitivity . . . 863
12.71.3.2 cd . . . 863
12.71.3.3 cdUp . . . 865
12.71.3.4 count . . . 865
12.71.3.5 dirName . . . 866
12.71.3.6 entryInfoList . . . 866
12.71.3.7 entryInfoList . . . 867
12.71.3.8 entryInfoList64 . . . 867
12.71.3.9 entryInfoList64 . . . 868
12.71.3.10 entryList . . . 868

12.71.3.11 entryList . . . 869
12.71.3.12 exists . . . 869
12.71.3.13 exists . . . 870
12.71.3.14 filePath . . . 871
12.71.3.15 filter . . . 872
12.71.3.16 isRoot . . . 872
12.71.3.17 nameFilters . . . 872
12.71.3.18 operator!= . . . 873
12.71.3.19 operator= . . . 873
12.71.3.20 operator== . . . 873
12.71.3.21 operator[] . . . 873
12.71.3.22 path . . . 874
12.71.3.23 relativeFilePath . . . 874
12.71.3.24 setCaseSensitivity . . . 874
12.71.3.25 setFilter . . . 875
12.71.3.26 setNameFilters . . . 875
12.71.3.27 setPath . . . 875
12.71.3.28 setSorting . . . 876
12.71.3.29 sorting . . . 876
12.71.4 Field Documentation . . . 876
12.71.4.1 d . . . 876
12.72 QuaZipFile Class Reference . . . 877
12.72.1 Detailed Description . . . 879
12.72.2 Sequential or random-access? . . . 880
12.72.3 Constructor & Destructor Documentation . . . 880
12.72.3.1 QuaZipFile . . . 880
12.72.3.2 QuaZipFile . . . 880
12.72.3.3 QuaZipFile . . . 880
12.72.3.4 QuaZipFile . . . 880
12.72.3.5 QuaZipFile . . . 881
12.72.3.6 QuaZipFile . . . 881
12.72.3.7 ~QuaZipFile . . . 882
12.72.4 Member Function Documentation . . . 882
12.72.4.1 atEnd . . . 882
12.72.4.2 bytesAvailable . . . 883
12.72.4.3 close . . . 883
12.72.4.4 csize . . . 884
12.72.4.5 getActualFileName . . . 885
12.72.4.6 getCaseSensitivity . . . 886
12.72.4.7 getFileInfo . . . 886

12.72.4.8 getFileInfo . . . 887
12.72.4.9 getFileName . . . 888
12.72.4.10 getZip . . . 888
12.72.4.11 getZipError . . . 888
12.72.4.12 getZipName . . . 889
12.72.4.13 isRaw . . . 889
12.72.4.14 isSequential . . . 890
12.72.4.15 open . . . 890
12.72.4.16 open . . . 891
12.72.4.17 open . . . 891
12.72.4.18 open . . . 892
12.72.4.19 operator= . . . 894
12.72.4.20 pos . . . 894
12.72.4.21 readData . . . 895
12.72.4.22 setFileName . . . 896
12.72.4.23 setZip . . . 896
12.72.4.24 setZipName . . . 897
12.72.4.25 size . . . 897
12.72.4.26 usize . . . 898
12.72.4.27 writeData . . . 899
12.72.5 Friends And Related Function Documentation . . . 899
12.72.5.1 QuaZipFilePrivate . . . 900
12.72.6 Field Documentation . . . 900
12.72.6.1 p . . . 900
12.73 QuaZipFileInfo Struct Reference . . . 900
12.73.1 Detailed Description . . . 901
12.73.2 Member Function Documentation . . . 901
12.73.2.1 getPermissions . . . 901
12.73.3 Field Documentation . . . 902
12.73.3.1 comment . . . 902
12.73.3.2 compressedSize . . . 902
12.73.3.3 crc . . . 902
12.73.3.4 dateTime . . . 902
12.73.3.5 diskNumberStart . . . 902
12.73.3.6 externalAttr . . . 902
12.73.3.7 extra . . . 902
12.73.3.8 flags . . . 903
12.73.3.9 internalAttr . . . 903
12.73.3.10 method . . . 903
12.73.3.11 name . . . 903

12.73.3.12 uncompressedSize . . . 903
12.73.3.13 versionCreated . . . 903
12.73.3.14 versionNeeded . . . 903
12.74 QuaZipFileInfo64 Struct Reference . . . 903
12.74.1 Detailed Description . . . 905
12.74.2 Member Function Documentation . . . 905
12.74.2.1 getNTFSaTime . . . 905
12.74.2.2 getNTFScTime . . . 906
12.74.2.3 getNTFSmTime . . . 906
12.74.2.4 getPermissions . . . 907
12.74.2.5 isEncrypted . . . 908
12.74.2.6 toQuaZipFileInfo . . . 908
12.74.3 Field Documentation . . . 909
12.74.3.1 comment . . . 909
12.74.3.2 compressedSize . . . 909
12.74.3.3 crc . . . 909
12.74.3.4 dateTime . . . 909
12.74.3.5 diskNumberStart . . . 909
12.74.3.6 externalAttr . . . 910
12.74.3.7 extra . . . 910
12.74.3.8 flags . . . 910
12.74.3.9 internalAttr . . . 910
12.74.3.10 method . . . 910
12.74.3.11 name . . . 910
12.74.3.12 uncompressedSize . . . 910
12.74.3.13 versionCreated . . . 910
12.74.3.14 versionNeeded . . . 910
12.75 QuaZipFilePrivate Class Reference . . . 911
12.75.1 Detailed Description . . . 912
12.75.2 Constructor & Destructor Documentation . . . 912
12.75.2.1 QuaZipFilePrivate . . . 912
12.75.2.2 QuaZipFilePrivate . . . 912
12.75.2.3 QuaZipFilePrivate . . . 912
12.75.2.4 QuaZipFilePrivate . . . 913
12.75.2.5 ~QuaZipFilePrivate . . . 913
12.75.3 Member Function Documentation . . . 913
12.75.3.1 resetZipError . . . 913
12.75.3.2 setZipError . . . 913
12.75.4 Friends And Related Function Documentation . . . 914
12.75.4.1 QuaZipFile . . . 914

12.75.5 Field Documentation . . . 914
12.75.5.1 caseSensitivity . . . 914
12.75.5.2 crc . . . 914
12.75.5.3 fileName . . . 914
12.75.5.4 internal . . . 915
12.75.5.5 q . . . 915
12.75.5.6 raw . . . 915
12.75.5.7 uncompressedSize . . . 915
12.75.5.8 writePos . . . 915
12.75.5.9 zip . . . 915
12.75.5.10 zipError . . . 915
12.76 QuaZipNewInfo Struct Reference . . . 916
12.76.1 Detailed Description . . . 917
12.76.2 Constructor & Destructor Documentation . . . 917
12.76.2.1 QuaZipNewInfo . . . 917
12.76.2.2 QuaZipNewInfo . . . 917
12.76.2.3 QuaZipNewInfo . . . 918
12.76.2.4 QuaZipNewInfo . . . 919
12.76.3 Member Function Documentation . . . 919
12.76.3.1 setFileDateTime . . . 919
12.76.3.2 setFileNTFSaTime . . . 919
12.76.3.3 setFileNTFScTime . . . 920
12.76.3.4 setFileNTFSmTime . . . 921
12.76.3.5 setFileNTFSTimes . . . 922
12.76.3.6 setFilePermissions . . . 922
12.76.3.7 setPermissions . . . 923
12.76.4 Field Documentation . . . 923
12.76.4.1 comment . . . 923
12.76.4.2 dateTime . . . 923
12.76.4.3 externalAttr . . . 924
12.76.4.4 extraGlobal . . . 924
12.76.4.5 extraLocal . . . 924
12.76.4.6 internalAttr . . . 924
12.76.4.7 name . . . 924
12.76.4.8 uncompressedSize . . . 924
12.77 QuaZipPrivate Class Reference . . . 924
12.77.1 Detailed Description . . . 926
12.77.2 Constructor & Destructor Documentation . . . 926
12.77.2.1 QuaZipPrivate . . . 926
12.77.2.2 QuaZipPrivate . . . 927

12.77.2.3 QuaZipPrivate . . . 927
12.77.3 Member Function Documentation . . . 927
12.77.3.1 addCurrentFileToDirectoryMap . . . 927
12.77.3.2 clearDirectoryMap . . . 928
12.77.3.3 getDefaultFileNameCodec . . . 929
12.77.3.4 getFileInfoList . . . 929
12.77.3.5 goToFirstUnmappedFile . . . 930
12.77.4 Friends And Related Function Documentation . . . 931
12.77.4.1 QuaZip . . . 931
12.77.5 Field Documentation . . . 931
12.77.5.1 "@4 . . . 931
12.77.5.2 autoClose . . . 931
12.77.5.3 comment . . . 931
12.77.5.4 commentCodec . . . 931
12.77.5.5 dataDescriptorWritingEnabled . . . 932
12.77.5.6 defaultFileNameCodec . . . 932
12.77.5.7 directoryCaseInsensitive . . . 932
12.77.5.8 directoryCaseSensitive . . . 932
12.77.5.9 fileNameCodec . . . 932
12.77.5.10 hasCurrentFile_f . . . 932
12.77.5.11 ioDevice . . . 932
12.77.5.12 lastMappedDirectoryEntry . . . 932
12.77.5.13 mode . . . 932
12.77.5.14 q . . . 933
12.77.5.15 zip64 . . . 933
12.77.5.16 zipError . . . 933
12.77.5.17 zipName . . . 933
12.78 QWidget Class Reference . . . 933
12.79 Ui::RebindWidget Class Reference . . . 934
12.79.1 Detailed Description . . . 937
12.79.2 Member Function Documentation . . . 937
12.79.2.1 retranslateUi . . . 937
12.79.2.2 setupUi . . . 940
12.79.3 Field Documentation . . . 950
12.79.3.1 actionTab . . . 950
12.79.3.2 animBox . . . 950
12.79.3.3 animButton . . . 950
12.79.3.4 animKrBox . . . 951
12.79.3.5 animOnceBox . . . 951
12.79.3.6 animTab . . . 951

12.79.3.7 applyButton . . . 951
12.79.3.8 btnClearMacro . . . 951
12.79.3.9 btnStartMacro . . . 951
12.79.3.10 btnStopMacro . . . 951
12.79.3.11 cancelButton . . . 951
12.79.3.12 delayButtons . . . 952
12.79.3.13 dpiBox . . . 952
12.79.3.14 dpiButton . . . 952
12.79.3.15 dpiCustLabel . . . 952
12.79.3.16 dpiCustXBox . . . 952
12.79.3.17 dpiCustYBox . . . 952
12.79.3.18 dpiWarning . . . 952
12.79.3.19 fnBox . . . 952
12.79.3.20 fnButton . . . 953
12.79.3.21 gridLayout . . . 953
12.79.3.22 gridLayout_10 . . . 953
12.79.3.23 gridLayout_2 . . . 953
12.79.3.24 gridLayout_3 . . . 953
12.79.3.25 gridLayout_4 . . . 953
12.79.3.26 gridLayout_5 . . . 953
12.79.3.27 gridLayout_8 . . . 953
12.79.3.28 groupBox_2 . . . 953
12.79.3.29 groupBox_3 . . . 954
12.79.3.30 groupBox_4 . . . 954
12.79.3.31 horizontalLayout . . . 954
12.79.3.32 horizontalLayout_2 . . . 954
12.79.3.33 horizontalLayout_3 . . . 954
12.79.3.34 horizontalSpacer . . . 954
12.79.3.35 horizontalSpacer_10 . . . 954
12.79.3.36 horizontalSpacer_11 . . . 954
12.79.3.37 horizontalSpacer_12 . . . 954
12.79.3.38 horizontalSpacer_13 . . . 955
12.79.3.39 horizontalSpacer_14 . . . 955
12.79.3.40 horizontalSpacer_15 . . . 955
12.79.3.41 horizontalSpacer_16 . . . 955
12.79.3.42 horizontalSpacer_17 . . . 955
12.79.3.43 horizontalSpacer_18 . . . 955
12.79.3.44 horizontalSpacer_19 . . . 955
12.79.3.45 horizontalSpacer_2 . . . 955
12.79.3.46 horizontalSpacer_20 . . . 955

12.79.3.47 horizontalSpacer_21 . . . 956
12.79.3.48 horizontalSpacer_22 . . . 956
12.79.3.49 horizontalSpacer_23 . . . 956
12.79.3.50 horizontalSpacer_24 . . . 956
12.79.3.51 horizontalSpacer_25 . . . 956
12.79.3.52 horizontalSpacer_26 . . . 956
12.79.3.53 horizontalSpacer_3 . . . 956
12.79.3.54 horizontalSpacer_4 . . . 956
12.79.3.55 horizontalSpacer_5 . . . 956
12.79.3.56 horizontalSpacer_6 . . . 957
12.79.3.57 horizontalSpacer_7 . . . 957
12.79.3.58 horizontalSpacer_8 . . . 957
12.79.3.59 horizontalSpacer_9 . . . 957
12.79.3.60 keyTab . . . 957
12.79.3.61 label . . . 957
12.79.3.62 layoutWidget . . . 957
12.79.3.63 layoutWidget1 . . . 957
12.79.3.64 layoutWidget2 . . . 957
12.79.3.65 layoutWidget_2 . . . 958
12.79.3.66 lbl_macro . . . 958
12.79.3.67 lightBox . . . 958
12.79.3.68 lightButton . . . 958
12.79.3.69 lightWrapBox . . . 958
12.79.3.70 lockBox . . . 958
12.79.3.71 lockButton . . . 958
12.79.3.72 macroTab . . . 958
12.79.3.73 mb2Box . . . 959
12.79.3.74 mb2Button . . . 959
12.79.3.75 mbBox . . . 959
12.79.3.76 mbButton . . . 959
12.79.3.77 mediaBox . . . 959
12.79.3.78 mediaButton . . . 959
12.79.3.79 modBox . . . 959
12.79.3.80 modButton . . . 959
12.79.3.81 modeBox . . . 960
12.79.3.82 modeButton . . . 960
12.79.3.83 modeWrapBox . . . 960
12.79.3.84 mouseTab . . . 960
12.79.3.85 numBox . . . 960
12.79.3.86 numButton . . . 960

12.79.3.87 programKpBox . . . 960
12.79.3.88 programKpButton . . . 961
12.79.3.89 programKpExtra . . . 961
12.79.3.90 programKpModeBox . . . 961
12.79.3.91 programKpSIBox . . . 961
12.79.3.92 programKrBox . . . 961
12.79.3.93 programKrButton . . . 961
12.79.3.94 programKrExtra . . . 961
12.79.3.95 programKrModeBox . . . 961
12.79.3.96 programKrSIBox . . . 962
12.79.3.97 programTab . . . 962
12.79.3.98 progTipLabel . . . 962
12.79.3.99 pteMacroBox . . . 962
12.79.3.100 pteMacroComment . . . 962
12.79.3.101 pteMacroText . . . 962
12.79.3.102 rb_delay_asTyped . . . 962
12.79.3.103 rb_delay_default . . . 962
12.79.3.104 rb_delay_no . . . 963
12.79.3.105 resetButton . . . 963
12.79.3.106 tabWidget . . . 963
12.79.3.107 txtBuffer . . . 963
12.79.3.108 typingBox . . . 963
12.79.3.109 typingButton . . . 963
12.79.3.110 unbindButton . . . 963
12.79.3.111 verticalLayout . . . 963
12.79.3.112 verticalLayout_2 . . . 964
12.79.3.113 verticalSpacer . . . 964
12.79.3.114 verticalSpacer_10 . . . 964
12.79.3.115 verticalSpacer_11 . . . 964
12.79.3.116 verticalSpacer_2 . . . 964
12.79.3.117 verticalSpacer_3 . . . 964
12.79.3.118 verticalSpacer_4 . . . 964
12.79.3.119 verticalSpacer_5 . . . 964
12.79.3.120 verticalSpacer_6 . . . 964
12.79.3.121 verticalSpacer_7 . . . 965
12.79.3.122 verticalSpacer_8 . . . 965
12.79.3.123 wheelBox . . . 965
12.79.3.124 wheelButton . . . 965
12.80 RebindWidget Class Reference . . . 965
12.80.1 Detailed Description . . . 969

12.80.2 Constructor & Destructor Documentation . . . 969
12.80.2.1 RebindWidget . . . 969
12.80.2.2 ~RebindWidget . . . 970
12.80.3 Member Function Documentation . . . 970
12.80.3.1 applyChanges . . . 970
12.80.3.2 convertMacroBox . . . 972
12.80.3.3 helpStatus . . . 973
12.80.3.4 on_animBox_currentIndexChanged . . . 974
12.80.3.5 on_animButton_clicked . . . 975
12.80.3.6 on_applyButton_clicked . . . 975
12.80.3.7 on_btnClearMacro_clicked . . . 975
12.80.3.8 on_btnStartMacro_clicked . . . 976
12.80.3.9 on_btnStopMacro_clicked . . . 976
12.80.3.10 on_cancelButton_clicked . . . 977
12.80.3.11 on_dpiBox_currentIndexChanged . . . 978
12.80.3.12 on_dpiButton_clicked . . . 979
12.80.3.13 on_fnBox_currentIndexChanged . . . 979
12.80.3.14 on_fnButton_clicked . . . 979
12.80.3.15 on_lightBox_currentIndexChanged . . . 980
12.80.3.16 on_lightButton_clicked . . . 980
12.80.3.17 on_lockBox_currentIndexChanged . . . 980
12.80.3.18 on_lockButton_clicked . . . 981
12.80.3.19 on_mb2Box_currentIndexChanged . . . 981
12.80.3.20 on_mb2Button_clicked . . . 981
12.80.3.21 on_mbBox_currentIndexChanged . . . 981
12.80.3.22 on_mbButton_clicked . . . 982
12.80.3.23 on_mediaBox_currentIndexChanged . . . 982
12.80.3.24 on_mediaButton_clicked . . . 982
12.80.3.25 on_modBox_currentIndexChanged . . . 983
12.80.3.26 on_modButton_clicked . . . 983
12.80.3.27 on_modeBox_currentIndexChanged . . . 983
12.80.3.28 on_modeButton_clicked . . . 984
12.80.3.29 on_numBox_currentIndexChanged . . . 984
12.80.3.30 on_numButton_clicked . . . 984
12.80.3.31 on_programKpBox_textChanged . . . 984
12.80.3.32 on_programKpButton_clicked . . . 985
12.80.3.33 on_programKpSlBox_clicked . . . 985
12.80.3.34 on_programKrBox_textChanged . . . 985
12.80.3.35 on_programKrButton_clicked . . . 986
12.80.3.36 on_programKrSlBox_clicked . . . 986

12.80.3.37 on_pteMacroBox_textChanged . . . 986
12.80.3.38 on_rb_delay_asTyped_toggled . . . 987
12.80.3.39 on_rb_delay_default_toggled . . . 987
12.80.3.40 on_rb_delay_no_toggled . . . 988
12.80.3.41 on_resetButton_clicked . . . 988
12.80.3.42 on_typingBox_currentIndexChanged . . . 989
12.80.3.43 on_typingButton_clicked . . . 989
12.80.3.44 on_unbindButton_clicked . . . 989
12.80.3.45 on_wheelBox_currentIndexChanged . . . 990
12.80.3.46 on_wheelButton_clicked . . . 990
12.80.3.47 setBind . . . 990
12.80.3.48 setBox . . . 993
12.80.3.49 setCorrectRadioButton . . . 994
12.80.3.50 setSelection . . . 995
12.80.4 Field Documentation . . . 998
12.80.4.1 bind . . . 999
12.80.4.2 fnKeys . . . 999
12.80.4.3 macReader . . . 999
12.80.4.4 mediaKeys . . . 999
12.80.4.5 modKeys . . . 999
12.80.4.6 mouseExtKeys . . . 999
12.80.4.7 mouseKeys . . . 999
12.80.4.8 numKeys . . . 999
12.80.4.9 profile . . . 999
12.80.4.10 selection . . . 1000
12.80.4.11 TAB_ANIM . . . 1000
12.80.4.12 TAB_KB . . . 1000
12.80.4.13 TAB_MACRO . . . 1000
12.80.4.14 TAB_MOUSE . . . 1000
12.80.4.15 TAB_PROGRAM . . . 1000
12.80.4.16 TAB_SPECIAL . . . 1000
12.80.4.17 typingKeys . . . 1000
12.80.4.18 ui . . . 1000
12.80.4.19 wheelKeys . . . 1001
12.81 RListWidget Class Reference . . . 1001
12.81.1 Detailed Description . . . 1002
12.81.2 Constructor & Destructor Documentation . . . 1003
12.81.2.1 RListWidget . . . 1003
12.81.3 Member Function Documentation . . . 1003
12.81.3.1 change . . . 1003

12.81.3.2 enter . . . 1004
12.81.3.3 orderChanged . . . 1004
12.81.3.4 rescanItems . . . 1005
12.81.3.5 timerTick . . . 1005
12.81.4 Field Documentation . . . 1006
12.81.4.1 currentData . . . 1006
12.81.4.2 DATA_ROLE . . . 1006
12.81.4.3 dragged . . . 1006
12.81.4.4 previousItems . . . 1006
12.81.4.5 reorderTimer . . . 1006
12.82 SettingsWidget Class Reference . . . 1006
12.82.1 Detailed Description . . . 1009
12.82.2 Constructor & Destructor Documentation . . . 1009
12.82.2.1 SettingsWidget . . . 1009
12.82.2.2 ~SettingsWidget . . . 1010
12.82.3 Member Function Documentation . . . 1010
12.82.3.1 on_altBox_activated . . . 1011
12.82.3.2 on_autoFWBox_clicked . . . 1011
12.82.3.3 on_capsBox_activated . . . 1011
12.82.3.4 on_ctrlBox_activated . . . 1012
12.82.3.5 on_extraButton_clicked . . . 1012
12.82.3.6 on_layoutBox_activated . . . 1012
12.82.3.7 on_loginItemBox_clicked . . . 1013
12.82.3.8 on_pushButton_clicked . . . 1013
12.82.3.9 on_shiftBox_activated . . . 1013
12.82.3.10 on_winBox_activated . . . 1014
12.82.3.11 pollUpdates . . . 1014
12.82.3.12 setStatus . . . 1014
12.82.3.13 setVersion . . . 1015
12.82.3.14 showLayoutDialog . . . 1015
12.82.3.15 updateModifiers . . . 1016
12.82.4 Friends And Related Function Documentation . . . 1017
12.82.4.1 MainWindow . . . 1017
12.82.5 Field Documentation . . . 1017
12.82.5.1 extra . . . 1017
12.82.5.2 ui . . . 1017
12.83 Ui::SettingsWidget Class Reference . . . 1018
12.83.1 Detailed Description . . . 1020
12.83.2 Member Function Documentation . . . 1020
12.83.2.1 retranslateUi . . . 1020

12.83.2.2 setupUi . . . 1021
12.83.3 Field Documentation . . . 1025
12.83.3.1 altBox . . . 1025
12.83.3.2 altLabel . . . 1025
12.83.3.3 autoFWBox . . . 1025
12.83.3.4 capsBox . . . 1025
12.83.3.5 capsLabel . . . 1026
12.83.3.6 ctrlBox . . . 1026
12.83.3.7 ctrlLabel . . . 1026
12.83.3.8 devicesLabel . . . 1026
12.83.3.9 extraButton . . . 1026
12.83.3.10 gridLayout . . . 1026
12.83.3.11 gridLayout_3 . . . 1026
12.83.3.12 gridLayout_5 . . . 1026
12.83.3.13 horizontalLayout . . . 1026
12.83.3.14 horizontalLayout_2 . . . 1027
12.83.3.15 horizontalSpacer . . . 1027
12.83.3.16 horizontalSpacer_10 . . . 1027
12.83.3.17 horizontalSpacer_2 . . . 1027
12.83.3.18 horizontalSpacer_7 . . . 1027
12.83.3.19 horizontalSpacer_9 . . . 1027
12.83.3.20 label . . . 1027
12.83.3.21 label_2 . . . 1027
12.83.3.22 label_3 . . . 1027
12.83.3.23 label_5 . . . 1028
12.83.3.24 label_6 . . . 1028
12.83.3.25 label_8 . . . 1028
12.83.3.26 layoutBox . . . 1028
12.83.3.27 line . . . 1028
12.83.3.28 line_2 . . . 1028
12.83.3.29 line_3 . . . 1028
12.83.3.30 loginItemBox . . . 1028
12.83.3.31 pushButton . . . 1028
12.83.3.32 shiftBox . . . 1029
12.83.3.33 shiftLabel . . . 1029
12.83.3.34 versionLabel . . . 1029
12.83.3.35 verticalSpacer . . . 1029
12.83.3.36 verticalSpacer_10 . . . 1029
12.83.3.37 verticalSpacer_2 . . . 1029
12.83.3.38 verticalSpacer_3 . . . 1029

12.83.3.39 verticalSpacer_4 . . . 1029
12.83.3.40 verticalSpacer_5 . . . 1029
12.83.3.41 verticalSpacer_6 . . . 1030
12.83.3.42 verticalSpacer_7 . . . 1030
12.83.3.43 verticalSpacer_8 . . . 1030
12.83.3.44 winBox . . . 1030
12.83.3.45 winLabel . . . 1030
12.84 SGroup Class Reference . . . 1030
12.84.1 Detailed Description . . . 1032
12.84.2 Constructor & Destructor Documentation . . . 1032
12.84.2.1 SGroup . . . 1032
12.84.2.2 ~SGroup . . . 1032
12.84.3 Field Documentation . . . 1032
12.84.3.1 _settings . . . 1032
12.85 Ui_AnimAddDialog Class Reference . . . 1033
12.85.1 Detailed Description . . . 1034
12.85.2 Member Function Documentation . . . 1034
12.85.2.1 retranslateUi . . . 1034
12.85.2.2 setupUi . . . 1034
12.85.3 Field Documentation . . . 1036
12.85.3.1 animBox . . . 1036
12.85.3.2 buttonBox . . . 1036
12.85.3.3 gridLayout . . . 1036
12.85.3.4 horizontalLayout . . . 1036
12.85.3.5 horizontalSpacer . . . 1036
12.85.3.6 horizontalSpacer_2 . . . 1037
12.85.3.7 label . . . 1037
12.85.3.8 label_2 . . . 1037
12.85.3.9 presetBox . . . 1037
12.85.3.10 previewBox . . . 1037
12.85.3.11 verticalSpacer_2 . . . 1037
12.85.3.12 verticalSpacer_3 . . . 1037
12.86 Ui_AnimSettingDialog Class Reference . . . 1037
12.86.1 Detailed Description . . . 1040
12.86.2 Member Function Documentation . . . 1040
12.86.2.1 retranslateUi . . . 1040
12.86.2.2 setupUi . . . 1041
12.86.3 Field Documentation . . . 1044
12.86.3.1 animName . . . 1044
12.86.3.2 buttonBox . . . 1045

12.86.3.3 delayBox . . . 1045
12.86.3.4 gridLayout . . . 1045
12.86.3.5 gridLayout_2 . . . 1045
12.86.3.6 horizontalSpacer . . . 1045
12.86.3.7 horizontalSpacer_2 . . . 1045
12.86.3.8 horizontalSpacer_3 . . . 1045
12.86.3.9 horizontalSpacer_4 . . . 1045
12.86.3.10 horizontalSpacer_5 . . . 1045
12.86.3.11 horizontalSpacer_6 . . . 1046
12.86.3.12 infoTab . . . 1046
12.86.3.13 kpDelayBox . . . 1046
12.86.3.14 kpModeStopBox . . . 1046
12.86.3.15 kpReleaseBox . . . 1046
12.86.3.16 kpRepeatBox . . . 1046
12.86.3.17 kpRepeatLabel . . . 1046
12.86.3.18 kpRepeatLabel_2 . . . 1046
12.86.3.19 label . . . 1046
12.86.3.20 label_10 . . . 1047
12.86.3.21 label_11 . . . 1047
12.86.3.22 label_12 . . . 1047
12.86.3.23 label_13 . . . 1047
12.86.3.24 label_2 . . . 1047
12.86.3.25 label_3 . . . 1047
12.86.3.26 label_4 . . . 1047
12.86.3.27 label_5 . . . 1047
12.86.3.28 label_6 . . . 1047
12.86.3.29 label_7 . . . 1048
12.86.3.30 label_8 . . . 1048
12.86.3.31 label_9 . . . 1048
12.86.3.32 line . . . 1048
12.86.3.33 line_2 . . . 1048
12.86.3.34 repeatBox . . . 1048
12.86.3.35 repeatLabel . . . 1048
12.86.3.36 repeatLabel_2 . . . 1048
12.86.3.37 sAuthorLabel . . . 1048
12.86.3.38 sDescLabel . . . 1049
12.86.3.39 settingsGrid . . . 1049
12.86.3.40 settingTab . . . 1049
12.86.3.41 sLicenseLabel . . . 1049
12.86.3.42 sNameLabel . . . 1049

12.86.3.43 sVerLabel . . . 1049
12.86.3.44 sYearLabel . . . 1049
12.86.3.45 tabWidget . . . 1049
12.86.3.46 timeGrid . . . 1049
12.86.3.47 timeTab . . . 1050
12.86.3.48 verticalSpacer . . . 1050
12.86.3.49 verticalSpacer_2 . . . 1050
12.86.3.50 verticalSpacer_3 . . . 1050
12.87 Ui_ExtraSettingsWidget Class Reference . . . 1050
12.87.1 Detailed Description . . . 1052
12.87.2 Member Function Documentation . . . 1053
12.87.2.1 retranslateUi . . . 1053
12.87.2.2 setupUi . . . 1054
12.87.3 Field Documentation . . . 1057
12.87.3.1 animCountLabel . . . 1057
12.87.3.2 animPathLabel . . . 1058
12.87.3.3 animScanButton . . . 1058
12.87.3.4 brightnessBox . . . 1058
12.87.3.5 buttonBox . . . 1058
12.87.3.6 delayBox . . . 1058
12.87.3.7 ditherBox . . . 1058
12.87.3.8 fpsBox . . . 1058
12.87.3.9 fpsWarnLabel . . . 1058
12.87.3.10 gridLayout . . . 1058
12.87.3.11 horizontalLayout . . . 1059
12.87.3.12 horizontalLayout_2 . . . 1059
12.87.3.13 horizontalSpacer . . . 1059
12.87.3.14 horizontalSpacer_2 . . . 1059
12.87.3.15 horizontalSpacer_3 . . . 1059
12.87.3.16 label . . . 1059
12.87.3.17 label_10 . . . 1059
12.87.3.18 label_11 . . . 1059
12.87.3.19 label_2 . . . 1059
12.87.3.20 label_3 . . . 1060
12.87.3.21 label_5 . . . 1060
12.87.3.22 label_6 . . . 1060
12.87.3.23 line . . . 1060
12.87.3.24 line_2 . . . 1060
12.87.3.25 line_3 . . . 1060
12.87.3.26 mAccelBox . . . 1060

12.87.3.27 osxLabel . . . 1060
12.87.3.28 osxLine . . . 1060
12.87.3.29 sAccelBox . . . 1061
12.87.3.30 sSpeedBox . . . 1061
12.87.3.31 sSpeedWidget . . . 1061
12.87.3.32 trayBox . . . 1061
12.87.3.33 verticalSpacer . . . 1061
12.87.3.34 verticalSpacer_10 . . . 1061
12.87.3.35 verticalSpacer_11 . . . 1061
12.87.3.36 verticalSpacer_2 . . . 1061
12.87.3.37 verticalSpacer_3 . . . 1061
12.87.3.38 verticalSpacer_4 . . . 1062
12.87.3.39 verticalSpacer_5 . . . 1062
12.87.3.40 verticalSpacer_6 . . . 1062
12.87.3.41 verticalSpacer_7 . . . 1062
12.87.3.42 verticalSpacer_8 . . . 1062
12.87.3.43 verticalSpacer_9 . . . 1062
12.88 Ui_FwUpgradeDialog Class Reference . . . 1062
12.88.1 Detailed Description . . . 1064
12.88.2 Member Function Documentation . . . 1064
12.88.2.1 retranslateUi . . . 1064
12.88.2.2 setupUi . . . 1065
12.88.3 Field Documentation . . . 1066
12.88.3.1 actionButton . . . 1067
12.88.3.2 cancelButton . . . 1067
12.88.3.3 curLabel . . . 1067
12.88.3.4 devLabel . . . 1067
12.88.3.5 gridLayout . . . 1067
12.88.3.6 gridLayout_2 . . . 1067
12.88.3.7 horizontalSpacer . . . 1067
12.88.3.8 horizontalSpacer_2 . . . 1067
12.88.3.9 horizontalSpacer_3 . . . 1067
12.88.3.10 horizontalSpacer_4 . . . 1068
12.88.3.11 label_2 . . . 1068
12.88.3.12 label_4 . . . 1068
12.88.3.13 newLabel . . . 1068
12.88.3.14 progressBar . . . 1068
12.88.3.15 statusLabel . . . 1068
12.89 Ui_GradientDialog Class Reference . . . 1068
12.89.1 Detailed Description . . . 1070

12.89.2 Member Function Documentation . . . 1070
12.89.2.1 retranslateUi . . . 1070
12.89.2.2 setupUi . . . 1071
12.89.3 Field Documentation . . . 1073
12.89.3.1 buttonBox . . . 1073
12.89.3.2 gridLayout . . . 1073
12.89.3.3 gridLayout_2 . . . 1073
12.89.3.4 gridLayout_3 . . . 1074
12.89.3.5 groupBox_2 . . . 1074
12.89.3.6 horizontalLayout . . . 1074
12.89.3.7 horizontalLayout_2 . . . 1074
12.89.3.8 horizontalSpacer . . . 1074
12.89.3.9 horizontalSpacer_2 . . . 1074
12.89.3.10 horizontalSpacer_3 . . . 1074
12.89.3.11 label . . . 1074
12.89.3.12 label_2 . . . 1074
12.89.3.13 presetDelete . . . 1075
12.89.3.14 presetList . . . 1075
12.89.3.15 presetName . . . 1075
12.89.3.16 presetSave . . . 1075
12.89.3.17 stopBox . . . 1075
12.89.3.18 stopColor . . . 1075
12.89.3.19 stopOpacity . . . 1075
12.89.3.20 stopPos . . . 1075
12.89.3.21 verticalSpacer . . . 1075
12.89.3.22 verticalSpacer_2 . . . 1076
12.89.3.23 widget . . . 1076
12.90 Ui_KbAnimWidget Class Reference . . . 1076
12.90.1 Detailed Description . . . 1078
12.90.2 Member Function Documentation . . . 1078
12.90.2.1 retranslateUi . . . 1078
12.90.2.2 setupUi . . . 1078
12.90.3 Field Documentation . . . 1081
12.90.3.1 aCopyLabel . . . 1081
12.90.3.2 aMissingLabel . . . 1081
12.90.3.3 aMissingPage . . . 1081
12.90.3.4 aNameLabel . . . 1081
12.90.3.5 animList . . . 1081
12.90.3.6 animPage . . . 1082
12.90.3.7 aVerLabel . . . 1082

12.90.3.8 blendBox . . . 1082
12.90.3.9 deleteButton . . . 1082
12.90.3.10 gridLayout . . . 1082
12.90.3.11 gridLayout_2 . . . 1082
12.90.3.12 gridLayout_3 . . . 1082
12.90.3.13 gridLayout_4 . . . 1082
12.90.3.14 horizontalLayout . . . 1083
12.90.3.15 horizontalSpacer . . . 1083
12.90.3.16 horizontalSpacer_2 . . . 1083
12.90.3.17 horizontalSpacer_3 . . . 1083
12.90.3.18 horizontalSpacer_4 . . . 1083
12.90.3.19 horizontalSpacer_5 . . . 1083
12.90.3.20 keyButton . . . 1083
12.90.3.21 label_5 . . . 1083
12.90.3.22 label_6 . . . 1083
12.90.3.23 label_7 . . . 1084
12.90.3.24 nameBox . . . 1084
12.90.3.25 noAnimLabel . . . 1084
12.90.3.26 noSelPage . . . 1084
12.90.3.27 opacityBox . . . 1084
12.90.3.28 propertyButton . . . 1084
12.90.3.29 selectionStack . . . 1084
12.90.3.30 verticalLayout . . . 1084
12.90.3.31 verticalSpacer . . . 1084
12.90.3.32 verticalSpacer_2 . . . 1085
12.90.3.33 verticalSpacer_3 . . . 1085
12.91 Ui_KbBindWidget Class Reference . . . 1085
12.91.1 Detailed Description . . . 1086
12.91.2 Member Function Documentation . . . 1087
12.91.2.1 retranslateUi . . . 1087
12.91.2.2 setupUi . . . 1087
12.91.3 Field Documentation . . . 1089
12.91.3.1 copyButton . . . 1089
12.91.3.2 gridLayout . . . 1089
12.91.3.3 gridLayout_2 . . . 1089
12.91.3.4 horizontalLayout_2 . . . 1090
12.91.3.5 horizontalSpacer . . . 1090
12.91.3.6 horizontalSpacer_2 . . . 1090
12.91.3.7 keyWidget . . . 1090
12.91.3.8 label . . . 1090

12.91.3.9 label_2 . . . 1090
12.91.3.10 line . . . 1090
12.91.3.11 line_2 . . . 1090
12.91.3.12 rbWidget . . . 1090
12.91.3.13 resetButton . . . 1091
12.91.3.14 selectLabel . . . 1091
12.91.3.15 verticalLayout . . . 1091
12.91.3.16 verticalSpacer . . . 1091
12.91.3.17 widget . . . 1091
12.92 Ui_KbLightWidget Class Reference . . . 1091
12.92.1 Detailed Description . . . 1093
12.92.2 Member Function Documentation . . . 1093
12.92.2.1 retranslateUi . . . 1093
12.92.2.2 setupUi . . . 1094
12.92.3 Field Documentation . . . 1096
12.92.3.1 animButton . . . 1097
12.92.3.2 animWidget . . . 1097
12.92.3.3 bgButton . . . 1097
12.92.3.4 brightnessBox . . . 1097
12.92.3.5 gridLayout . . . 1097
12.92.3.6 gridLayout_2 . . . 1097
12.92.3.7 gridLayout_4 . . . 1097
12.92.3.8 horizontalLayout . . . 1097
12.92.3.9 horizontalLayout_2 . . . 1097
12.92.3.10 horizontalLayout_3 . . . 1098
12.92.3.11 horizontalSpacer_3 . . . 1098
12.92.3.12 horizontalSpacer_4 . . . 1098
12.92.3.13 horizontalSpacer_5 . . . 1098
12.92.3.14 horizontalSpacer_6 . . . 1098
12.92.3.15 keyWidget . . . 1098
12.92.3.16 label . . . 1098
12.92.3.17 label_2 . . . 1098
12.92.3.18 label_3 . . . 1098
12.92.3.19 line . . . 1099
12.92.3.20 line_2 . . . 1099
12.92.3.21 selLabel . . . 1099
12.92.3.22 showAnimBox . . . 1099
12.92.3.23 verticalLayout . . . 1099
12.92.3.24 verticalSpacer_2 . . . 1099
12.92.3.25 verticalSpacer_3 . . . 1099

12.92.3.26 widget . . . 1099
12.93 Ui_KbProfileDialog Class Reference . . . 1099
12.93.1 Detailed Description . . . 1101
12.93.2 Member Function Documentation . . . 1101
12.93.2.1 retranslateUi . . . 1101
12.93.2.2 setupUi . . . 1101
12.93.3 Field Documentation . . . 1102
12.93.3.1 buttonBox . . . 1102
12.93.3.2 gridLayout . . . 1102
12.93.3.3 horizontalSpacer . . . 1103
12.93.3.4 horizontalSpacer_2 . . . 1103
12.93.3.5 profileList . . . 1103
12.94 Ui_KbWidget Class Reference . . . 1103
12.94.1 Detailed Description . . . 1105
12.94.2 Member Function Documentation . . . 1105
12.94.2.1 retranslateUi . . . 1105
12.94.2.2 setupUi . . . 1106
12.94.3 Field Documentation . . . 1109
12.94.3.1 bindTab . . . 1109
12.94.3.2 bindWidget . . . 1109
12.94.3.3 devTab . . . 1109
12.94.3.4 fwLabel . . . 1109
12.94.3.5 fwUpdButton . . . 1109
12.94.3.6 fwUpdLabel . . . 1110
12.94.3.7 fwUpdLayout . . . 1110
12.94.3.8 gridLayout . . . 1110
12.94.3.9 gridLayout_2 . . . 1110
12.94.3.10 gridLayout_3 . . . 1110
12.94.3.11 gridLayout_4 . . . 1110
12.94.3.12 gridLayout_5 . . . 1110
12.94.3.13 gridLayout_7 . . . 1110
12.94.3.14 horizontalSpacer_2 . . . 1110
12.94.3.15 horizontalSpacer_3 . . . 1111
12.94.3.16 hwSaveButton . . . 1111
12.94.3.17 kPerfTab . . . 1111
12.94.3.18 kPerfWidget . . . 1111
12.94.3.19 label_3 . . . 1111
12.94.3.20 label_4 . . . 1111
12.94.3.21 label_9 . . . 1111
12.94.3.22 lightTab . . . 1111

12.94.3.23 lightWidget . . . 1111
12.94.3.24 modesList . . . 1112
12.94.3.25 mPerfTab . . . 1112
12.94.3.26 mPerfWidget . . . 1112
12.94.3.27 pollLabel . . . 1112
12.94.3.28 pollLabel2 . . . 1112
12.94.3.29 profileBox . . . 1112
12.94.3.30 serialLabel . . . 1112
12.94.3.31 tabWidget . . . 1112
12.94.3.32 verticalSpacer . . . 1112
12.95 Ui_KPerfWidget Class Reference . . . 1113
12.95.1 Detailed Description . . . 1116
12.95.2 Member Function Documentation . . . 1116
12.95.2.1 retranslateUi . . . 1116
12.95.2.2 setupUi . . . 1117
12.95.3 Field Documentation . . . 1124
12.95.3.1 capsBox . . . 1124
12.95.3.2 capsColorOff . . . 1124
12.95.3.3 capsColorOn . . . 1124
12.95.3.4 copyButton . . . 1124
12.95.3.5 gridLayout . . . 1124
12.95.3.6 horizontalLayout . . . 1124
12.95.3.7 horizontalLayout_2 . . . 1124
12.95.3.8 horizontalSpacer . . . 1125
12.95.3.9 horizontalSpacer_2 . . . 1125
12.95.3.10 horizontalSpacer_3 . . . 1125
12.95.3.11 horizontalSpacer_4 . . . 1125
12.95.3.12 horizontalSpacer_5 . . . 1125
12.95.3.13 horizontalSpacer_6 . . . 1125
12.95.3.14 horizontalSpacer_7 . . . 1125
12.95.3.15 horizontalSpacer_8 . . . 1125
12.95.3.16 intensityBox . . . 1125
12.95.3.17 k95Label1 . . . 1126
12.95.3.18 k95Label2 . . . 1126
12.95.3.19 k95Label3 . . . 1126
12.95.3.20 k95Label4 . . . 1126
12.95.3.21 k95Label5 . . . 1126
12.95.3.22 k95Label6 . . . 1126
12.95.3.23 k95Line . . . 1126
12.95.3.24 k95Spacer . . . 1126

12.95.3.25 label . . . 1126
12.95.3.26 label_10 . . . 1127
12.95.3.27 label_11 . . . 1127
12.95.3.28 label_12 . . . 1127
12.95.3.29 label_14 . . . 1127
12.95.3.30 label_15 . . . 1127
12.95.3.31 label_16 . . . 1127
12.95.3.32 label_17 . . . 1127
12.95.3.33 label_18 . . . 1127
12.95.3.34 label_19 . . . 1127
12.95.3.35 label_2 . . . 1128
12.95.3.36 label_20 . . . 1128
12.95.3.37 label_21 . . . 1128
12.95.3.38 label_22 . . . 1128
12.95.3.39 label_23 . . . 1128
12.95.3.40 label_24 . . . 1128
12.95.3.41 label_3 . . . 1128
12.95.3.42 label_4 . . . 1128
12.95.3.43 label_5 . . . 1128
12.95.3.44 label_6 . . . 1129
12.95.3.45 label_7 . . . 1129
12.95.3.46 label_8 . . . 1129
12.95.3.47 lightBox . . . 1129
12.95.3.48 lightColor1 . . . 1129
12.95.3.49 lightColor2 . . . 1129
12.95.3.50 lightColor3 . . . 1129
12.95.3.51 line . . . 1129
12.95.3.52 line_2 . . . 1129
12.95.3.53 line_3 . . . 1130
12.95.3.54 lockBox . . . 1130
12.95.3.55 lockColorOff . . . 1130
12.95.3.56 lockColorOn . . . 1130
12.95.3.57 macroBox . . . 1130
12.95.3.58 macroColorOff . . . 1130
12.95.3.59 macroColorOn . . . 1130
12.95.3.60 modeBox . . . 1130
12.95.3.61 modeColorOff . . . 1130
12.95.3.62 modeColorOn . . . 1131
12.95.3.63 muteBox . . . 1131
12.95.3.64 muteColorNA . . . 1131

12.95.3.65 muteColorOff . . . 1131
12.95.3.66 muteColorOn . . . 1131
12.95.3.67 numBox . . . 1131
12.95.3.68 numColorOff . . . 1131
12.95.3.69 numColorOn . . . 1131
12.95.3.70 scrollBox . . . 1131
12.95.3.71 scrollColorOff . . . 1132
12.95.3.72 scrollColorOn . . . 1132
12.95.3.73 verticalSpacer . . . 1132
12.95.3.74 verticalSpacer_2 . . . 1132
12.95.3.75 verticalSpacer_3 . . . 1132
12.95.3.76 verticalSpacer_4 . . . 1132
12.96 Ui_LayoutDialog Class Reference . . . 1132
12.96.1 Detailed Description . . . 1134
12.96.2 Member Function Documentation . . . 1134
12.96.2.1 retranslateUi . . . 1134
12.96.2.2 setupUi . . . 1134
12.96.3 Field Documentation . . . 1136
12.96.3.1 buttonBox . . . 1136
12.96.3.2 comboBox . . . 1136
12.96.3.3 gridLayout . . . 1136
12.96.3.4 horizontalSpacer . . . 1136
12.96.3.5 horizontalSpacer_2 . . . 1136
12.96.3.6 label . . . 1137
12.96.3.7 label_2 . . . 1137
12.96.3.8 verticalSpacer . . . 1137
12.96.3.9 verticalSpacer_2 . . . 1137
12.96.3.10 verticalSpacer_3 . . . 1137
12.96.3.11 verticalSpacer_4 . . . 1137
12.97 Ui_MainWindow Class Reference . . . 1137
12.97.1 Detailed Description . . . 1139
12.97.2 Member Function Documentation . . . 1139
12.97.2.1 retranslateUi . . . 1139
12.97.2.2 setupUi . . . 1139
12.97.3 Field Documentation . . . 1140
12.97.3.1 actionAbout . . . 1140
12.97.3.2 actionExit . . . 1140
12.97.3.3 centralWidget . . . 1140
12.97.3.4 gridLayout . . . 1141
12.97.3.5 tabWidget . . . 1141

12.98 Ui_ModeSelectDialog Class Reference . . . 1141
12.98.1 Detailed Description . . . 1142
12.98.2 Member Function Documentation . . . 1142
12.98.2.1 retranslateUi . . . 1142
12.98.2.2 setupUi . . . 1143
12.98.3 Field Documentation . . . 1144
12.98.3.1 buttonBox . . . 1144
12.98.3.2 gridLayout . . . 1144
12.98.3.3 label . . . 1144
12.98.3.4 modeList . . . 1144
12.98.3.5 selAllButton . . . 1144
12.98.3.6 selNoneButton . . . 1145
12.99 Ui_MPerfWidget Class Reference . . . 1145
12.99.1 Detailed Description . . . 1147
12.99.2 Member Function Documentation . . . 1148
12.99.2.1 retranslateUi . . . 1148
12.99.2.2 setupUi . . . 1149
12.99.3 Field Documentation . . . 1156
12.99.3.1 aSnapBox . . . 1156
12.99.3.2 copyButton . . . 1156
12.99.3.3 eBox1 . . . 1157
12.99.3.4 eBox2 . . . 1157
12.99.3.5 eBox3 . . . 1157
12.99.3.6 eBox4 . . . 1157
12.99.3.7 eBox5 . . . 1157
12.99.3.8 gridLayout . . . 1157
12.99.3.9 horizontalLayout . . . 1157
12.99.3.10 horizontalLayout_2 . . . 1157
12.99.3.11 horizontalLayout_4 . . . 1157
12.99.3.12 horizontalSpacer . . . 1158
12.99.3.13 horizontalSpacer_2 . . . 1158
12.99.3.14 horizontalSpacer_3 . . . 1158
12.99.3.15 horizontalSpacer_4 . . . 1158
12.99.3.16 horizontalSpacer_5 . . . 1158
12.99.3.17 horizontalSpacer_6 . . . 1158
12.99.3.18 horizontalSpacer_7 . . . 1158
12.99.3.19 iButton0 . . . 1158
12.99.3.20 iButton1 . . . 1158
12.99.3.21 iButton2 . . . 1159
12.99.3.22 iButton3 . . . 1159

12.99.3.23 Button4 . . . 1159
12.99.3.24 Button5 . . . 1159
12.99.3.25 ButtonO . . . 1159
12.99.3.26 ndicBox . . . 1159
12.99.3.27 abel . . . 1159
12.99.3.28 abel_11 . . . 1159
12.99.3.29 abel_12 . . . 1159
12.99.3.30 abel_13 . . . 1160
12.99.3.31 abel_14 . . . 1160
12.99.3.32 abel_15 . . . 1160
12.99.3.33 abel_2 . . . 1160
12.99.3.34 abel_25 . . . 1160
12.99.3.35 abel_3 . . . 1160
12.99.3.36 abel_4 . . . 1160
12.99.3.37 abel_5 . . . 1160
12.99.3.38 abel_6 . . . 1160
12.99.3.39 abel_7 . . . 1161
12.99.3.40 abel_8 . . . 1161
12.99.3.41 lHeightBox . . . 1161
12.99.3.42 ine . . . 1161
12.99.3.43 ine_2 . . . 1161
12.99.3.44 ine_3 . . . 1161
12.99.3.45 spinBox . . . 1161
12.99.3.46 verticalLayout . . . 1161
12.99.3.47 verticalSpacer . . . 1161
12.99.3.48 verticalSpacer_2 . . . 1162
12.99.3.49 verticalSpacer_3 . . . 1162
12.99.3.50 xBox0 . . . 1162
12.99.3.51 xBox1 . . . 1162
12.99.3.52 xBox2 . . . 1162
12.99.3.53 xBox3 . . . 1162
12.99.3.54 xBox4 . . . 1162
12.99.3.55 xBox5 . . . 1162
12.99.3.56 xSlider0 . . . 1162
12.99.3.57 xSlider1 . . . 1163
12.99.3.58 xSlider2 . . . 1163
12.99.3.59 xSlider3 . . . 1163
12.99.3.60 xSlider4 . . . 1163
12.99.3.61 xSlider5 . . . 1163
12.99.3.62 xyBox . . . 1163

12.99.3.63 yBox0 . . . 1163
12.99.3.64 yBox1 . . . 1163
12.99.3.65 yBox2 . . . 1163
12.99.3.66 yBox3 . . . 1164
12.99.3.67 yBox4 . . . 1164
12.99.3.68 yBox5 . . . 1164
12.99.3.69 ySlider0 . . . 1164
12.99.3.70 ySlider1 . . . 1164
12.99.3.71 ySlider2 . . . 1164
12.99.3.72 ySlider3 . . . 1164
12.99.3.73 ySlider4 . . . 1164
12.99.3.74 ySlider5 . . . 1164
12.100 Ui_RebindWidget Class Reference . . . 1165
12.100.1 Detailed Description . . . 1168
12.100.2 Member Function Documentation . . . 1168
12.100.2.1 retranslateUi . . . 1168
12.100.2.2 setupUi . . . 1171
12.100.3 Field Documentation . . . 1180
12.100.3.1 actionTab . . . 1180
12.100.3.2 animBox . . . 1180
12.100.3.3 animButton . . . 1181
12.100.3.4 animKrBox . . . 1181
12.100.3.5 animOnceBox . . . 1181
12.100.3.6 animTab . . . 1181
12.100.3.7 applyButton . . . 1181
12.100.3.8 btnClearMacro . . . 1181
12.100.3.9 btnStartMacro . . . 1181
12.100.3.10 btnStopMacro . . . 1181
12.100.3.11 cancelButton . . . 1182
12.100.3.12 delayButtons . . . 1182
12.100.3.13 dpiBox . . . 1182
12.100.3.14 dpiButton . . . 1182
12.100.3.15 dpiCustLabel . . . 1182
12.100.3.16 dpiCustXBox . . . 1182
12.100.3.17 dpiCustYBox . . . 1182
12.100.3.18 dpiWarning . . . 1182
12.100.3.19 fnBox . . . 1183
12.100.3.20 fnButton . . . 1183
12.100.3.21 gridLayout . . . 1183
12.100.3.22 gridLayout_10 . . . 1183

12.100.3.23 gridLayout_2 . . . 1183
12.100.3.24 gridLayout_3 . . . 1183
12.100.3.25 gridLayout_4 . . . 1183
12.100.3.26 gridLayout_5 . . . 1183
12.100.3.27 gridLayout_8 . . . 1183
12.100.3.28 groupBox_2 . . . 1184
12.100.3.29 groupBox_3 . . . 1184
12.100.3.30 groupBox_4 . . . 1184
12.100.3.31 horizontalLayout . . . 1184
12.100.3.32 horizontalLayout_2 . . . 1184
12.100.3.33 horizontalLayout_3 . . . 1184
12.100.3.34 horizontalSpacer . . . 1184
12.100.3.35 horizontalSpacer_10 . . . 1184
12.100.3.36 horizontalSpacer_11 . . . 1184
12.100.3.37 horizontalSpacer_12 . . . 1185
12.100.3.38 horizontalSpacer_13 . . . 1185
12.100.3.39 horizontalSpacer_14 . . . 1185
12.100.3.40 horizontalSpacer_15 . . . 1185
12.100.3.41 horizontalSpacer_16 . . . 1185
12.100.3.42 horizontalSpacer_17 . . . 1185
12.100.3.43 horizontalSpacer_18 . . . 1185
12.100.3.44 horizontalSpacer_19 . . . 1185
12.100.3.45 horizontalSpacer_2 . . . 1185
12.100.3.46 horizontalSpacer_20 . . . 1186
12.100.3.47 horizontalSpacer_21 . . . 1186
12.100.3.48 horizontalSpacer_22 . . . 1186
12.100.3.49 horizontalSpacer_23 . . . 1186
12.100.3.50 horizontalSpacer_24 . . . 1186
12.100.3.51 horizontalSpacer_25 . . . 1186
12.100.3.52 horizontalSpacer_26 . . . 1186
12.100.3.53 horizontalSpacer_3 . . . 1186
12.100.3.54 horizontalSpacer_4 . . . 1186
12.100.3.55 horizontalSpacer_5 . . . 1187
12.100.3.56 horizontalSpacer_6 . . . 1187
12.100.3.57 horizontalSpacer_7 . . . 1187
12.100.3.58 horizontalSpacer_8 . . . 1187
12.100.3.59 horizontalSpacer_9 . . . 1187
12.100.3.60 keyTab . . . 1187
12.100.3.61 label . . . 1187
12.100.3.62 layoutWidget . . . 1187

12.100.3.63 layoutWidget1 . . . 1187
12.100.3.64 layoutWidget2 . . . 1188
12.100.3.65 layoutWidget_2 . . . 1188
12.100.3.66 ml_macro . . . 1188
12.100.3.67 rightBox . . . 1188
12.100.3.68 rightButton . . . 1188
12.100.3.69 rightWrapBox . . . 1188
12.100.3.70 rockBox . . . 1188
12.100.3.71 rockButton . . . 1188
12.100.3.72 macroTab . . . 1188
12.100.3.73 rmb2Box . . . 1189
12.100.3.74 rmb2Button . . . 1189
12.100.3.75 rmbBox . . . 1189
12.100.3.76 rmbButton . . . 1189
12.100.3.77 rmediaBox . . . 1189
12.100.3.78 rmediaButton . . . 1189
12.100.3.79 rmodBox . . . 1189
12.100.3.80 rmodButton . . . 1189
12.100.3.81 rmodeBox . . . 1190
12.100.3.82 rmodeButton . . . 1190
12.100.3.83 rmodeWrapBox . . . 1190
12.100.3.84 rmouseTab . . . 1190
12.100.3.85 rnumBox . . . 1190
12.100.3.86 rnumButton . . . 1190
12.100.3.87 rprogramKpBox . . . 1190
12.100.3.88 rprogramKpButton . . . 1190
12.100.3.89 rprogramKpExtra . . . 1191
12.100.3.90 rprogramKpModeBox . . . 1191
12.100.3.91 rprogramKpSIBox . . . 1191
12.100.3.92 rprogramKrBox . . . 1191
12.100.3.93 rprogramKrButton . . . 1191
12.100.3.94 rprogramKrExtra . . . 1191
12.100.3.95 rprogramKrModeBox . . . 1191
12.100.3.96 rprogramKrSIBox . . . 1191
12.100.3.97 rprogramTab . . . 1191
12.100.3.98 rprogTipLabel . . . 1192
12.100.3.99 rsteMacroBox . . . 1192
12.100.3.100 rsteMacroComment . . . 1192
12.100.3.101 rsteMacroText . . . 1192
12.100.3.102 rx_delay_asTyped . . . 1192

12.100.3.103 delay_default . . . 1192
12.100.3.104 delay_no . . . 1192
12.100.3.105 resetButton . . . 1192
12.100.3.106 tab Widget . . . 1193
12.100.3.107 text Buffer . . . 1193
12.100.3.108 typingBox . . . 1193
12.100.3.109 typingButton . . . 1193
12.100.3.110 unbindButton . . . 1193
12.100.3.111 verticalLayout . . . 1193
12.100.3.112 verticalLayout_2 . . . 1193
12.100.3.113 verticalSpacer . . . 1193
12.100.3.114 verticalSpacer_10 . . . 1193
12.100.3.115 verticalSpacer_11 . . . 1194
12.100.3.116 verticalSpacer_2 . . . 1194
12.100.3.117 verticalSpacer_3 . . . 1194
12.100.3.118 verticalSpacer_4 . . . 1194
12.100.3.119 verticalSpacer_5 . . . 1194
12.100.3.120 verticalSpacer_6 . . . 1194
12.100.3.121 verticalSpacer_7 . . . 1194
12.100.3.122 verticalSpacer_8 . . . 1194
12.100.3.123 wheelBox . . . 1194
12.100.3.124 wheelButton . . . 1195
12.101 Ui_SettingsWidget Class Reference . . . 1195
12.101.1 Detailed Description . . . 1197
12.101.2 Member Function Documentation . . . 1197
12.101.2.1 retranslateUi . . . 1197
12.101.2.2 setupUi . . . 1198
12.101.3 Field Documentation . . . 1202
12.101.3.1 altBox . . . 1202
12.101.3.2 altLabel . . . 1202
12.101.3.3 autoFWBox . . . 1202
12.101.3.4 capsBox . . . 1202
12.101.3.5 capsLabel . . . 1202
12.101.3.6 ctrlBox . . . 1203
12.101.3.7 ctrlLabel . . . 1203
12.101.3.8 devicesLabel . . . 1203
12.101.3.9 extraButton . . . 1203
12.101.3.10 gridLayout . . . 1203
12.101.3.11 gridLayout_3 . . . 1203
12.101.3.12 gridLayout_5 . . . 1203

12.101.3.13 HorizontalLayout . . . 1203
12.101.3.14 HorizontalLayout_2 . . . 1203
12.101.3.15 HorizontalSpacer . . . 1204
12.101.3.16 HorizontalSpacer_10 . . . 1204
12.101.3.17 HorizontalSpacer_2 . . . 1204
12.101.3.18 HorizontalSpacer_7 . . . 1204
12.101.3.19 HorizontalSpacer_9 . . . 1204
12.101.3.20 label . . . 1204
12.101.3.21 label_2 . . . 1204
12.101.3.22 label_3 . . . 1204
12.101.3.23 label_5 . . . 1204
12.101.3.24 label_6 . . . 1205
12.101.3.25 label_8 . . . 1205
12.101.3.26 layoutBox . . . 1205
12.101.3.27 line . . . 1205
12.101.3.28 line_2 . . . 1205
12.101.3.29 line_3 . . . 1205
12.101.3.30 loginItemBox . . . 1205
12.101.3.31 pushButton . . . 1205
12.101.3.32 shiftBox . . . 1205
12.101.3.33 shiftLabel . . . 1206
12.101.3.34 versionLabel . . . 1206
12.101.3.35 verticalSpacer . . . 1206
12.101.3.36 verticalSpacer_10 . . . 1206
12.101.3.37 verticalSpacer_2 . . . 1206
12.101.3.38 verticalSpacer_3 . . . 1206
12.101.3.39 verticalSpacer_4 . . . 1206
12.101.3.40 verticalSpacer_5 . . . 1206
12.101.3.41 verticalSpacer_6 . . . 1206
12.101.3.42 verticalSpacer_7 . . . 1207
12.101.3.43 verticalSpacer_8 . . . 1207
12.101.3.44 winBox . . . 1207
12.101.3.45 winLabel . . . 1207
12.102 UsbId Struct Reference . . . 1207
12.102.1 Detailed Description . . . 1209
12.102.2 Constructor & Destructor Documentation . . . 1209
12.102.2.1 UsbId . . . 1209
12.102.2.2 UsbId . . . 1209
12.102.2.3 UsbId . . . 1209
12.102.3 Member Function Documentation . . . 1209

12.102.3.1 guidString . . . 1209
12.102.3.2 guidString . . . 1209
12.102.3.3 hwModifiedString . . . 1210
12.102.3.4 hwModifiedString . . . 1210
12.102.3.5 modifiedString . . . 1210
12.102.3.6 modifiedString . . . 1211
12.102.3.7 newGuid . . . 1211
12.102.3.8 newModified . . . 1211
12.102.4 Field Documentation . . . 1212
12.102.4.1 guid . . . 1212
12.102.4.2 hwModified . . . 1212
12.102.4.3 modified . . . 1212

13 File Documentation **1213**

13.1 BUILD.md File Reference . . . 1213
13.2 DAEMON.md File Reference . . . 1213
13.3 README.md File Reference . . . 1213
13.4 ROADMAP.md File Reference . . . 1213
13.5 src/ckb/animadddialog.cpp File Reference . . . 1213
13.6 src/ckb/animadddialog.h File Reference . . . 1213
13.7 src/ckb/animscript.cpp File Reference . . . 1214
13.7.1 Function Documentation . . . 1215
13.7.1.1 urlParam . . . 1215
13.7.2 Variable Documentation . . . 1215
13.7.2.1 ONE_DAY . . . 1215
13.8 src/ckb/animscript.h File Reference . . . 1216
13.8.1 Data Structure Documentation . . . 1216
13.8.1.1 struct AnimScript::Param . . . 1216
13.9 src/ckb/animsettingdialog.cpp File Reference . . . 1217
13.9.1 Function Documentation . . . 1218
13.9.1.1 angleFlip . . . 1218
13.10 src/ckb/animsettingdialog.h File Reference . . . 1219
13.11 src/ckb/autorun.cpp File Reference . . . 1219
13.11.1 Variable Documentation . . . 1220
13.11.1.1 settingPath . . . 1220
13.12 src/ckb/autorun.h File Reference . . . 1220
13.13 src/ckb/ckb-anim.h File Reference . . . 1221
13.13.1 Data Structure Documentation . . . 1223
13.13.1.1 struct ckb_key . . . 1223
13.13.1.2 struct ckb_runctx . . . 1224

13.13.1.3 struct ckb_gradient . . . 1225
13.13.2 Macro Definition Documentation . . . 1226
13.13.2.1 CKB_CONTAINER . . . 1226
13.13.2.2 CKB_COPYRIGHT . . . 1226
13.13.2.3 CKB_DESCRIPTION . . . 1226
13.13.2.4 CKB_DOWN . . . 1226
13.13.2.5 CKB_GRAD_MAX . . . 1227
13.13.2.6 CKB_GUID . . . 1227
13.13.2.7 CKB_KEYCLEAR . . . 1227
13.13.2.8 CKB_KEYNAME_MAX . . . 1227
13.13.2.9 CKB_KP_NAME . . . 1227
13.13.2.10 CKB_KP_NONE . . . 1227
13.13.2.11 CKB_KP_POSITION . . . 1227
13.13.2.12 CKB_KPMODE . . . 1227
13.13.2.13 CKB_LICENSE . . . 1227
13.13.2.14 CKB_LIVEPARAMS . . . 1227
13.13.2.15 CKB_MAX_WORD . . . 1227
13.13.2.16 CKB_NAME . . . 1228
13.13.2.17 CKB_PARAM . . . 1228
13.13.2.18 CKB_PARAM_AGRADIENT . . . 1228
13.13.2.19 CKB_PARAM_ANGLE . . . 1228
13.13.2.20 CKB_PARAM_ARGB . . . 1228
13.13.2.21 CKB_PARAM_BOOL . . . 1228
13.13.2.22 CKB_PARAM_DOUBLE . . . 1228
13.13.2.23 CKB_PARAM_GRADIENT . . . 1228
13.13.2.24 CKB_PARAM_LABEL . . . 1228
13.13.2.25 CKB_PARAM_LONG . . . 1228
13.13.2.26 CKB_PARAM_RGB . . . 1229
13.13.2.27 CKB_PARAM_STRING . . . 1229
13.13.2.28 CKB_PARSE_AGRADIENT . . . 1229
13.13.2.29 CKB_PARSE_ANGLE . . . 1229
13.13.2.30 CKB_PARSE_ARGB . . . 1229
13.13.2.31 CKB_PARSE_BOOL . . . 1229
13.13.2.32 CKB_PARSE_DOUBLE . . . 1229
13.13.2.33 CKB_PARSE_GRADIENT . . . 1229
13.13.2.34 CKB_PARSE_LONG . . . 1229
13.13.2.35 CKB_PARSE_RGB . . . 1229
13.13.2.36 CKB_PARSE_STRING . . . 1230
13.13.2.37 CKB_PREEMPT . . . 1230
13.13.2.38 CKB_PRESET_END . . . 1230

13.13.2.39 CKB_PRESET_PARAM . . . 1230
13.13.2.40 CKB_PRESET_START . . . 1230
13.13.2.41 CKB_REAL_ANGLE . . . 1230
13.13.2.42 CKB_REPEAT . . . 1230
13.13.2.43 CKB_TIME_ABSOLUTE . . . 1230
13.13.2.44 CKB_TIME_DURATION . . . 1230
13.13.2.45 CKB_TIMEMODE . . . 1230
13.13.2.46 CKB_UP . . . 1230
13.13.2.47 CKB_VERSION . . . 1230
13.13.2.48 FALSE . . . 1231
13.13.2.49 M_PI . . . 1231
13.13.2.50 TRUE . . . 1231
13.13.3 Function Documentation . . . 1231
13.13.3.1 ckb_alpha_blend . . . 1231
13.13.3.2 ckb_frame . . . 1231
13.13.3.3 ckb_getline . . . 1231
13.13.3.4 ckb_grad_color . . . 1232
13.13.3.5 ckb_info . . . 1233
13.13.3.6 ckb_init . . . 1233
13.13.3.7 ckb_keypress . . . 1233
13.13.3.8 ckb_parameter . . . 1234
13.13.3.9 ckb_read_params . . . 1234
13.13.3.10 ckb_scan_grad . . . 1235
13.13.3.11 ckb_start . . . 1235
13.13.3.12 ckb_time . . . 1236
13.13.3.13 main . . . 1236
13.13.3.14 printurl . . . 1238
13.13.3.15 urldecode . . . 1239
13.14 src/ckb/ckbsettings.cpp File Reference . . . 1240
13.14.1 Macro Definition Documentation . . . 1240
13.14.1.1 lockMutex . . . 1240
13.14.1.2 lockMutexCache . . . 1241
13.14.1.3 lockMutexStatic . . . 1241
13.14.1.4 lockMutexStatic2 . . . 1241
13.14.2 Function Documentation . . . 1241
13.14.2.1 cacheWritesInProgress . . . 1241
13.14.2.2 globalSettings . . . 1241
13.14.2.3 settingsCacheMutex . . . 1242
13.14.2.4 settingsMutex . . . 1242
13.14.3 Variable Documentation . . . 1242

13.14.3.1 _globalSettings . . . 1242
13.14.3.2 globalCache . . . 1242
13.14.3.3 globalThread . . . 1242
13.15 src/ckb/ckbsettings.h File Reference . . . 1242
13.16 src/ckb/ckbsettingswriter.cpp File Reference . . . 1243
13.16.1 Macro Definition Documentation . . . 1244
13.16.1.1 lockMutex . . . 1244
13.16.1.2 lockMutexCache . . . 1244
13.16.1.3 lockMutexStatic . . . 1244
13.16.1.4 lockMutexStatic2 . . . 1244
13.16.2 Variable Documentation . . . 1245
13.16.2.1 cacheWritesInProgress . . . 1245
13.16.2.2 settingsCacheMutex . . . 1245
13.16.2.3 settingsMutex . . . 1245
13.17 src/ckb/ckbsettingswriter.h File Reference . . . 1245
13.18 src/ckb/colorbutton.cpp File Reference . . . 1245
13.19 src/ckb/colorbutton.h File Reference . . . 1246
13.20 src/ckb/colormap.cpp File Reference . . . 1247
13.20.1 Function Documentation . . . 1247
13.20.1.1 qs_strcmp . . . 1247
13.21 src/ckb/colormap.h File Reference . . . 1248
13.21.1 Typedef Documentation . . . 1249
13.21.1.1 QColorMap . . . 1249
13.21.1.2 QColorMapIterator . . . 1249
13.21.1.3 QMutableColorMapIterator . . . 1249
13.22 src/ckb/extrasettingswidget.cpp File Reference . . . 1249
13.22.1 Variable Documentation . . . 1249
13.22.1.1 lastSharedDimming . . . 1249
13.23 src/ckb/extrasettingswidget.h File Reference . . . 1250
13.24 src/ckb/fwupgradedialog.cpp File Reference . . . 1250
13.24.1 Data Structure Documentation . . . 1251
13.24.1.1 struct KbId . . . 1251
13.24.2 Function Documentation . . . 1252
13.24.2.1 DIALOG_HEIGHT_MAX . . . 1252
13.24.2.2 verifyFw . . . 1252
13.24.3 Variable Documentation . . . 1253
13.24.3.1 DIALOG_HEIGHT_MIN . . . 1253
13.24.3.2 DIALOG_WIDTH . . . 1253
13.24.3.3 ids . . . 1253
13.25 src/ckb/fwupgradedialog.h File Reference . . . 1253

13.26 src/ckb/gradientbutton.cpp File Reference . . . 1254
13.27 src/ckb/gradientbutton.h File Reference . . . 1255
13.28 src/ckb/gradientdialog.cpp File Reference . . . 1256
13.28.1 Variable Documentation . . . 1256
13.28.1.1 prefsPath . . . 1256
13.29 src/ckb/gradientdialog.h File Reference . . . 1256
13.30 src/ckb/gradientdialogwidget.cpp File Reference . . . 1257
13.31 src/ckb/gradientdialogwidget.h File Reference . . . 1257
13.32 src/ckb/kb.cpp File Reference . . . 1258
13.32.1 Variable Documentation . . . 1259
13.32.1.1 activeDevices . . . 1259
13.32.1.2 notifyPathMutex . . . 1259
13.32.1.3 notifyPaths . . . 1259
13.33 src/ckb/kb.h File Reference . . . 1259
13.34 src/ckb/kbanim.cpp File Reference . . . 1260
13.34.1 Typedef Documentation . . . 1261
13.34.1.1 blendFunc . . . 1261
13.34.2 Function Documentation . . . 1261
13.34.2.1 blendAdd . . . 1261
13.34.2.2 blendDivide . . . 1261
13.34.2.3 blendMultiply . . . 1261
13.34.2.4 blendNormal . . . 1262
13.34.2.5 blendSubtract . . . 1262
13.34.3 Variable Documentation . . . 1262
13.34.3.1 functions . . . 1262
13.35 src/ckb/kbanim.h File Reference . . . 1262
13.36 src/ckb/kbanimwidget.cpp File Reference . . . 1263
13.37 src/ckb/kbanimwidget.h File Reference . . . 1263
13.38 src/ckb/kbbind.cpp File Reference . . . 1264
13.39 src/ckb/kbbind.h File Reference . . . 1265
13.40 src/ckb/kbbindwidget.cpp File Reference . . . 1266
13.41 src/ckb/kbbindwidget.h File Reference . . . 1266
13.42 src/ckb/kbfirmware.cpp File Reference . . . 1267
13.42.1 Function Documentation . . . 1268
13.42.1.1 tableName . . . 1268
13.42.2 Variable Documentation . . . 1268
13.42.2.1 AUTO_CHECK_TIME . . . 1268
13.43 src/ckb/kbfirmware.h File Reference . . . 1268
13.44 src/ckb/kblight.cpp File Reference . . . 1269
13.44.1 Function Documentation . . . 1270

13.44.1.1 lToS . . . 1270
13.44.1.2 monoRgb . . . 1271
13.44.1.3 sToL . . . 1271
13.44.2 Variable Documentation . . . 1272
13.44.2.1 _shareDimming . . . 1272
13.44.2.2 activeLights . . . 1272
13.45 src/ckb/kblight.h File Reference . . . 1272
13.46 src/ckb/kblightwidget.cpp File Reference . . . 1273
13.47 src/ckb/kblightwidget.h File Reference . . . 1273
13.48 src/ckb/kbmanager.cpp File Reference . . . 1274
13.48.1 Variable Documentation . . . 1275
13.48.1.1 devpath . . . 1275
13.49 src/ckb/kbmanager.h File Reference . . . 1275
13.49.1 Macro Definition Documentation . . . 1276
13.49.1.1 DAEMON_UNAVAILABLE_STR . . . 1276
13.50 src/ckb/kbmode.cpp File Reference . . . 1276
13.51 src/ckb/kbmode.h File Reference . . . 1276
13.52 src/ckb/kbperf.cpp File Reference . . . 1277
13.52.1 Macro Definition Documentation . . . 1277
13.52.1.1 map_last . . . 1277
13.53 src/ckb/kbperf.h File Reference . . . 1277
13.54 src/ckb/kbprofile.cpp File Reference . . . 1278
13.55 src/ckb/kbprofile.h File Reference . . . 1279
13.56 src/ckb/kbprofiledialog.cpp File Reference . . . 1279
13.57 src/ckb/kbprofiledialog.h File Reference . . . 1280
13.58 src/ckb/kbwidget.cpp File Reference . . . 1281
13.59 src/ckb/kbwidget.h File Reference . . . 1281
13.60 src/ckb/keyaction.cpp File Reference . . . 1282
13.60.1 Variable Documentation . . . 1283
13.60.1.1 ANIM_KRSTOP . . . 1283
13.60.1.2 ANIM_ONCE . . . 1283
13.61 src/ckb/keyaction.h File Reference . . . 1283
13.62 src/ckb/keymap.cpp File Reference . . . 1284
13.62.1 Data Structure Documentation . . . 1286
13.62.1.1 struct KeyPatch . . . 1286
13.62.2 Macro Definition Documentation . . . 1286
13.62.2.1 ANSI_ENTER_H . . . 1286
13.62.2.2 ANSI_ENTER_W . . . 1287
13.62.2.3 ANSI_ENTER_X . . . 1287
13.62.2.4 ANSI_ENTER_Y . . . 1287

13.62.2.5 ANSI_LSHIFT_W . . . 1287
13.62.2.6 ANSI_LSHIFT_X . . . 1287
13.62.2.7 GLAIVE_HEIGHT . . . 1287
13.62.2.8 GLAIVE_WIDTH . . . 1287
13.62.2.9 HARPOON_HEIGHT . . . 1287
13.62.2.10 HARPOON_WIDTH . . . 1287
13.62.2.11 K63_HEIGHT . . . 1287
13.62.2.12 K63_TOP_COUNT . . . 1288
13.62.2.13 K63_WIDTH . . . 1288
13.62.2.14 K65_HEIGHT . . . 1288
13.62.2.15 K65_TOP_COUNT . . . 1288
13.62.2.16 K65_WIDTH . . . 1288
13.62.2.17 K68_HEIGHT . . . 1288
13.62.2.18 K68_TOP_COUNT . . . 1288
13.62.2.19 K68_WIDTH . . . 1288
13.62.2.20 K70_HEIGHT . . . 1288
13.62.2.21 K70_WIDTH . . . 1288
13.62.2.22 K70_X_START . . . 1289
13.62.2.23 K95_HEIGHT . . . 1289
13.62.2.24 K95_WIDTH . . . 1289
13.62.2.25 K95P_HEIGHT . . . 1289
13.62.2.26 K95P_WIDTH . . . 1289
13.62.2.27 K95P_X_START . . . 1289
13.62.2.28 KEYCOUNT_GLAIVE . . . 1289
13.62.2.29 KEYCOUNT_HARPOON . . . 1289
13.62.2.30 KEYCOUNT_K95 . . . 1289
13.62.2.31 KEYCOUNT_M65 . . . 1290
13.62.2.32 KEYCOUNT_SABRE . . . 1290
13.62.2.33 KEYCOUNT_SCIM . . . 1290
13.62.2.34 KSTRAFE_HEIGHT . . . 1290
13.62.2.35 KSTRAFE_WIDTH . . . 1290
13.62.2.36 KSTRAFE_X_START . . . 1290
13.62.2.37 LBARCOUNT_K95P . . . 1290
13.62.2.38 LBS . . . 1290
13.62.2.39 M65_HEIGHT . . . 1290
13.62.2.40 M65_WIDTH . . . 1290
13.62.2.41 N_LAYOUTS . . . 1291
13.62.2.42 N_MODELS . . . 1291
13.62.2.43 NS . . . 1291
13.62.2.44 patch . . . 1291

13.62.2.45 PATCH_COUNT . . . 1291
13.62.2.46 SABRE_HEIGHT . . . 1291
13.62.2.47 SABRE_WIDTH . . . 1291
13.62.2.48 SCIM_HEIGHT . . . 1291
13.62.2.49 SCIM_WIDTH . . . 1291
13.62.3 Function Documentation . . . 1291
13.62.3.1 _patch . . . 1291
13.62.3.2 getMap . . . 1292
13.62.3.3 patchABNT2 . . . 1296
13.62.3.4 patchANSI . . . 1297
13.62.3.5 patchISO . . . 1298
13.62.3.6 patchJP106 . . . 1298
13.62.3.7 patchJP106fn . . . 1299
13.62.3.8 patchnonJP106 . . . 1299
13.62.4 Variable Documentation . . . 1300
13.62.4.1 GlaiveKeys . . . 1300
13.62.4.2 HarpoonKeys . . . 1300
13.62.4.3 K63TopRow . . . 1300
13.62.4.4 K65TopRow . . . 1301
13.62.4.5 K68TopRow . . . 1301
13.62.4.6 K95BaseMap . . . 1301
13.62.4.7 K95Keys . . . 1301
13.62.4.8 K95PLbar . . . 1302
13.62.4.9 KStrafeKeys . . . 1302
13.62.4.10 M65Keys . . . 1303
13.62.4.11 patchDE . . . 1303
13.62.4.12 patchDK . . . 1303
13.62.4.13 patchDvorak . . . 1303
13.62.4.14 patchES . . . 1304
13.62.4.15 patchEU . . . 1304
13.62.4.16 patchFR . . . 1304
13.62.4.17 patchIT . . . 1304
13.62.4.18 patchJP . . . 1304
13.62.4.19 patchMX . . . 1305
13.62.4.20 patchNO . . . 1305
13.62.4.21 patchPTBR . . . 1305
13.62.4.22 patchSE . . . 1305
13.62.4.23 SabreKeys . . . 1305
13.62.4.24 ScimKeys . . . 1306
13.62.4.25 standardMaps . . . 1306

13.63 src/ckb/keymap.h File Reference . . . 1306
13.64 src/ckb/keywidget.cpp File Reference . . . 1307
13.64.1 Function Documentation . . . 1308
13.64.1.1 monoRgb . . . 1308
13.64.2 Variable Documentation . . . 1308
13.64.2.1 glaiveOverlay . . . 1308
13.64.2.2 harpOverlay . . . 1309
13.64.2.3 KEY_SIZE . . . 1309
13.64.2.4 m65Overlay . . . 1309
13.64.2.5 sabOverlay . . . 1309
13.64.2.6 scimOverlay . . . 1309
13.65 src/ckb/keywidget.h File Reference . . . 1309
13.66 src/ckb/kperfwidget.cpp File Reference . . . 1310
13.67 src/ckb/kperfwidget.h File Reference . . . 1310
13.67.1 Data Structure Documentation . . . 1311
13.67.1.1 struct KPerfWidget::IndicatorUi . . . 1311
13.68 src/ckb/layoutdialog.cpp File Reference . . . 1312
13.69 src/ckb/layoutdialog.h File Reference . . . 1313
13.70 src/ckb/macroreader.cpp File Reference . . . 1314
13.71 src/ckb/macroreader.h File Reference . . . 1314
13.72 src/ckb/main.cpp File Reference . . . 1315
13.72.1 Macro Definition Documentation . . . 1316
13.72.1.1 SHMEM_SIZE . . . 1316
13.72.1.2 SHMEM_SIZE_V015 . . . 1316
13.72.2 Enumeration Type Documentation . . . 1316
13.72.2.1 CommandLineParseResults . . . 1316
13.72.3 Function Documentation . . . 1317
13.72.3.1 appShare . . . 1317
13.72.3.2 isRunning . . . 1317
13.72.3.3 main . . . 1318
13.72.3.4 parseCommandLine . . . 1319
13.72.3.5 pidActive . . . 1321
13.73 src/ckb/mainwindow.cpp File Reference . . . 1321
13.73.1 Variable Documentation . . . 1322
13.73.1.1 appShare . . . 1322
13.73.1.2 configLabel . . . 1322
13.73.1.3 daemonDialogText . . . 1322
13.73.1.4 devpath . . . 1322
13.74 src/ckb/mainwindow.h File Reference . . . 1322
13.75 src/ckb/media.h File Reference . . . 1324

13.75.1 Macro Definition Documentation . . . 1324
13.75.1.1 ENUM_C . . . 1324
13.75.1.2 ENUM_END_C . . . 1324
13.75.1.3 EXTERN_C . . . 1324
13.75.2 Enumeration Type Documentation . . . 1324
13.75.2.1 muteState . . . 1324
13.75.3 Function Documentation . . . 1325
13.75.3.1 getMuteState . . . 1325
13.76 src/ckb/media_linux.cpp File Reference . . . 1326
13.76.1 Function Documentation . . . 1326
13.76.1.1 getMuteState . . . 1326
13.76.2 Variable Documentation . . . 1327
13.76.2.1 lastKnown . . . 1327
13.77 src/ckb/media_mac.m File Reference . . . 1327
13.77.1 Function Documentation . . . 1327
13.77.1.1 disableAppNap . . . 1327
13.77.1.2 getMuteState . . . 1328
13.78 src/ckb/modeselectdialog.cpp File Reference . . . 1328
13.79 src/ckb/modeselectdialog.h File Reference . . . 1329
13.80 src/ckb/mperfwidget.cpp File Reference . . . 1330
13.80.1 Macro Definition Documentation . . . 1330
13.80.1.1 SET_END . . . 1330
13.80.1.2 SET_START . . . 1330
13.80.2 Function Documentation . . . 1330
13.80.2.1 dpiExp . . . 1330
13.80.2.2 dpiLog . . . 1331
13.80.2.3 dpiRound . . . 1331
13.80.3 Variable Documentation . . . 1332
13.80.3.1 DPI_BASE . . . 1332
13.80.3.2 xyLinkPath . . . 1332
13.81 src/ckb/mperfwidget.h File Reference . . . 1332
13.81.1 Data Structure Documentation . . . 1333
13.81.1.1 struct MPerfWidget::DpiUi . . . 1333
13.82 src/ckb/quazip/crypt.h File Reference . . . 1334
13.82.1 Macro Definition Documentation . . . 1335
13.82.1.1 CRC32 . . . 1336
13.82.1.2 zdecode . . . 1336
13.82.1.3 zencode . . . 1336
13.82.2 Function Documentation . . . 1336
13.82.2.1 decrypt_byte . . . 1336

13.82.2.2 init_keys . . . 1336
13.82.2.3 update_keys . . . 1337
13.83 src/ckb/quazip/ioapi.h File Reference . . . 1338
13.83.1 Data Structure Documentation . . . 1340
13.83.1.1 struct zlib_filefunc_def_s . . . 1340
13.83.1.2 struct zlib_filefunc64_def_s . . . 1340
13.83.1.3 struct zlib_filefunc64_32_def_s . . . 1341
13.83.2 Macro Definition Documentation . . . 1341
13.83.2.1 __USE_FILE_OFFSET64 . . . 1341
13.83.2.2 __USE_LARGEFILE64 . . . 1341
13.83.2.3 _FILE_OFFSET_BIT . . . 1342
13.83.2.4 _LARGEFILE64_SOURCE . . . 1342
13.83.2.5 OF . . . 1342
13.83.2.6 ZCALLBACK . . . 1342
13.83.2.7 ZCLOSE64 . . . 1342
13.83.2.8 ZERROR64 . . . 1342
13.83.2.9 ZFAKECLOSE64 . . . 1342
13.83.2.10 ZLIB_FILEFUNC_MODE_CREATE . . . 1342
13.83.2.11 ZLIB_FILEFUNC_MODE_EXISTING . . . 1342
13.83.2.12 ZLIB_FILEFUNC_MODE_READ . . . 1342
13.83.2.13 ZLIB_FILEFUNC_MODE_READWRITEFILTER . . . 1343
13.83.2.14 ZLIB_FILEFUNC_MODE_WRITE . . . 1343
13.83.2.15 ZLIB_FILEFUNC_SEEK_CUR . . . 1343
13.83.2.16 ZLIB_FILEFUNC_SEEK_END . . . 1343
13.83.2.17 ZLIB_FILEFUNC_SEEK_SET . . . 1343
13.83.2.18 ZOPEN64 . . . 1343
13.83.2.19 ZREAD64 . . . 1343
13.83.2.20 ZSEEK64 . . . 1343
13.83.2.21 ZTELL64 . . . 1344
13.83.2.22 ZWRITE64 . . . 1344
13.83.3 Typedef Documentation . . . 1344
13.83.3.1 _Z_OF . . . 1344
13.83.3.2 zlib_filefunc64_32_def . . . 1344
13.83.3.3 zlib_filefunc64_def . . . 1344
13.83.3.4 zlib_filefunc_def . . . 1344
13.83.3.5 ZPOS64_T . . . 1344
13.83.4 Function Documentation . . . 1344
13.83.4.1 _Z_OF . . . 1344
13.83.4.2 _Z_OF . . . 1344
13.83.4.3 _Z_OF . . . 1344

13.83.4.4 _Z_OF . . . 1344
13.83.4.5 _Z_OF . . . 1344
13.83.4.6 fill_zlib_filefunc64_32_def_from_filefunc32 . . . 1345
13.84 src/ckb/quazip/JlCompress.cpp File Reference . . . 1345
13.84.1 Function Documentation . . . 1346
13.84.1.1 copyData . . . 1346
13.85 src/ckb/quazip/JlCompress.h File Reference . . . 1347
13.86 src/ckb/quazip/qioapi.cpp File Reference . . . 1347
13.86.1 Macro Definition Documentation . . . 1349
13.86.1.1 SEEK_CUR . . . 1349
13.86.1.2 SEEK_END . . . 1349
13.86.1.3 SEEK_SET . . . 1349
13.86.2 Function Documentation . . . 1349
13.86.2.1 call_zopen64 . . . 1349
13.86.2.2 call_zseek64 . . . 1349
13.86.2.3 call_ztell64 . . . 1349
13.86.2.4 fill_qiodevice64_filefunc . . . 1350
13.86.2.5 fill_qiodevice_filefunc . . . 1351
13.86.2.6 fill_zlib_filefunc64_32_def_from_filefunc32 . . . 1352
13.86.2.7 qiodevice64_seek_file_func . . . 1353
13.86.2.8 qiodevice64_tell_file_func . . . 1354
13.86.2.9 qiodevice_close_file_func . . . 1354
13.86.2.10 qiodevice_error_file_func . . . 1355
13.86.2.11 qiodevice_fakeclose_file_func . . . 1355
13.86.2.12 qiodevice_open_file_func . . . 1356
13.86.2.13 qiodevice_read_file_func . . . 1357
13.86.2.14 qiodevice_seek_file_func . . . 1357
13.86.2.15 qiodevice_tell_file_func . . . 1358
13.86.2.16 qiodevice_write_file_func . . . 1359
13.87 src/ckb/quazip/quaadler32.cpp File Reference . . . 1359
13.88 src/ckb/quazip/quaadler32.h File Reference . . . 1360
13.89 src/ckb/quazip/quachecksum32.h File Reference . . . 1361
13.90 src/ckb/quazip/quacrc32.cpp File Reference . . . 1362
13.91 src/ckb/quazip/quacrc32.h File Reference . . . 1363
13.92 src/ckb/quazip/quagzipfile.cpp File Reference . . . 1364
13.93 src/ckb/quazip/quagzipfile.h File Reference . . . 1365
13.94 src/ckb/quazip/quaziodevice.cpp File Reference . . . 1366
13.94.1 Macro Definition Documentation . . . 1367
13.94.1.1 QUAZIO_INBUFSIZE . . . 1367
13.94.1.2 QUAZIO_OUTBUFSIZE . . . 1367

13.95 src/ckb/quazip/quaziodevice.h File Reference . . . 1367
13.96 src/ckb/quazip/quazip.cpp File Reference . . . 1368
13.96.1 Function Documentation . . . 1369
13.96.1.1 QuaZip_getFileInfo . . . 1369
13.96.1.2 QuaZip_getFileInfo . . . 1369
13.96.1.3 QuaZip_getFileInfo . . . 1370
13.96.1.4 QuaZip_getFileInfo . . . 1370
13.97 src/ckb/quazip/quazip.h File Reference . . . 1370
13.97.1 Macro Definition Documentation . . . 1371
13.97.1.1 UNZ_OPENERROR . . . 1371
13.98 src/ckb/quazip/quazip_global.h File Reference . . . 1371
13.98.1 Macro Definition Documentation . . . 1372
13.98.1.1 QUAZIP_EXPORT . . . 1372
13.98.1.2 QUAZIP_EXTRA_NTFS_MAGIC . . . 1372
13.98.1.3 QUAZIP_EXTRA_NTFS_TIME_MAGIC . . . 1372
13.98.1.4 UNUSED . . . 1373
13.99 src/ckb/quazip/quazipdir.cpp File Reference . . . 1373
13.99.1 Function Documentation . . . 1373
13.99.1.1 QuaZipDir_convertInfoList . . . 1373
13.99.1.2 QuaZipDir_convertInfoList . . . 1373
13.99.1.3 QuaZipDir_convertInfoList . . . 1374
13.99.1.4 QuaZipDir_getFileInfo . . . 1374
13.100 src/ckb/quazip/quazipdir.h File Reference . . . 1375
13.101 src/ckb/quazip/quazipfile.cpp File Reference . . . 1375
13.102 src/ckb/quazip/quazipfile.h File Reference . . . 1376
13.103 src/ckb/quazip/quazipfileinfo.cpp File Reference . . . 1377
13.103.1 Function Documentation . . . 1377
13.103.1.1 getNTFSTime . . . 1378
13.103.1.2 permissionsFromExternalAttr . . . 1379
13.104 src/ckb/quazip/quazipfileinfo.h File Reference . . . 1379
13.105 src/ckb/quazip/quazipnewinfo.cpp File Reference . . . 1380
13.105.1 Function Documentation . . . 1381
13.105.1.1 QuaZipNewInfo_init . . . 1381
13.105.1.2 QuaZipNewInfo_setPermissions . . . 1382
13.105.1.3 setNTFSTime . . . 1382
13.106 src/ckb/quazip/quazipnewinfo.h File Reference . . . 1384
13.107 src/ckb/quazip/unzip.c File Reference . . . 1385
13.107.1 Data Structure Documentation . . . 1388
13.107.1.1 struct unz_file_info64_internal_s . . . 1388
13.107.1.2 struct file_in_zip64_read_info_s . . . 1389

13.107.1.3 struct unz64_s . . . 1389
13.107.2 Macro Definition Documentation . . . 1390
13.107.2.1 ALLOC . . . 1390
13.107.2.2 BUFREADCOMMENT . . . 1391
13.107.2.3 CASESENSITIVITYDEFAULT_NO . . . 1391
13.107.2.4 CASESENSITIVITYDEFAULTVALUE . . . 1391
13.107.2.5 local . . . 1391
13.107.2.6 SIZECENTRALDIRITEM . . . 1391
13.107.2.7 SIZEZIPLOCALHEADER . . . 1391
13.107.2.8 STRCMPCASENOSENTIVEFUNCTION . . . 1391
13.107.2.9 TRYFREE . . . 1391
13.107.2.10 UNZ_BUFSIZE . . . 1391
13.107.2.11 UNZ_MAXFILENAMEINZIP . . . 1392
13.107.3 Typedef Documentation . . . 1392
13.107.3.1 unz_file_info64_internal . . . 1392
13.107.3.2 z_crc_t . . . 1392
13.107.4 Function Documentation . . . 1392
13.107.4.1 _Z_OF . . . 1392
13.107.4.2 _Z_OF . . . 1392
13.107.4.3 _Z_OF . . . 1392
13.107.4.4 _Z_OF . . . 1392
13.107.4.5 _Z_OF . . . 1392
13.107.4.6 strcmpcasenosensitive_internal . . . 1392
13.107.4.7 unz64local_CheckCurrentFileCoherencyHeader . . . 1392
13.107.4.8 unz64local_DosDateToTmuDate . . . 1394
13.107.4.9 unz64local_getByte . . . 1395
13.107.4.10 unz64local_GetCurrentFileInfoInternal . . . 1396
13.107.4.11 unz64local_getLong . . . 1399
13.107.4.12 unz64local_getLong64 . . . 1400
13.107.4.13 unz64local_getShort . . . 1402
13.107.4.14 unz64local_SearchCentralDir . . . 1403
13.107.4.15 unz64local_SearchCentralDir64 . . . 1404
13.107.4.16 unzClearFlags . . . 1405
13.107.4.17 unzClose . . . 1406
13.107.4.18 unzCloseCurrentFile . . . 1407
13.107.4.19 unzeof . . . 1408
13.107.4.20 unzGetCurrentFileInfo . . . 1408
13.107.4.21 unzGetCurrentFileInfo64 . . . 1409
13.107.4.22 unzGetCurrentFileZStreamPos64 . . . 1410
13.107.4.23 unzGetFilePos . . . 1410

13.107.4.24 unzGetFilePos64 . . . 1411
13.107.4.25 unzGetGlobalComment . . . 1411
13.107.4.26 unzGetGlobalInfo . . . 1412
13.107.4.27 unzGetGlobalInfo64 . . . 1412
13.107.4.28 unzGetLocalExtrafield . . . 1413
13.107.4.29 unzGetOffset . . . 1414
13.107.4.30 unzGetOffset64 . . . 1414
13.107.4.31 unzGoToFilePos . . . 1415
13.107.4.32 unzGoToFilePos64 . . . 1415
13.107.4.33 unzGoToFirstFile . . . 1416
13.107.4.34 unzGoToNextFile . . . 1417
13.107.4.35 unzLocateFile . . . 1418
13.107.4.36 unzOpen . . . 1419
13.107.4.37 unzOpen2 . . . 1419
13.107.4.38 unzOpen2_64 . . . 1420
13.107.4.39 unzOpen64 . . . 1421
13.107.4.40 unzOpenCurrentFile . . . 1422
13.107.4.41 unzOpenCurrentFile2 . . . 1422
13.107.4.42 unzOpenCurrentFile3 . . . 1422
13.107.4.43 unzOpenCurrentFilePassword . . . 1425
13.107.4.44 unzOpenInternal . . . 1426
13.107.4.45 unzReadCurrentFile . . . 1429
13.107.4.46 unzSetFlags . . . 1432
13.107.4.47 unzSetOffset . . . 1433
13.107.4.48 unzSetOffset64 . . . 1433
13.107.4.49 unzStringFileNameCompare . . . 1434
13.107.4.50 unztell . . . 1434
13.107.4.51 unztell64 . . . 1435
13.107.5 Variable Documentation . . . 1435
13.107.5.1 unz_copyright . . . 1435
13.108 src/ckb/quazip/unzip.h File Reference . . . 1436
13.108.1 Data Structure Documentation . . . 1438
13.108.1.1 struct tm_unz_s . . . 1438
13.108.1.2 struct unz_global_info64_s . . . 1438
13.108.1.3 struct unz_global_info_s . . . 1439
13.108.1.4 struct unz_file_info64_s . . . 1440
13.108.1.5 struct unz_file_info_s . . . 1441
13.108.1.6 struct unz_file_pos_s . . . 1443
13.108.1.7 struct unz64_file_pos_s . . . 1444
13.108.2 Macro Definition Documentation . . . 1444

13.108.2.1 UNZ_AUTO_CLOSE . . . 1444
13.108.2.2 UNZ_BADZIPFILE . . . 1444
13.108.2.3 UNZ_CRCERROR . . . 1444
13.108.2.4 UNZ_DEFAULT_FLAGS . . . 1445
13.108.2.5 UNZ_END_OF_LIST_OF_FILE . . . 1445
13.108.2.6 UNZ_EOF . . . 1445
13.108.2.7 UNZ_ERRNO . . . 1445
13.108.2.8 UNZ_INTERNALERROR . . . 1445
13.108.2.9 UNZ_OK . . . 1445
13.108.2.10 UNZ_PARAMERROR . . . 1445
13.108.2.11 Z_BZIP2ED . . . 1446
13.108.3 Typedef Documentation . . . 1446
13.108.3.1 tm_unz . . . 1446
13.108.3.2 unz64_file_pos . . . 1446
13.108.3.3 unz_file_info . . . 1446
13.108.3.4 unz_file_info64 . . . 1446
13.108.3.5 unz_file_pos . . . 1446
13.108.3.6 unz_global_info . . . 1446
13.108.3.7 unz_global_info64 . . . 1446
13.108.3.8 unzFile . . . 1446
13.108.4 Function Documentation . . . 1446
13.108.4.1 _Z_OF . . . 1446
13.108.4.2 _Z_OF . . . 1446
13.108.4.3 _Z_OF . . . 1446
13.108.4.4 _Z_OF . . . 1446
13.108.4.5 _Z_OF . . . 1446
13.108.4.6 _Z_OF . . . 1446
13.108.4.7 _Z_OF . . . 1446
13.108.4.8 _Z_OF . . . 1446
13.108.4.9 _Z_OF . . . 1446
13.108.4.10 _Z_OF . . . 1446
13.108.4.11 _Z_OF . . . 1447
13.108.4.12 _Z_OF . . . 1447
13.108.4.13 _Z_OF . . . 1447
13.108.4.14 _Z_OF . . . 1447
13.108.4.15 _Z_OF . . . 1447
13.108.4.16 unzClearFlags . . . 1447
13.108.4.17 unzGetFilePos . . . 1447
13.108.4.18 unzGetFilePos64 . . . 1448
13.108.4.19 unzGetOffset . . . 1448

13.108.4.20 unzGetOffset64 . . . 1449
13.108.4.21 unzGoToFilePos . . . 1449
13.108.4.22 unzGoToFilePos64 . . . 1450
13.108.4.23 unzOpenInternal . . . 1451
13.108.4.24 unzSetFlags . . . 1454
13.108.4.25 unzSetOffset . . . 1455
13.108.4.26 unzSetOffset64 . . . 1455
13.109 src/ckb/quazip/zip.c File Reference . . . 1456
13.109.1 Data Structure Documentation . . . 1459
13.109.1.1 struct linkedlist_datablock_internal_s . . . 1459
13.109.1.2 struct linkedlist_data_s . . . 1460
13.109.1.3 struct curfile64_info . . . 1461
13.109.1.4 struct zip64_internal . . . 1462
13.109.2 Macro Definition Documentation . . . 1462
13.109.2.1 ALLOC . . . 1463
13.109.2.2 BUFREADCOMMENT . . . 1463
13.109.2.3 CENTRALHEADERMAGIC . . . 1463
13.109.2.4 CRC_LOCALHEADER_OFFSET . . . 1463
13.109.2.5 DESCRIPTORHEADERMAGIC . . . 1463
13.109.2.6 ENDHEADERMAGIC . . . 1463
13.109.2.7 FLAG_LOCALHEADER_OFFSET . . . 1463
13.109.2.8 INCLUDECRYPTINGCODE_IFCRYPTALLOWED . . . 1463
13.109.2.9 local . . . 1463
13.109.2.10 LOCALHEADERMAGIC . . . 1463
13.109.2.11 MAKEULONG64 . . . 1464
13.109.2.12 SEEK_CUR . . . 1464
13.109.2.13 SEEK_END . . . 1464
13.109.2.14 SEEK_SET . . . 1464
13.109.2.15 SIZECENTRALHEADER . . . 1464
13.109.2.16 SIZEDATA_INDATABLOCK . . . 1464
13.109.2.17 TRYFREE . . . 1464
13.109.2.18 VERSIONMADEBY . . . 1464
13.109.2.19 WRITEBUFFERSIZE . . . 1464
13.109.2.20 Z_MAXFILENAMEINZIP . . . 1464
13.109.2.21 ZIP64ENDHEADERMAGIC . . . 1465
13.109.2.22 ZIP64ENDLOCHEADERMAGIC . . . 1465
13.109.3 Typedef Documentation . . . 1465
13.109.3.1 linkedlist_data . . . 1465
13.109.3.2 linkedlist_datablock_internal . . . 1465
13.109.3.3 z_crc_t . . . 1465

13.109.4 Function Documentation . . . 1465
13.109.4.1 _Z_OF . . . 1465
13.109.4.2 _Z_OF . . . 1465
13.109.4.3 _Z_OF . . . 1465
13.109.4.4 _Z_OF . . . 1465
13.109.4.5 _Z_OF . . . 1465
13.109.4.6 _Z_OF . . . 1465
13.109.4.7 add_data_in_datablock . . . 1465
13.109.4.8 allocate_new_datablock . . . 1467
13.109.4.9 free_datablock . . . 1467
13.109.4.10 free_linkedlist . . . 1468
13.109.4.11 init_linkedlist . . . 1468
13.109.4.12 LoadCentralDirectoryRecord . . . 1469
13.109.4.13 Write_EndOfCentralDirectoryRecord . . . 1472
13.109.4.14 Write_GlobalComment . . . 1474
13.109.4.15 Write_LocalFileHeader . . . 1474
13.109.4.16 Write_Zip64EndOfCentralDirectoryLocator . . . 1476
13.109.4.17 Write_Zip64EndOfCentralDirectoryRecord . . . 1477
13.109.4.18 zip64FlushWriteBuffer . . . 1479
13.109.4.19 zip64local_getByte . . . 1479
13.109.4.20 zip64local_getLong . . . 1480
13.109.4.21 zip64local_getLong64 . . . 1481
13.109.4.22 zip64local_getShort . . . 1482
13.109.4.23 zip64local_putValue . . . 1483
13.109.4.24 zip64local_putValue_inmemory . . . 1484
13.109.4.25 zip64local_SearchCentralDir . . . 1484
13.109.4.26 zip64local_SearchCentralDir64 . . . 1485
13.109.4.27 zip64local_TmzDateToDosDate . . . 1487
13.109.4.28 zipClearFlags . . . 1488
13.109.4.29 zipClose . . . 1488
13.109.4.30 zipCloseFileInZip . . . 1490
13.109.4.31 zipCloseFileInZipRaw . . . 1491
13.109.4.32 zipCloseFileInZipRaw64 . . . 1492
13.109.4.33 zipOpen . . . 1496
13.109.4.34 zipOpen2 . . . 1496
13.109.4.35 zipOpen2_64 . . . 1497
13.109.4.36 zipOpen3 . . . 1498
13.109.4.37 zipOpen64 . . . 1500
13.109.4.38 zipOpenNewFileInZip . . . 1501
13.109.4.39 zipOpenNewFileInZip2 . . . 1501

13.109.4.40 zipOpenNewFileInZip2_64 . . . 1502
13.109.4.41 zipOpenNewFileInZip3 . . . 1502
13.109.4.42 zipOpenNewFileInZip3_64 . . . 1503
13.109.4.43 zipOpenNewFileInZip4 . . . 1503
13.109.4.44 zipOpenNewFileInZip4_64 . . . 1504
13.109.4.45 zipOpenNewFileInZip64 . . . 1508
13.109.4.46 zipRemoveExtraInfoBlock . . . 1508
13.109.4.47 zipSetFlags . . . 1509
13.109.4.48 zipWriteInFileInZip . . . 1510
13.109.5 Variable Documentation . . . 1512
13.109.5.1 zip_copyright . . . 1512
13.110 src/ckb/quazip/zip.h File Reference . . . 1512
13.110.1 Data Structure Documentation . . . 1514
13.110.1.1 struct tm_zip_s . . . 1514
13.110.1.2 struct zip_fileinfo . . . 1515
13.110.2 Macro Definition Documentation . . . 1516
13.110.2.1 APPEND_STATUS_ADDINZIP . . . 1516
13.110.2.2 APPEND_STATUS_CREATE . . . 1516
13.110.2.3 APPEND_STATUS_CREATEAFTER . . . 1517
13.110.2.4 DEF_MEM_LEVEL . . . 1517
13.110.2.5 Z_BZIP2ED . . . 1517
13.110.2.6 ZIP_AUTO_CLOSE . . . 1517
13.110.2.7 ZIP_BADZIPFILE . . . 1517
13.110.2.8 ZIP_DEFAULT_FLAGS . . . 1517
13.110.2.9 ZIP_EOF . . . 1517
13.110.2.10 ZIP_ERRNO . . . 1517
13.110.2.11 ZIP_INTERNALERROR . . . 1517
13.110.2.12 ZIP_OK . . . 1518
13.110.2.13 ZIP_PARAMERROR . . . 1518
13.110.2.14 ZIP_SEQUENTIAL . . . 1518
13.110.2.15 ZIP_WRITE_DATA_DESCRIPTOR . . . 1518
13.110.3 Typedef Documentation . . . 1518
13.110.3.1 tm_zip . . . 1518
13.110.3.2 zipcharpc . . . 1518
13.110.3.3 zipFile . . . 1518
13.110.4 Function Documentation . . . 1518
13.110.4.1 _Z_OF . . . 1518
13.110.4.2 _Z_OF . . . 1518
13.110.4.3 _Z_OF . . . 1518
13.110.4.4 _Z_OF . . . 1519

13.110.4.5 _Z_OF . . . 1519
13.110.4.6 _Z_OF . . . 1519
13.110.4.7 _Z_OF . . . 1519
13.110.4.8 _Z_OF . . . 1519
13.110.4.9 _Z_OF . . . 1519
13.110.4.10 _Z_OF . . . 1519
13.110.4.11 _Z_OF . . . 1519
13.110.4.12 _Z_OF . . . 1519
13.110.4.13 _Z_OF . . . 1519
13.110.4.14 _Z_OF . . . 1519
13.110.4.15 _Z_OF . . . 1519
13.110.4.16 _Z_OF . . . 1519
13.110.4.17 _Z_OF . . . 1519
13.110.4.18 dpClearFlags . . . 1519
13.110.4.19 dpOpen3 . . . 1520
13.110.4.20 dpSetFlags . . . 1522
13.111 src/ckb/rebindwidget.cpp File Reference . . . 1523
13.111.1 Variable Documentation . . . 1523
13.111.1.1 DPI_CUST_IDX . . . 1523
13.111.1.2 DPI_OFFSET . . . 1523
13.112 src/ckb/rebindwidget.h File Reference . . . 1523
13.113 src/ckb/release/moc_animadddialog.cpp File Reference . . . 1524
13.113.1 Data Structure Documentation . . . 1525
13.113.1.1 struct qt_meta_stringdata_AnimAddDialog_t . . . 1525
13.113.2 Macro Definition Documentation . . . 1526
13.113.2.1 QT_MOC_LITERAL . . . 1526
13.113.3 Variable Documentation . . . 1526
13.113.3.1 qt_meta_data_AnimAddDialog . . . 1526
13.113.3.2 qt_meta_stringdata_AnimAddDialog . . . 1527
13.114 src/ckb/release/moc_animscript.cpp File Reference . . . 1527
13.114.1 Data Structure Documentation . . . 1528
13.114.1.1 struct qt_meta_stringdata_AnimScript_t . . . 1528
13.114.2 Macro Definition Documentation . . . 1529
13.114.2.1 QT_MOC_LITERAL . . . 1529
13.114.3 Variable Documentation . . . 1529
13.114.3.1 qt_meta_data_AnimScript . . . 1529
13.114.3.2 qt_meta_stringdata_AnimScript . . . 1530
13.115 src/ckb/release/moc_animsettingdialog.cpp File Reference . . . 1530
13.115.1 Data Structure Documentation . . . 1531
13.115.1.1 struct qt_meta_stringdata_AnimSettingDialog_t . . . 1531

13.115.2 Macro Definition Documentation . . . 1531
13.115.2.1 QT_MOC_LITERAL . . . 1531
13.115.3 Variable Documentation . . . 1532
13.115.3.1 qt_meta_data_AnimSettingDialog . . . 1532
13.115.3.2 qt_meta_stringdata_AnimSettingDialog . . . 1532
13.116 src/ckb/release/moc_ckbsettingswriter.cpp File Reference . . . 1533
13.116.1 Data Structure Documentation . . . 1533
13.116.1.1 struct qt_meta_stringdata_CkbSettingsWriter_t . . . 1533
13.116.2 Macro Definition Documentation . . . 1534
13.116.2.1 QT_MOC_LITERAL . . . 1534
13.116.3 Variable Documentation . . . 1534
13.116.3.1 qt_meta_data_CkbSettingsWriter . . . 1534
13.116.3.2 qt_meta_stringdata_CkbSettingsWriter . . . 1535
13.117 src/ckb/release/moc_colorbutton.cpp File Reference . . . 1535
13.117.1 Data Structure Documentation . . . 1536
13.117.1.1 struct qt_meta_stringdata_ColorButton_t . . . 1536
13.117.2 Macro Definition Documentation . . . 1536
13.117.2.1 QT_MOC_LITERAL . . . 1536
13.117.3 Variable Documentation . . . 1537
13.117.3.1 qt_meta_data_ColorButton . . . 1537
13.117.3.2 qt_meta_stringdata_ColorButton . . . 1537
13.118 src/ckb/release/moc_extrasettingswidget.cpp File Reference . . . 1537
13.118.1 Data Structure Documentation . . . 1538
13.118.1.1 struct qt_meta_stringdata_ExtraSettingsWidget_t . . . 1538
13.118.2 Macro Definition Documentation . . . 1539
13.118.2.1 QT_MOC_LITERAL . . . 1539
13.118.3 Variable Documentation . . . 1539
13.118.3.1 qt_meta_data_ExtraSettingsWidget . . . 1539
13.118.3.2 qt_meta_stringdata_ExtraSettingsWidget . . . 1539
13.119 src/ckb/release/moc_fwupgradedialog.cpp File Reference . . . 1540
13.119.1 Data Structure Documentation . . . 1541
13.119.1.1 struct qt_meta_stringdata_FwUpgradeDialog_t . . . 1541
13.119.2 Macro Definition Documentation . . . 1542
13.119.2.1 QT_MOC_LITERAL . . . 1542
13.119.3 Variable Documentation . . . 1542
13.119.3.1 qt_meta_data_FwUpgradeDialog . . . 1542
13.119.3.2 qt_meta_stringdata_FwUpgradeDialog . . . 1543
13.120 src/ckb/release/moc_gradientbutton.cpp File Reference . . . 1543
13.120.1 Data Structure Documentation . . . 1544
13.120.1.1 struct qt_meta_stringdata_GradientButton_t . . . 1544

13.120.2 Macro Definition Documentation . . . 1545
13.120.2.1 QT_MOC_LITERAL . . . 1545
13.120.3 Variable Documentation . . . 1545
13.120.3.1 qt_meta_data_GradientButton . . . 1545
13.120.3.2 qt_meta_stringdata_GradientButton . . . 1546
13.121 src/ckb/release/moc_gradientdialog.cpp File Reference . . . 1546
13.121.1 Data Structure Documentation . . . 1547
13.121.1.1 struct qt_meta_stringdata_GradientDialog_t . . . 1547
13.121.2 Macro Definition Documentation . . . 1547
13.121.2.1 QT_MOC_LITERAL . . . 1547
13.121.3 Variable Documentation . . . 1548
13.121.3.1 qt_meta_data_GradientDialog . . . 1548
13.121.3.2 qt_meta_stringdata_GradientDialog . . . 1548
13.122 src/ckb/release/moc_gradientdialogwidget.cpp File Reference . . . 1549
13.122.1 Data Structure Documentation . . . 1549
13.122.1.1 struct qt_meta_stringdata_GradientDialogWidget_t . . . 1549
13.122.2 Macro Definition Documentation . . . 1550
13.122.2.1 QT_MOC_LITERAL . . . 1550
13.122.3 Variable Documentation . . . 1550
13.122.3.1 qt_meta_data_GradientDialogWidget . . . 1550
13.122.3.2 qt_meta_stringdata_GradientDialogWidget . . . 1551
13.123 src/ckb/release/moc_kb.cpp File Reference . . . 1551
13.123.1 Data Structure Documentation . . . 1552
13.123.1.1 struct qt_meta_stringdata_Kb_t . . . 1552
13.123.2 Macro Definition Documentation . . . 1553
13.123.2.1 QT_MOC_LITERAL . . . 1553
13.123.3 Variable Documentation . . . 1553
13.123.3.1 qt_meta_data_Kb . . . 1553
13.123.3.2 qt_meta_stringdata_Kb . . . 1553
13.124 src/ckb/release/moc_kbanim.cpp File Reference . . . 1554
13.124.1 Data Structure Documentation . . . 1554
13.124.1.1 struct qt_meta_stringdata_KbAnim_t . . . 1554
13.124.2 Macro Definition Documentation . . . 1555
13.124.2.1 QT_MOC_LITERAL . . . 1555
13.124.3 Variable Documentation . . . 1555
13.124.3.1 qt_meta_data_KbAnim . . . 1555
13.124.3.2 qt_meta_stringdata_KbAnim . . . 1556
13.125 src/ckb/release/moc_kbanimwidget.cpp File Reference . . . 1556
13.125.1 Data Structure Documentation . . . 1557
13.125.1.1 struct qt_meta_stringdata_KbAnimWidget_t . . . 1557

13.125.2 Macro Definition Documentation . . . 1558
13.125.2.1 QT_MOC_LITERAL . . . 1558
13.125.3 Variable Documentation . . . 1558
13.125.3.1 qt_meta_data_KbAnimWidget . . . 1558
13.125.3.2 qt_meta_stringdata_KbAnimWidget . . . 1558
13.126 src/ckb/release/moc_kbbind.cpp File Reference . . . 1559
13.126.1 Data Structure Documentation . . . 1559
13.126.1.1 struct qt_meta_stringdata_KbBind_t . . . 1559
13.126.2 Macro Definition Documentation . . . 1560
13.126.2.1 QT_MOC_LITERAL . . . 1560
13.126.3 Variable Documentation . . . 1560
13.126.3.1 qt_meta_data_KbBind . . . 1560
13.126.3.2 qt_meta_stringdata_KbBind . . . 1560
13.127 src/ckb/release/moc_kbbindwidget.cpp File Reference . . . 1561
13.127.1 Data Structure Documentation . . . 1562
13.127.1.1 struct qt_meta_stringdata_KbBindWidget_t . . . 1562
13.127.2 Macro Definition Documentation . . . 1562
13.127.2.1 QT_MOC_LITERAL . . . 1562
13.127.3 Variable Documentation . . . 1563
13.127.3.1 qt_meta_data_KbBindWidget . . . 1563
13.127.3.2 qt_meta_stringdata_KbBindWidget . . . 1563
13.128 src/ckb/release/moc_kbfirmware.cpp File Reference . . . 1563
13.128.1 Data Structure Documentation . . . 1564
13.128.1.1 struct qt_meta_stringdata_KbFirmware_t . . . 1564
13.128.2 Macro Definition Documentation . . . 1565
13.128.2.1 QT_MOC_LITERAL . . . 1565
13.128.3 Variable Documentation . . . 1565
13.128.3.1 qt_meta_data_KbFirmware . . . 1565
13.128.3.2 qt_meta_stringdata_KbFirmware . . . 1566
13.129 src/ckb/release/moc_kblight.cpp File Reference . . . 1566
13.129.1 Data Structure Documentation . . . 1567
13.129.1.1 struct qt_meta_stringdata_KbLight_t . . . 1567
13.129.2 Macro Definition Documentation . . . 1568
13.129.2.1 QT_MOC_LITERAL . . . 1568
13.129.3 Variable Documentation . . . 1568
13.129.3.1 qt_meta_data_KbLight . . . 1568
13.129.3.2 qt_meta_stringdata_KbLight . . . 1569
13.130 src/ckb/release/moc_kblightwidget.cpp File Reference . . . 1569
13.130.1 Data Structure Documentation . . . 1570
13.130.1.1 struct qt_meta_stringdata_KbLightWidget_t . . . 1570

13.130.2 Macro Definition Documentation . . . 1571
13.130.2.1 QT_MOC_LITERAL . . . 1571
13.130.3 Variable Documentation . . . 1571
13.130.3.1 qt_meta_data_KbLightWidget . . . 1571
13.130.3.2 qt_meta_stringdata_KbLightWidget . . . 1571
13.131 src/ckb/release/moc_kbmanager.cpp File Reference . . . 1572
13.131.1 Data Structure Documentation . . . 1572
13.131.1.1 struct qt_meta_stringdata_KbManager_t . . . 1572
13.131.2 Macro Definition Documentation . . . 1573
13.131.2.1 QT_MOC_LITERAL . . . 1573
13.131.3 Variable Documentation . . . 1573
13.131.3.1 qt_meta_data_KbManager . . . 1573
13.131.3.2 qt_meta_stringdata_KbManager . . . 1573
13.132 src/ckb/release/moc_kbmode.cpp File Reference . . . 1574
13.132.1 Data Structure Documentation . . . 1575
13.132.1.1 struct qt_meta_stringdata_KbMode_t . . . 1575
13.132.2 Macro Definition Documentation . . . 1575
13.132.2.1 QT_MOC_LITERAL . . . 1575
13.132.3 Variable Documentation . . . 1576
13.132.3.1 qt_meta_data_KbMode . . . 1576
13.132.3.2 qt_meta_stringdata_KbMode . . . 1576
13.133 src/ckb/release/moc_kbperf.cpp File Reference . . . 1576
13.133.1 Data Structure Documentation . . . 1577
13.133.1.1 struct qt_meta_stringdata_KbPerf_t . . . 1577
13.133.2 Macro Definition Documentation . . . 1578
13.133.2.1 QT_MOC_LITERAL . . . 1578
13.133.3 Variable Documentation . . . 1578
13.133.3.1 qt_meta_data_KbPerf . . . 1578
13.133.3.2 qt_meta_stringdata_KbPerf . . . 1579
13.134 src/ckb/release/moc_kbprofile.cpp File Reference . . . 1579
13.134.1 Data Structure Documentation . . . 1580
13.134.1.1 struct qt_meta_stringdata_KbProfile_t . . . 1580
13.134.2 Macro Definition Documentation . . . 1580
13.134.2.1 QT_MOC_LITERAL . . . 1580
13.134.3 Variable Documentation . . . 1581
13.134.3.1 qt_meta_data_KbProfile . . . 1581
13.134.3.2 qt_meta_stringdata_KbProfile . . . 1581
13.135 src/ckb/release/moc_kbprofiledialog.cpp File Reference . . . 1581
13.135.1 Data Structure Documentation . . . 1582
13.135.1.1 struct qt_meta_stringdata_KbProfileDialog_t . . . 1582

13.135.2 Macro Definition Documentation . . . 1583
13.135.2.1 QT_MOC_LITERAL . . . 1583
13.135.3 Variable Documentation . . . 1583
13.135.3.1 qt_meta_data_KbProfileDialog . . . 1583
13.135.3.2 qt_meta_stringdata_KbProfileDialog . . . 1584
13.136 src/ckb/release/moc_kbwidget.cpp File Reference . . . 1585
13.136.1 Data Structure Documentation . . . 1585
13.136.1.1 struct qt_meta_stringdata_KbWidget_t . . . 1585
13.136.2 Macro Definition Documentation . . . 1586
13.136.2.1 QT_MOC_LITERAL . . . 1586
13.136.3 Variable Documentation . . . 1586
13.136.3.1 qt_meta_data_KbWidget . . . 1586
13.136.3.2 qt_meta_stringdata_KbWidget . . . 1586
13.137 src/ckb/release/moc_keyaction.cpp File Reference . . . 1587
13.137.1 Data Structure Documentation . . . 1587
13.137.1.1 struct qt_meta_stringdata_KeyAction_t . . . 1587
13.137.2 Macro Definition Documentation . . . 1588
13.137.2.1 QT_MOC_LITERAL . . . 1588
13.137.3 Variable Documentation . . . 1588
13.137.3.1 qt_meta_data_KeyAction . . . 1588
13.137.3.2 qt_meta_stringdata_KeyAction . . . 1589
13.138 src/ckb/release/moc_keywidget.cpp File Reference . . . 1589
13.138.1 Data Structure Documentation . . . 1590
13.138.1.1 struct qt_meta_stringdata_KeyWidget_t . . . 1590
13.138.2 Macro Definition Documentation . . . 1590
13.138.2.1 QT_MOC_LITERAL . . . 1590
13.138.3 Variable Documentation . . . 1591
13.138.3.1 qt_meta_data_KeyWidget . . . 1591
13.138.3.2 qt_meta_stringdata_KeyWidget . . . 1591
13.139 src/ckb/release/moc_kperfwidget.cpp File Reference . . . 1591
13.139.1 Data Structure Documentation . . . 1592
13.139.1.1 struct qt_meta_stringdata_KPerfWidget_t . . . 1592
13.139.2 Macro Definition Documentation . . . 1593
13.139.2.1 QT_MOC_LITERAL . . . 1593
13.139.3 Variable Documentation . . . 1593
13.139.3.1 qt_meta_data_KPerfWidget . . . 1593
13.139.3.2 qt_meta_stringdata_KPerfWidget . . . 1594
13.140 src/ckb/release/moc_layoutdialog.cpp File Reference . . . 1594
13.140.1 Data Structure Documentation . . . 1595
13.140.1.1 struct qt_meta_stringdata_LayoutDialog_t . . . 1595

13.140.2 Macro Definition Documentation . . . 1596
13.140.2.1 QT_MOC_LITERAL . . . 1596
13.140.3 Variable Documentation . . . 1596
13.140.3.1 qt_meta_data_LayoutDialog . . . 1596
13.140.3.2 qt_meta_stringdata_LayoutDialog . . . 1597
13.141 src/ckb/release/moc_macroreader.cpp File Reference . . . 1597
13.141.1 Data Structure Documentation . . . 1598
13.141.1.1 struct qt_meta_stringdata_MacroReaderThread_t . . . 1598
13.141.1.2 struct qt_meta_stringdata_MacroReader_t . . . 1598
13.141.2 Macro Definition Documentation . . . 1599
13.141.2.1 QT_MOC_LITERAL . . . 1599
13.141.2.2 QT_MOC_LITERAL . . . 1599
13.141.3 Variable Documentation . . . 1599
13.141.3.1 qt_meta_data_MacroReader . . . 1600
13.141.3.2 qt_meta_data_MacroReaderThread . . . 1600
13.141.3.3 qt_meta_stringdata_MacroReader . . . 1600
13.141.3.4 qt_meta_stringdata_MacroReaderThread . . . 1600
13.142 src/ckb/release/moc_mainwindow.cpp File Reference . . . 1601
13.142.1 Data Structure Documentation . . . 1602
13.142.1.1 struct qt_meta_stringdata_MainWindow_t . . . 1602
13.142.2 Macro Definition Documentation . . . 1602
13.142.2.1 QT_MOC_LITERAL . . . 1602
13.142.3 Variable Documentation . . . 1602
13.142.3.1 qt_meta_data_MainWindow . . . 1602
13.142.3.2 qt_meta_stringdata_MainWindow . . . 1603
13.143 src/ckb/release/moc_modeselectdialog.cpp File Reference . . . 1603
13.143.1 Data Structure Documentation . . . 1603
13.143.1.1 struct qt_meta_stringdata_ModeSelectDialog_t . . . 1603
13.143.2 Macro Definition Documentation . . . 1604
13.143.2.1 QT_MOC_LITERAL . . . 1604
13.143.3 Variable Documentation . . . 1604
13.143.3.1 qt_meta_data_ModeSelectDialog . . . 1604
13.143.3.2 qt_meta_stringdata_ModeSelectDialog . . . 1605
13.144 src/ckb/release/moc_mperfwidget.cpp File Reference . . . 1605
13.144.1 Data Structure Documentation . . . 1606
13.144.1.1 struct qt_meta_stringdata_MPerfWidget_t . . . 1606
13.144.2 Macro Definition Documentation . . . 1606
13.144.2.1 QT_MOC_LITERAL . . . 1606
13.144.3 Variable Documentation . . . 1607
13.144.3.1 qt_meta_data_MPerfWidget . . . 1607

13.144.3.2 qt_meta_stringdata_MPerfWidget . . . 1607
13.145 src/ckb/release/moc_quagzipfile.cpp File Reference . . . 1608
13.145.1 Data Structure Documentation . . . 1608
13.145.1.1 struct qt_meta_stringdata_QuaGzipFile_t . . . 1608
13.145.2 Macro Definition Documentation . . . 1609
13.145.2.1 QT_MOC_LITERAL . . . 1609
13.145.3 Variable Documentation . . . 1609
13.145.3.1 qt_meta_data_QuaGzipFile . . . 1609
13.145.3.2 qt_meta_stringdata_QuaGzipFile . . . 1610
13.146 src/ckb/release/moc_quaziodevice.cpp File Reference . . . 1610
13.146.1 Data Structure Documentation . . . 1611
13.146.1.1 struct qt_meta_stringdata_QuaZIODevice_t . . . 1611
13.146.2 Macro Definition Documentation . . . 1611
13.146.2.1 QT_MOC_LITERAL . . . 1611
13.146.3 Variable Documentation . . . 1612
13.146.3.1 qt_meta_data_QuaZIODevice . . . 1612
13.146.3.2 qt_meta_stringdata_QuaZIODevice . . . 1612
13.147 src/ckb/release/moc_quazipfile.cpp File Reference . . . 1612
13.147.1 Data Structure Documentation . . . 1613
13.147.1.1 struct qt_meta_stringdata_QuaZipFile_t . . . 1613
13.147.2 Macro Definition Documentation . . . 1613
13.147.2.1 QT_MOC_LITERAL . . . 1613
13.147.3 Variable Documentation . . . 1614
13.147.3.1 qt_meta_data_QuaZipFile . . . 1614
13.147.3.2 qt_meta_stringdata_QuaZipFile . . . 1614
13.148 src/ckb/release/moc_rebindwidget.cpp File Reference . . . 1614
13.148.1 Data Structure Documentation . . . 1615
13.148.1.1 struct qt_meta_stringdata_RebindWidget_t . . . 1615
13.148.2 Macro Definition Documentation . . . 1616
13.148.2.1 QT_MOC_LITERAL . . . 1616
13.148.3 Variable Documentation . . . 1616
13.148.3.1 qt_meta_data_RebindWidget . . . 1616
13.148.3.2 qt_meta_stringdata_RebindWidget . . . 1616
13.149 src/ckb/release/moc_rlistwidget.cpp File Reference . . . 1617
13.149.1 Data Structure Documentation . . . 1617
13.149.1.1 struct qt_meta_stringdata_RListWidget_t . . . 1617
13.149.2 Macro Definition Documentation . . . 1618
13.149.2.1 QT_MOC_LITERAL . . . 1618
13.149.3 Variable Documentation . . . 1618
13.149.3.1 qt_meta_data_RListWidget . . . 1618

13.149.3.2 qt_meta_stringdata_RListWidget . . . 1618
13.150 src/ckb/release/moc_settingswidget.cpp File Reference . . . 1619
13.150.1 Data Structure Documentation . . . 1620
13.150.1.1 struct qt_meta_stringdata_SettingsWidget_t . . . 1620
13.150.2 Macro Definition Documentation . . . 1620
13.150.2.1 QT_MOC_LITERAL . . . 1620
13.150.3 Variable Documentation . . . 1621
13.150.3.1 qt_meta_data_SettingsWidget . . . 1621
13.150.3.2 qt_meta_stringdata_SettingsWidget . . . 1621
13.151 src/ckb/release/qrc_binary.cpp File Reference . . . 1621
13.151.1 Macro Definition Documentation . . . 1622
13.151.1.1 QT_RCC_MANGLE_NAMESPACE . . . 1622
13.151.1.2 QT_RCC_PREPEND_NAMESPACE . . . 1622
13.151.2 Function Documentation . . . 1622
13.151.2.1 qCleanupResources_binary . . . 1622
13.151.2.2 qInitResources_binary . . . 1623
13.151.2.3 qRegisterResourceData . . . 1623
13.151.2.4 qUnregisterResourceData . . . 1623
13.151.3 Variable Documentation . . . 1624
13.151.3.1 qt_resource_data . . . 1624
13.151.3.2 qt_resource_name . . . 1624
13.151.3.3 qt_resource_struct . . . 1624
13.152 src/ckb/release/qrc_image.cpp File Reference . . . 1624
13.152.1 Macro Definition Documentation . . . 1625
13.152.1.1 QT_RCC_MANGLE_NAMESPACE . . . 1625
13.152.1.2 QT_RCC_PREPEND_NAMESPACE . . . 1625
13.152.2 Function Documentation . . . 1625
13.152.2.1 qCleanupResources_image . . . 1625
13.152.2.2 qInitResources_image . . . 1625
13.152.2.3 qRegisterResourceData . . . 1626
13.152.2.4 qUnregisterResourceData . . . 1626
13.152.3 Variable Documentation . . . 1626
13.152.3.1 qt_resource_data . . . 1626
13.152.3.2 qt_resource_name . . . 1626
13.152.3.3 qt_resource_struct . . . 1627
13.153 src/ckb/release/qrc_text.cpp File Reference . . . 1627
13.153.1 Macro Definition Documentation . . . 1627
13.153.1.1 QT_RCC_MANGLE_NAMESPACE . . . 1627
13.153.1.2 QT_RCC_PREPEND_NAMESPACE . . . 1627
13.153.2 Function Documentation . . . 1627

13.153.2.1 qCleanupResources_text . . . 1627
13.153.2.2 qInitResources_text . . . 1628
13.153.2.3 qRegisterResourceData . . . 1628
13.153.2.4 qUnregisterResourceData . . . 1629
13.153.3 Variable Documentation . . . 1629
13.153.3.1 qt_resource_data . . . 1629
13.153.3.2 qt_resource_name . . . 1629
13.153.3.3 qt_resource_struct . . . 1629
13.154 src/ckb/rlistwidget.cpp File Reference . . . 1630
13.155 src/ckb/rlistwidget.h File Reference . . . 1630
13.156 src/ckb/settingswidget.cpp File Reference . . . 1631
13.156.1 Function Documentation . . . 1632
13.156.1.1 right . . . 1632
13.156.2 Variable Documentation . . . 1632
13.156.2.1 devpath . . . 1632
13.156.2.2 modKeys . . . 1633
13.156.2.3 modNames . . . 1633
13.157 src/ckb/settingswidget.h File Reference . . . 1633
13.158 src/ckb/ui_animadddialog.h File Reference . . . 1634
13.159 src/ckb/ui_animsettingdialog.h File Reference . . . 1635
13.160 src/ckb/ui_extrasettingswidget.h File Reference . . . 1635
13.161 src/ckb/ui_fwupgradedialog.h File Reference . . . 1636
13.162 src/ckb/ui_gradientdialog.h File Reference . . . 1637
13.163 src/ckb/ui_kbanimwidget.h File Reference . . . 1638
13.164 src/ckb/ui_kbbindwidget.h File Reference . . . 1639
13.165 src/ckb/ui_kblightwidget.h File Reference . . . 1640
13.166 src/ckb/ui_kbprofiledialog.h File Reference . . . 1641
13.167 src/ckb/ui_kbwidget.h File Reference . . . 1642
13.168 src/ckb/ui_kperfwidget.h File Reference . . . 1643
13.169 src/ckb/ui_layoutdialog.h File Reference . . . 1644
13.170 src/ckb/ui_mainwindow.h File Reference . . . 1645
13.171 src/ckb/ui_modeselectdialog.h File Reference . . . 1646
13.172 src/ckb/ui_mperfwidget.h File Reference . . . 1647
13.173 src/ckb/ui_rebindwidget.h File Reference . . . 1648
13.174 src/ckb/ui_settingswidget.h File Reference . . . 1649

Index 1651

Chapter 1

README

ckb-next: RGB Driver for Linux and macOS

ckb-next is an open-source driver for Corsair keyboards and mice. It aims to bring the features of their proprietary CUE software to the Linux and Mac operating systems. This project is currently a work in progress, but it already supports much of the same functionality, including full RGB animations. More features are coming soon. Testing and bug reports are appreciated!

Disclaimer: ckb-next is not an official Corsair product. It is licensed under the GNU General Public License (version 2) in the hope that it will be useful, but with NO WARRANTY of any kind.

- Device Support
  - Keyboards
  - Mice
- Linux Installation
  - Pre-made packages
  - Preparation
  - Installing
  - Upgrading
  - Uninstalling
- OS X/macOS Installation
  - Binary download
  - Building from source
  - Upgrading (binary)
  - Upgrading (source)
  - Uninstalling
- Usage
  - Major features
  - Roadmap
- Troubleshooting
  - Linux

    - OS X/macOS
    - General
    - Reporting issues
- Known issues
- Contributing
- Contact us
- What happened to the original ckb

See also:

- https://github.com/mattanger/ckb-next/blob/master/DAEMON.md "Manual for the driver daemon"

Device Support

Keyboards

- K63
- K65:
    - RGB
    - non-RGB
    - LUX RGB
    - RGB RAPIDFIRE
- K68
- K70:
    - RGB
    - non-RGB
    - LUX RGB
    - LUX non-RGB
    - RGB RAPIDFIRE
    - non-RGB RAPIDFIRE
- K95:
    - RGB
    - non-RGB∗
    - Platinum∗∗
- Strafe:
    - RGB
    - non-RGB
- = hardware playback not supported. Settings will be saved to software only.

∗∗ = partial support, static hardware playback only.

Mice

- M65:
  - non-RGB
  - PRO RGB
- Sabre:
  - Optical RGB
  - Laser RGB
- Scimitar:
  - RGB
  - PRO RGB
- Harpoon
- Glaive

Linux Installation

Pre-made packages

- Fedora 24/25, CentOS/RHEL 7 (maintained by):
  - `'johanh/ckb'` - based on `master` branch
- Arch Linux (maintained by ,):
  - `'aur/ckb-next'` - based on GitHub releases
  - `'aur/ckb-next-git'` - based on `master` branch
  - `'aur/ckb-next-latest-git'` - based on `newdev` branch
  - `'aur/ckb-next-newfw-git'` - based on `newfw` branch

If you are a package maintainer or want to discuss something with package maintainers let us know in #5, so we can have an accountable and centralized communication about this. *If you would like to maintain a package for your favorite distro/OS, please let us know as well.*

Preparation

ckb-next requires Qt5 (Qt 5.9 is recommended), libudev, zlib, gcc, g++, and glibc.

- Ubuntu: `sudo apt-get install build-essential libudev-dev qt5-default zlib1g-dev libappindicator-dev`
- Fedora: `sudo dnf install zlib-devel qt5-qtbase-devel libgudev-devel libappindicator-de systemd-devel gcc-c++`
- Arch: `sudo pacman -S base-devel qt5-base zlib`
- Other distros: Look for `qt5` or `libqt5*-devel`

Note: If you build your own kernels, ckb-next requires the `CONFIG_INPUT_UINPUT` flag to be enabled. It is located in `Device Drivers -> Input Device Support -> Miscellaneous devices -> User level driver support`. If you don't know what this means, you can ignore this.

Installing

You can download ckb-next using the "Download zip" option on the right or clone it using `git clone`. Extract it and open the ckb-master directory in a terminal. Run `./quickinstall`. It will attempt to build ckb and then ask if you'd like to install/run the application. If the build doesn't succeed, or if you'd like to hand-tune the compilation of ckb, see https://github.com/mattanger/ckb-next/blob/master/BUILD.md "'BUILD.md'" for instructions.

Upgrading

To install a new version of ckb, or to reinstall the same version, first delete the ckb-master directory and the zip file from your previous download. Then download the source code again and re-run `./quickinstall`. The script will automatically replace the previous installation. You may need to reboot afterward.

Uninstalling

First, stop the ckb-daemon service and remove the service file.

- If you have systemd (Ubuntu versions starting with 15.04): "' sudo systemctl stop ckb-daemon sudo rm -f /usr/lib/systemd/system/ckb-daemon.service "'
- If you have Upstart (Ubuntu versions earlier than 15.04): "' sudo service ckb-daemon stop sudo rm -f /etc/init/ckb-daemon.conf "'
- If you have OpenRC: "' sudo rc-service ckb-daemon stop sudo rc-update del ckb-daemon default sudo rm -f /etc/init.d/ckb-daemon "
- `If you're not sure, re-run the`quickinstall`script and proceed to the service installation. The script will say`System service: Upstart detected`or`System service: systemd detected'. Please be aware that OpenRC is currently not detected automatically.

Afterward, remove the applications and related files: "' sudo rm -f /usr/bin/ckb /usr/bin/ckb-daemon /usr/share/applications/ckb.desktop /usr/share/icons/hicolor/512x512/apps/ckb.png sudo rm -rf /usr/lib/ckb-animations "'

Before https://github.com/mattanger/ckb-next/commit/f347e60df211c60452f95084b6c46dc4ec5f42 animations were located elsewhere, try removing them as well: "' sudo rm -rf /usr/bin/ckb-animations "'

OS X/macOS Installation

Binary download

macOS `pkg` can be downloaded from `GitHub Releases`. It is always built with the last available stable Qt version and tagrets 10.10 SDK. If you run 10.9.x, you'll need to build the project from source and comment out `src/ckb-heat` (and the backslash above it) inside `ckb.pro`.

Building from source

Install the latest version of Xcode from the App Store. While it's downloading, open the Terminal and execute `xcode-select --install` to install Command Line Tools. Then open Xcode, accept the license agreement and wait for it to install any additional components (if necessary). When you see the "Welcome to Xcode" screen, from the top bar choose `Xcode -> Preferences -> Locations -> Command Line Tools` and select an SDK version. Afterwards install `Homebrew` and execute `brew install qt5` in the Terminal.

> **Note**: If you decide to use the official Qt5 package from Qt website instead, you will have to edit the installation script and provide installation paths manually due to a qmake bug.

The easiest way to build the driver is with the `quickinstall` script, which is present in the ckb-master folder. Double-click on `quickinstall` and it will compile the app for you, then ask if you'd like to install it system-wide. If the build fails for any reason, or if you'd like to compile and install manually, see https://github.com/ccMS-C/ckb/blob/master/BUILD.md "'BUILD.md'".

Upgrading (binary)

Download the latest `ckb.pkg`, run the installer, and reboot. The newly-installed driver will replace the old one.

Upgrading (source)

Remove the existing ckb-master directory and zip file. Re-download the source code and run the `quickinstall` script again. The script will automatically replace the previous installation. You may need to reboot afterward.

Uninstalling

Drag `ckb.app` into the trash. Then stop and remove the agent:

"'sh sudo unload /Library/LaunchDaemons/com.ckb.daemon.plist sudo rm /Library/LaunchDaemons/com.ckb.-daemon.plist "'

Usage

The user interface is still a work in progress.

Major features

- Control multiple devices independently
- United States and European keyboard layouts
- Customizable key bindings
- Per-key lighting and animation
- Reactive lighting
- Multiple profiles/modes with hardware save function
- Adjustable mouse DPI with ability to change DPI on button press

Closing ckb will actually minimize it to the system tray. Use the Quit option from the tray icon or the settings screen to exit the application.

Roadmap

- **v0.3 release:**
- Ability to store profiles separately from devices, import/export them
- More functions for the Win Lock key
- Key macros
- **v0.4 release:**
- Ability to import CUE profiles

- Ability to tie profiles to which application has focus
- **v0.5 release:**
- Key combos
- Timers?
- **v1.0 release:**
- OSD? (Not sure if this can actually be done)
- Extra settings?
- ????

Troubleshooting

Linux

If you have problems connecting the device to your system (device doesn't respond, ckb-daemon doesn't recognize or can't connect it) and/or you experience long boot times when using the keyboard, try adding the following to your kernel's `cmdline`:

- K65 RGB: `usbhid.quirks=0x1B1C:0x1B17:0x20000408`
- K65 LUX: `usbhid.quirks=0x1B1C:0x1B37:0x20000408`
- K68: `usbhid.quirks=0x1B1C:0x1B3F:0x20000408`
- K70: `usbhid.quirks=0x1B1C:0x1B09:0x20000408`
- K70 LUX: `usbhid.quirks=0x1B1C:0x1B36:0x20000408`
- K70 RGB: `usbhid.quirks=0x1B1C:0x1B13:0x20000408`
- K95: `usbhid.quirks=0x1B1C:0x1B08:0x20000408`
- K95 RGB: `usbhid.quirks=0x1B1C:0x1B11:0x20000408`
- Strafe: `usbhid.quirks=0x1B1C:0x1B15:0x20000408`
- Strafe RGB: `usbhid.quirks=0x1B1C:0x1B20:0x20000408`
- M65 RGB: `usbhid.quirks=0x1B1C:0x1B12:0x20000408`
- Sabre RGB Optical: `usbhid.quirks=0x1B1C:0x1B14:0x20000408`
- Sabre RGB Laser: `usbhid.quirks=0x1B1C:0x1B19:0x20000408`
- Scimitar RGB: `usbhid.quirks=0x1B1C:0x1B1E:0x20000408`

For instructions on adding `cmdline` parameters in Ubuntu, see `https://wiki.ubuntu.com/Kernel/KernelBootParameters`

If you have multiple devices, combine them with commas, starting after the `=`. For instance, for K70 RGB + M65 RGB: `usbhid.quirks=0x1B1C:0x1B13:0x20000408,0x1B1C:0x1B12:0x20000408`

If it still doesn't work, try replacing `0x20000408` with `0x4`. Note that this will cause the kernel driver to ignore the device(s) completely, so you need to ensure ckb-daemon is running at boot or else you'll have no input. This will not work if you are using full-disk encryption.

If you see **GLib** critical errors like "' GLib-GObject-CRITICAL **: g_type_add_interface_static: assertion 'G_TYPE_IS_INSTANTIATABLE (instance_type)' failed "' read `this Arch Linux thread` and try different combinations from it. If it doesn't help, you might want get support from your distribution community and tell them you cannot solve the problem in this thread.

If you're using **Unity** and the tray icon doesn't appear correctly, run `sudo apt-get install libappindicator-dev`. Then reinstall ckb.

Fedora 26 Color Changer Freeze Fix

If you're running Fedora 26, a working solution for the color changer freezing issue is to install qt5ct `dnf install qt5ct` then modify your /etc/environment file to contain the line `QT_QPA_PLATFORMTHEME=qt5ct`

OS X/macOS

- ∗∗"ckb.pkg" can't be opened because it is from an unidentified developer∗∗ Right-click (control-click) on ckb.pkg and select Open. This new dialog box will give you the option to open anyway, without changing your system preferences.
- **Modifier keys (Shift, Ctrl, etc.) are not rebound correctly** ckb does not recognize modifier keys rebound from System Preferences. You can rebind them again within the application.
- ∗∗∼ key prints `§±`∗∗ Check your keyboard layout on ckb's Settings screen. Choose the layout that matches your physical keyboard.
- **Compile problems** Can usually be resolved by rebooting your computer and/or reinstalling Qt. Make sure that Xcode works on its own. If a compile fails, delete the `ckb-master` directory as well as any automatically generated `build-ckb` folders and try again from a new download.
- **Scroll wheel does not scroll** As of #c3474d2 it's now possible to **disable scroll acceleration** from the GUI. You can access it under "OSX tweaks" in the "More settings" screen. Once disabled, the scroll wheel should behave consistently.

General

Please ensure your keyboard firmware is up to date. If you've just bought the keyboard, connect it to a Windows computer first and update the firmware from Corsair's official utility.

Before reporting an issue, connect your keyboard to a Windows computer and see if the problem still occurs. If it does, contact Corsair. Additionally, please check the Corsair user forums to see if your issue has been reported by other users. If so, try their solutions first.

Common issues:

- **Problem:** ckb says "No devices connected" or "Driver inactive"
- **Solution:** Try rebooting the computer and/or reinstalling ckb. Try removing the keyboard and plugging it back in. If the error doesn't go away, try the following:
- **Problem:** Keyboard doesn't work in BIOS, doesn't work at boot
- **Solution:** Some BIOSes have trouble communicating with the keyboard. They may prevent the keyboard from working correctly in the operating system as well. First, try booting the OS *without* the keyboard attached, and plug the keyboard in after logging in. If the keyboard works after the computer is running but does not work at boot, you may need to use the keyboard's BIOS mode option.
- BIOS mode can be activated using the poll rate switch at the back of the keyboard. Slide it all the way to the position marked "BIOS". You should see the scroll lock light blinking to indicate that it is on. (Note: Unfortunately, this has its own problems - see Known Issues. You may need to activate BIOS mode when booting the computer and deactivate it after logging in).
- **Problem:** Keyboard isn't detected when plugged in, even if driver is already running
- **Solution:** Try moving to a different USB port. Be sure to follow `Corsair's USB connection requirements`. Note that the keyboard does not work with some USB3 controllers - if you have problems with USB3 ports, try USB2 instead. If you have any USB hubs on hand, try those as well. You may also have success sliding the poll switch back and forth a few times.

Reporting issues

If you have a problem that you can't solve (and it isn't mentioned in the Known Issues section below), you can report it on the GitHub issue tracker. Before opening a new issue, please check to see if someone else has reported your problem already - if so, feel free to leave a comment there.

Known issues

- Using the keyboard in BIOS mode prevents the media keys (including mute and volume wheel), as well as the K95's G-keys from working. This is a hardware limitation.
- The tray icon doesn't appear in some desktop environments. This is a known Qt bug. If you can't see the icon, reopen ckb to bring the window back.
- When starting the driver manually, the Terminal window sometimes gets spammed with enter keys. You can stop it by unplugging and replugging the keyboard or by moving the poll rate switch.
- When stopping the driver manually, the keyboard sometimes stops working completely. You can reconnect it by moving the poll rate switch.
- On newer versions of macOS (i.e. 10.12 and up) CMD/Shift+select does not work, yet. Stopping the daemon and GUI for `ckb` will fix this issue temporarily.

Contributing

You can contribute to the project by opening a pull request. It's best if you base your changes off of the `testing` branch as opposed to the `master`, because the pull request will be merged there first. If you'd like to contribute but don't know what you can do, take a look at the issue tracker and see if any features/problems are still unresolved. Feel free to ask if you'd like some ideas.

Contact us

There are multiple ways you can get in touch with us:

- join `ckb-next` mailing list
- open a GitHub Issue
- hop on `#ckb-next` on Freenode to chat

What happened to the original ckb

Due to time restrictions, the original author of **ckb** ccMSC hasn't been able to further develop the software. So the community around it decided to take the project over and continue its development. That's how **ckb-next** was created. Currently it's not rock solid and not very easy to set up on newer systems but we are actively working on this. Nevertheless the project already incorporates a notable amount of fixes and patches in comparison to the original ckb.

Chapter 2

Building ckb

Linux

You can build the project by running `./qmake-auto && make` in a Terminal inside the ckb-master directory. The binaries will be placed in a new `bin` directory assuming they compile successfully. If you get a `No suitable qmake found` error, make sure Qt5 is installed and up to date. You may have to invoke qmake manually, then run `make` on its own. If you have Qt Creator installed, you can open `ckb.pro` (when asked to configure the project, make sure "Desktop" is checked) and use `Build > Build Project "ckb" (Ctrl+B)` to build the application instead.

Running as a service:

First copy the binary and the service files to their system directories:

- Upstart (Ubuntu, prior to 15.04): `sudo cp -R bin/* /usr/bin && sudo cp service/upstart/ckb-daemon.conf /etc/init`
- Systemd (Ubuntu 15.04 and later): `sudo cp -R bin/* /usr/bin && sudo cp service/systemd/ckb-daemon.service /usr/lib/systemd/system`
- OpenRC: `sudo cp -R bin/* /usr/bin && sudo cp service/openrc/ckb-daemon /etc/init.d/`

To launch the driver and enable it at start-up:

- Upstart: `sudo service ckb-daemon start`
- Systemd: `sudo systemctl start ckb-daemon && sudo systemctl enable ckb-daemon`
- OpenRC: `sudo rc-service ckb-daemon start && sudo rc-update add ckb-daemon default`

Open the `bin` directory and double-click on `ckb` to launch the user interface. If you want to run it at login, add `ckb --background` to your Startup Applications.

Running manually:

Open the `bin` directory in a Terminal and run `sudo ./ckb-daemon` to start the driver. To start the user interface, run `./ckb`. Running the driver manually may be useful for testing/debugging purposes, but you must leave the terminal window open and you'll have to re-run it at every reboot, so installing it as a service is the best long-term solution.

OSX

Open ckb.pro in Qt Creator. You should be prompted to configure the project (make sure the "Desktop" configuration is selected and not iOS). Once it's finished loading, press `Cmd+B` or select `Build > Build Project "ckb"` from the menu bar. When it's done, you should see a newly-created `ckb.app` in the project directory. Exit Qt Creator.

Alternatively, open a Terminal in the ckb-master directory and run `./qmake-auto && make`. It will detect Qt automatically if you installed it to one of the standard locations. You should see a newly created `ckb.app` if the build is successful.

Running as a service:

Copy `ckb.app` to your Applications folder. Copy the file `'service/launchd/com.ckb.daemon.plist'` to your computer's `/Library/LaunchDaemons` folder (you can get to it by pressing `Cmd+Shift+G` in Finder and typing the location). Then open a Terminal and run the following commands to launch the driver:

"' sudo chown root:wheel /Library/LaunchDaemons/com.ckb.daemon.plist sudo chmod 0700 /Library/LaunchDaemons/com.ckb.daemon.plist sudo launchctl load /Library/LaunchDaemons/com.ckb.daemon.plist "'

After you're done, open `ckb.app` to launch the user interface.

Running manually:

Open a Terminal in the ckb directory and run `sudo ckb.app/Contents/Resources/ckb-daemon` to start the driver. Open `ckb.app` to start the user interface. Note that you must leave the terminal window open and must re-launch the driver at every boot if you choose this; installing as a service is the better long term solution.

Chapter 3

cbk Improvements Roadmap

Short term plan

- merge existing PR submitted to original ckb repo
- Contact other developers interested in collaboration on a new and improved version of ckb
- Figure out the issues relating to MacOS Sierra and other version
- Device support:
  - Determine which will need support other than just USB id additions
- Address existing bugs. Not help requests.

Chapter 4

DAEMON

The daemon provides devices at `/dev/input/ckb*`, where `*` is the device number, starting at 1. Up to 9 devices may be connected at once and controlled independently. The daemon additionally provides `/dev/input/ckb0`, which stores driver information.

Mac note: The devices on OSX are located at `/var/run/ckb*` and not `/dev/input/ckb*`. So wherever you see `/dev/input` in this document, replace it with `/var/run`.

`/dev/input/ckb0` contains the following files:

- `connected`: A list of all connected devices, one per line. Each line contains a device path followed by the device's serial number and its description.
- `pid`: The process identifier of the daemon.
- `version`: The daemon version.

Other `ckb*` devices contain the following:

- `cmd`: Keyboard controller.
- `notify0`: Keyboard- or mouse notifications.
- `notify1`: Keyboard- or mouse notifications, used for macro recording.
- `features`: Device features.
- `fwversion`: Device firmware version (not present on all devices).
- `model`: Device description/model.
- `pollrate`: Poll rate in milliseconds (not present on all devices).
- `productid`: Contains the USB productID of the hardware
- `serial`: Device serial number. `model` and `serial` will match the info found in `ckb0/connected`

Commands

The `/dev/input/ckb*/cmd` nodes accept input in the form of text commands. They are normally accessible to all users on the system (see Security section). Commands should be given in the following format: `[mode <n>] command1 [paramter1] [command2] [parameter2] [command3] [parameter3] ...`

In a terminal shell, you can do this like `echo mycommand > /dev/input/ckb1/cmd`. Programmatically, you can open and write them as regular files. When programming, you must append a newline character and flush the output before your command(s) will actually be read.

The mode parameter is used to group settings. Most (but not all) settings are mode-specific; that is, changing mode 1 will not affect mode 2. By default, all commands affect the current mode. Use mode <n> switch to change the current mode.

When plugged in, all devices start in hardware-controlled mode (also known as idle mode) and will not respond to commands. Before issuing any other commands, write active to the command node, like echo active > /dev/input/ckb1/cmd. To put the device back into hardware mode, issue the idle command.

Features

The features node describes features supported by the device, which may not be present on all devices. The first two words in the features node are always <vendor> <model>, like corsair k70. After that, any of the following features may appear:

- adjrate: Device supports adjustable poll rate.
- bind: Device supports key rebinding.
- fwupdate: Device supports firmware updates.
- fwversion: Device has a detectable firmware version (stored in the fwversion node).
- notify: Device supports key notifications.
- pollrate: Device has a detectable poll rate (stored in the pollrate node).
- rgb: Device supports RGB lighting.

Keyboard layout

The driver has no concept of keyboard layouts; all keys are referred to by their English names regardless of the underlying hardware. This means that, for instance, in an AZERTY layout the q key in ckb-daemon corresponds to A on the physical keyboard. Note that on UK/european (ISO) layouts, the backslash key (beside left shift) is called bslash_iso, while bslash refers to the backslash on the US keyboard. The key next to Enter on the ISO keyboard is known as hash. See src/ckb-daemon/keymap.c for the full table of supported keys.

For technical reasons, the OSX driver may swap the bslash_iso and grave keys if the keyboard layout is not set correctly. To compensate for this, write layout iso or layout ansi to the command node.

Poll rate

A device's current poll rate can be read from its pollrate node, assuming it has one. Keyboards have a hardware switch to control poll rate and cannot be adjusted via software. However, mice have a software-controlled poll rate. You can change it by issuing pollrate <interval> to the command node, where interval is the time in milliseconds. Valid poll rates are 1, 2, 4, and 8.

Profiles and modes

Each mode has its own independent binding and lighting setup. When the daemon starts or a keyboard is plugged in, the profile will be loaded from the hardware. By default, all commands will update the currently selected mode. The mode <n> command may be used to change the settings for a different mode. Up to 6 modes are available. Each keyboard has one profile, which may be given a name. Mode 1 may be saved to the device hardware, or modes 1-3 in the case of the K95. Modes 4 through 6 are software-only. Profile management commands are as follows:

- profilename <name> sets the profile's name. The name must be written without spaces; to add a space, use %20.

- `name <name>` sets the current mode's name. Use `mode <n> name <name>` to set a different mode's name.
- `profileid <guid> [<modification>]` sets a profile's ID. The GUID must be written in registry format, like `{12345678-ABCD-EF01-2345-6789ABCDEF01}`. The optional modification number must be written with 8 hex digits, like `ABCDEF01`.
- `id <guid> [<modification>]` sets a mode's ID.
- `mode <n> switch` switches the keyboard to mode N. If the mode does not exist, it will be created with a blank ID, black lighting, and default bindings.
- `hwload` loads the RGB profile from the hardware. Key bindings and non-hardware RGB modes are unaffected.
- `hwsave` saves the RGB profile to the hardware.
- `erase` erases the current mode, resetting its lighting and bindings. Use `mode <n> erase` to erase a different mode.
- `eraseprofile` erases the entire profile, deleting its name, ID, and all of its modes.

Examples:

- `profilename My%20Profile mode 1 name Mode%201 mode 2 name Mode%202 mode 3 name Mode%203` will name the profile "My Profile" and name modes 1-3 "Mode 1", "Mode 2", and "Mode 3".
- `eraseprofile hwload` resets the entire profile to its hardware settings.

LED commands

The backlighting is controlled by the `rgb` commands.

- `rgb <RRGGBB>` sets the entire keyboard to the color specified by the hex constant RRGGBB.
- `rgb <key>:<RRGGBB>` sets the specified key to the specified hex color.

Examples:

- `rgb ffffff` makes the whole keyboard white.
- `rgb 000000` makes the whole keyboard black.
- `rgb esc:ff0000` sets the Esc key red but leaves the rest of the keyboard unchanged.

Multiple keys may be changed to one color when separated with commas, for instance:

- `rgb w,a,s,d:0000ff` sets the WASD keys to blue.

Additionally, multiple commands may be combined into one, for instance:

- `rgb ffffff esc:ff0000 w,a,s,d:0000ff` sets the Esc key red, the WASD keys blue, and the rest of the keyboard white (note the lack of a key name before `ffffff`, implying the whole keyboard is to be set).

By default, the controller runs at 30 FPS, meaning that attempts to animate the LEDs faster than that will be ignored. If you wish to change it, send the command `fps <n>`. The maximum frame rate is 60.

For devices running in 512-color mode, color dithering can be enabled by sending the command `dither 1`. The command `dither 0` disables dithering.

Indicators

The indicator LEDs (Num Lock, Caps Lock, Scroll Lock) are controlled with the `i` commands.

- `ioff <led>` turns an indicator off permanently. Valid LED names are `num`, `caps`, and `scroll`.
- `ion <led>` turns an indicator on permanently.
- `iauto <led>` turns an indicator off or on automatically (default behavior).

Binding keys

Keys may be rebound through use of the `bind` commands. Binding is a 1-to-1 operation that translates one keypress to a different keypress regardless of circumstance.

- `bind <key1>:<key2>` remaps key1 to key2.
- `unbind <key>` unbinds a key, causing it to lose all function.
- `rebind <key>` resets a key, returning it to its default binding.

Examples:

- `bind g1:esc` makes G1 become an alternate Esc key (the actual Esc key is not changed).
- `bind caps:tab tab:caps` switches the functions of the Tab and Caps Lock keys.
- `unbind lwin rwin` disables both Windows keys, even without using the keyboard's Windows Lock function.
- `rebind all` resets the whole keyboard to its default bindings.

Key macros

Macros are a more advanced form of key binding, controlled with the `macro` command.

- `macro <keys>:<command>` binds a key combination to a command, where the command is a series of key presses. To combine keys, separate them with `+`; for instance, `lctrl+a` binds a macro to (left) Ctrl+A. In the command field, enter `+<key>` to trigger a key down or `-<key>` to trigger a key up. To simulate a key press, use `+<key>,-<key>`.
- `macro <keys>:clear` clears commands associated with a key combination. Only one macro may be assigned per combination; assigning a second one will overwrite the first.
- `macro clear` clears all macros.

Examples:

- `macro g1:+lctrl,+a,-a,-lctrl` triggers a Ctrl+A when G1 is pressed.
- `macro g2+g3:+lalt,+f4,-f4,-lalt` triggers an Alt+F4 when G2 and G3 are pressed simultaneously.

Assigning a macro to a key will cause its binding to be ignored; for instance, `macro a:+b,-b` will cause A to generate a B character regardless of its binding. However, `macro lctrl+a:+b,-b` will cause A to generate a B only when Ctrl is also held down.

Macro playback delay

There are two types of playback delay that can be set with macros; global and local. Setting a *global delay* value introduces a time delay between events during macro execution or playback. *Local delay* allows setting the delay after an individual event, overriding the global delay value for that event. Thus global delay can be used to set the overall playback speed of macros and local delays can be used to tune individual events within a macro.

All delay values are specified in microseconds (us) and are positive values from `0` to `UINT_MAX - 1`. This means delays range from 0 to just over 1 hour (4,294,967,294us, 4,294 seconds, 71 minutes, or 1.19 hours). A value of zero (0) represents no delay between actions.

Global macro delay (default delay)

Global delay allows macro playback speed to be changed. It sets the time between (actually after) each recorded macro event. If global delay is set to 1 microsecond then a 1 ms delay will follow each individual macro event when the macro is triggered.

The *global delay* is set with the ckb-daemon's existing (in testing branch) `delay` command followed by an unsigned integer representing the number of microseconds to wait after each macro action and before the next.

Global delay can also be set to `on` which maintains backwards compatibility with the current development of `ckb-daemon` for long macro playback. That is, setting the global delay to `on` introduces a 30us and a 100us delay based on the macro's length during playback.

NOTE: This setting also introduces a delay after the last macro action. This functionality exists in the current testing branch and was left as-is. It is still to be determined if this is a bug or a feature.

Examples:

- `delay 1000` sets a 1,000us delay between action playback.
- `delay on` sets long macro delay; 30us for actions between 20 and 200, 100us for actions > 200.
- `delay off` sets no delay (same as 0).
- `delay 0` sets no delay (same as off).
- `delay spearmint-potato` is invalid input, sets no delay (same as off).

Local macro delay (keystroke delay)

Local Delay allows each macro action to have a post-action delay associated with it. This allows a macro to vary it's playback speed for each event. If no local delay is specified for a macro action, then the global `delay` (above) is used. All delay values are in microsecods (us) as with the global delay setting.

Examples:

- `macro g5:+d,-d,+e=5000,-e,+l,-l=10000,+a,-a,+y,-y=1000000,+enter,-enter` define a macro for `g5` with a 5,000us delay between the `e` down and `e` up actions. A 1,000us delay between `l` up and `a` down, a delay of one second (1,000,000us) after `y` up and before `enter`, and the global delay for all other actions.
- `macro g5:+d,-d=0` use default delay between `d` down and `d` up and no delay (0us) after `d` up. This removes the noted feature/bug (above) where the last action has a trailing delay associated with it.

DPI and mouse settings

DPI settings are stored in a bank. They are controlled with the `dpi` command.

- `dpi <stage>:<x>,<y>` sets the DPI for a given `stage` to `x` by `y`. Valid stages are `0` through `5`. In hardware, `1` is the first (lowest) stage and `5` is the highest. Stage `0` is used for Sniper mode.
- `dpi <stage>:<xy>` sets both X and Y.

- `dpi <stage>:off` disables a DPI stage.
- `dpisel <stage>` sets the current stage selection.

In order to change the mouse's current DPI, first update one of the stages with the value you want, then select that stage. For instance:

- `dpi 1:1000 dpisel 1` sets the current DPI to 1000x1000.

Additional mouse settings:

- `lift <height>` sets the lift height, from `1` (lowest) to `5` (highest)
- `snap <on|off>` enables or disables Angle Snap.

Notifications

The keyboard can be configured to generate user-readable notifications on keypress events. These are controlled with the `notify` commands. In order to see events, read from `/dev/input/ckb*/notify0`. In a terminal, you can do this like `cat /dev/input/ckb1/notify0`. Programmatically, you can open it for reading like a regular file.

Note that the file can only reliably be read by one application: if you try to open it in two different programs, they may both fail to get data. Data will be buffered as long as no programs are reading, so you will receive all unread notifications as soon as you open the file. If you'd like to read notifications from two separate applications, send the command `notifyon <n>` to the keyboard you wish to receive notifications from, where N is a number between 1 and 9. If `/dev/input/ckb*/notify<n>` does not already exist, it will be created, and you can read notifications from there without disrupting any other program. To close a notification node, send `notifyoff <n>`.

`notify0` is always open and will not be affected by `notifyon`/`notifyoff` commands. By default, all notifications are printed to `notify0`. To print output to a different node, prefix your command with `@<node>`.

Notifications are printed with one notification per line. Commands are as follows:

- `notify <key>:on` or simply `notify <key>` enables notifications for a key. Each key will generate two notifications: `key +<key>` when the key is pressed, and `key -<key>` when it is released.
- `notify <key>:off` turns notifications off for a key.

Examples:

- `notify w a s d` sends notifications whenever W, A, S, or D is pressed.
- `notify g1 g2 g3 g4 g5 g6 g7 g8 g9 g10 g11 g12 g13 g14 g15 g16 g17 g18 mr m1 m2 m3 light lock` prints a notification whenever a non-standard key is pressed.
- `notify all:off` turns all key notifications off.
- `@5 notify esc` prints Esc key notifications to `notify5`.

Note: Key notifications are *not* affected by bindings. For instance, if you run `echo bind a:b notify a > /dev/input/ckb1/cmd` and then press the A key, the notifications will read `key +a key -a`, despite the fact that the character printed on screen will be `b`. Likewise, unbinding a key or assigning a macro to a key does not affect the notifications.

Indicator notifications

You can also choose to receive notifications for the indicator LEDs by using the `inotify` command. For instance, `inotify caps:on` or simply `inotify caps` will print notifications whenever the Caps Lock LED is toggled. The notifications will read `i +caps` when the light is turned on and `i -caps` when it is turned off. It is also possible to toggle all indicators at once using `inotify all` or `inotify all:off`.

Like key notifications, indicator notifications are not affected by bindings, nor by the `ion`, `ioff`, or `iauto` commands. The notifications will reflect the state of the LEDs as seen be the event device.

Getting parameters

Parameters can be retrieved using the `get` command. The data will be sent out as a notification. Generally, the syntax to get the data associated with a command is `get :<command>` (note the colon), and the associated data will be returned in the form of `<command> <data>`. The following data may be gotten:

- `get :mode` returns the current mode in the form of a `switch` command. (Note: Do not use this in a line containing a `mode` command or it will return the mode that you selected, rather than the keyboard's current mode.)
- `get :name` returns the current mode's name in the form of `mode <n> name <name>`. To see the name of another mode, use `mode <n> get :name`. The name is URL-encoded; spaces are written as %20. The name may be truncated, so `name <some long string> get :name` may return something shorter than what was entered.
- `get :profilename` returns the profile's name, in the form of `profilename <name>`. As above, it is URL-encoded and may be truncated.
- `get :hwname` and `get :hwprofilename` return the same thing except taken from the current hardware profile instead of the in-memory profile. The output is identical but will read `hwname` instead of `name` and `hwprofilename` instead of `profilename`.
- `get :id` returns the current mode's ID and modification number in the form of `mode <n> id <guid> <modification>`.
- `get :profileid` returns the current profile's ID and modification number in the form of `profileid <guid> <modification>`.
- `get :hwid` and `get :hwprofileid` return the same thing except from the current hardware profile/mode. As before, the ouput will be the same but with `hwid` and `hwprofileid` instead of `id` and `profileid`.
- `get :rgb` returns an `rgb` command equivalent to the current RGB state.
- `get :hwrgb` does the same thing, but retrieves the colors currently stored in the hardware profile. The output will say `hwrgb` instead of `rgb`.
- `get :dpi` returns a `dpi` command equivalent to the current DPI bank.
- `get :dpisel` returns a `dpisel` command for the currently-selected DPI stage.
- `get :lift` returns a `lift` command for the current lift height.
- `get :snap` returns the current angle snap status.
- `get :hwdpi`, `get :hwdpisel`, `get :hwlift`, and `get :hwsnap` return the same properties, but for the current hardware profile.
- `get :keys` and `get :i` return the current keypress status and indicator status, respectively. They will indicate all currently pressed keys and all currently active indicators, like `key +enter` and `i +num`.

Like `notify`, you must prefix your command with `@<node>` to get data printed to a node other than `notify0`.

Firmware updates

WARNING: Improper use of `fwupdate` may brick your device; use this command *at your own risk*. I accept no responsibility for broken keyboards.

The latest firmware versions and their URLs can be found in the `FIRMWARE` document. To update your keyboard's firmware, first extract the contents of the zip file and then issue the command `fwupdate /path/to/fw/file.bin` to the keyboard you wish to update. The path name must be absolute and must not include spaces. If it succeeded, you should see `fwupdate <path> ok` logged to the keyboard's notification node and then the device will disconnect and reconnect. If you see `fwupdate <path> invalid` it means that the firmware file was not valid for the device; more info may be available in the daemon's `stdout`. If you see `fwupdate <path> fail` it means that the file was valid but the update failed at a hardware level. The keyboard may disconnect/reconnect anyway or it may remain in operation.

When the device reconnects you should see the new firmware version in its `fwversion` node; if you see `0000` instead it means that the keyboard did not update successfully and will need another `fwupdate` command in order to function again. If the update fails repeatedly, try connecting the keyboard to a Windows PC and using the official firmware update in CUE.

Restart

Because sometimes the communication between the daemon and the keyboard is corrupted after resuming from standby or suspend, a restart function is implemented. It first calls the quit() function, then it calls main() again with the original parameter list.

There are two ways to restart the daemon:

- send the string "restart some-description-as-one-word" to the cmd-pipe (normally /dev/input/ckb1/cmd or /dev/input/ckb2/cmd, depending on what device gets which ID.
- send SIGUSR1 to the daemon process (as root).

Later on, there may be a user interface in the client for the first method.

Security

By default, all of the `ckb*` nodes may be accessed by any user. For most single-user systems this should not present any security issues, since only one person will have access to the computer anyway. However, if you'd like to restrict the users that can write to the `cmd` nodes or read from the `notify` nodes, you can specify the `--gid=<group>` option at start up. For instance, on most systems you could run `ckb-daemon --gid=1000` to make them accessible only by the system's primary user. `ckb-daemon` must still be run as root, regardless of which `gid` you specify. The `gid` option may be set only at startup and cannot be changed while the daemon is running.

The daemon additionally supports a `--nonotify` option to disable key notifications, to prevent unauthorized programs from logging key input. Note that this will interfere with some of `ckb`'s abilities. It is also highly unlikely to increase security unless you are using the program in a stripped down terminal environment without Xorg. For most use cases there are many other (more likely) ways that a keylogger program could compromise your system. Nevertheless, the option is provided for the sake of paranoia. If you'd like to disable key rebinding as well, launch the daemon with `--nobind`. `--nobind` implies `--nonotify`, so notifications will also be disabled. As with `--gid`, these options must be set at startup and cannot be changed while the daemon is running.

Chapter 5

Todo List

Global Kb::Kb (QObject *parent, const QString &path)

Is adding notify2 to the notifypaths neccessary?

Global Kb::~Kb ()

I don't think, that notifypaths is used somewhere. So why do we have it? If we do not need it, searching for an ununsed notify channel can easy be refactored to a private member function.

Global RebindWidget::applyChanges (const QStringList &keys, bool doUnbind)

There is still a bug in the state machine: If you record a macro in asTyped-mode, switch to another mode and change the vontent of the pteMacroBox manually, then the changes are not saved in the timing buffer. But anyhow, let's do more relevant things...

Global RebindWidget::on_btnClearMacro_clicked ()

I do not know what is the better solution with the delay-buttons in case of clicking clear: Reset the button to the default value or do not touch it? Not clear is ignored.

Chapter 6

Deprecated List

Class QuaZipFileInfo

Use QuaZipFileInfo64 instead. Not only it supports large files, but also more convenience methods as well.

Chapter 7

Namespace Index

7.1 Namespace List

Here is a list of all namespaces with brief descriptions:

Ui 39

Chapter 8

Hierarchical Index

8.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

AutoRun . . . 118
ckb_gradient . . . 1221
ckb_key . . . 1221
ckb_runctx . . . 1221
CkbSettings . . . 122
ColorMap . . . 149
curfile64_info . . . 1456
MPerfWidget::DpiUi . . . 776
file_in_zip64_read_info_s . . . 1385
KbFirmware::FW . . . 179
KPerfWidget::IndicatorUi . . . 700
JlCompress . . . 238
KbId . . . 1250
Key . . . 597
KeyMap . . . 631
KeyPatch . . . 1284
linkedlist_data_s . . . 1456
linkedlist_datablock_internal_s . . . 1456
AnimScript::Param . . . 52
GradientDialog::Preset . . . 793
QDialog . . . 795
- AnimAddDialog . . . 41
- AnimSettingDialog . . . 97
- ExtraSettingsWidget . . . 170
- FwUpgradeDialog . . . 180
- GradientDialog . . . 209
- KbProfileDialog . . . 551
- LayoutDialog . . . 717
- ModeSelectDialog . . . 751

QIODevice . . . 796
- QuaGzipFile . . . 812
- QuaZIODevice . . . 819
- QuaZipFile . . . 877

QIODevice_descriptor . . . 797
QListWidget . . . 798
- RListWidget . . . 1001

QMainWindow . . . 799

MainWindow . . . 732
QObject . . . 800
AnimScript . . . 52
CkbSettingsWriter . . . 137
KbAnim . . . 306
KbBind . . . 361
KbFirmware . . . 409
KbLight . . . 421
KbManager . . . 472
KbMode . . . 484
KbPerf . . . 498
KbProfile . . . 533
KeyAction . . . 600
QPushButton . . . 801
ColorButton . . . 139
GradientButton . . . 195
qt_meta_stringdata_AnimAddDialog_t . . . 1524
qt_meta_stringdata_AnimScript_t . . . 1527
qt_meta_stringdata_AnimSettingDialog_t . . . 1530
qt_meta_stringdata_CkbSettingsWriter_t . . . 1533
qt_meta_stringdata_ColorButton_t . . . 1535
qt_meta_stringdata_ExtraSettingsWidget_t . . . 1537
qt_meta_stringdata_FwUpgradeDialog_t . . . 1540
qt_meta_stringdata_GradientButton_t . . . 1543
qt_meta_stringdata_GradientDialog_t . . . 1546
qt_meta_stringdata_GradientDialogWidget_t . . . 1549
qt_meta_stringdata_Kb_t . . . 1551
qt_meta_stringdata_KbAnim_t . . . 1554
qt_meta_stringdata_KbAnimWidget_t . . . 1556
qt_meta_stringdata_KbBind_t . . . 1559
qt_meta_stringdata_KbBindWidget_t . . . 1561
qt_meta_stringdata_KbFirmware_t . . . 1563
qt_meta_stringdata_KbLight_t . . . 1566
qt_meta_stringdata_KbLightWidget_t . . . 1569
qt_meta_stringdata_KbManager_t . . . 1572
qt_meta_stringdata_KbMode_t . . . 1574
qt_meta_stringdata_KbPerf_t . . . 1576
qt_meta_stringdata_KbProfile_t . . . 1579
qt_meta_stringdata_KbProfileDialog_t . . . 1581
qt_meta_stringdata_KbWidget_t . . . 1585
qt_meta_stringdata_KeyAction_t . . . 1587
qt_meta_stringdata_KeyWidget_t . . . 1589
qt_meta_stringdata_KPerfWidget_t . . . 1591
qt_meta_stringdata_LayoutDialog_t . . . 1594
qt_meta_stringdata_MacroReader_t . . . 1597
qt_meta_stringdata_MacroReaderThread_t . . . 1597
qt_meta_stringdata_MainWindow_t . . . 1601
qt_meta_stringdata_ModeSelectDialog_t . . . 1603
qt_meta_stringdata_MPerfWidget_t . . . 1605
qt_meta_stringdata_QuaGzipFile_t . . . 1608
qt_meta_stringdata_QuaZIODevice_t . . . 1610
qt_meta_stringdata_QuaZipFile_t . . . 1612
qt_meta_stringdata_RebindWidget_t . . . 1614
qt_meta_stringdata_RListWidget_t . . . 1617
qt_meta_stringdata_SettingsWidget_t . . . 1619
QThread . . . 802
Kb . . . 256

MacroReader . . . 720
MacroReaderThread . . . 724
QuaChecksum32 . . . 806
QuaAdler32 . . . 803
QuaCrc32 . . . 809
QuaZip . . . 827
QuaZipDir . . . 859
QuaZipFileInfo . . . 900
QuaZipFileInfo64 . . . 903
QuaZipFilePrivate . . . 911
QuaZipNewInfo . . . 916
QuaZipPrivate . . . 924
QWidget . . . 933
GradientDialogWidget . . . 222
KbAnimWidget . . . 343
KbBindWidget . . . 398
KbLightWidget . . . 459
KbWidget . . . 565
KeyWidget . . . 656
KPerfWidget . . . 700
MPerfWidget . . . 776
RebindWidget . . . 965
SettingsWidget . . . 1006
SGroup . . . 1030
tm_unz_s . . . 1436
tm_zip_s . . . 1512
Ui_AnimAddDialog . . . 1033
Ui::AnimAddDialog . . . 47
Ui_AnimSettingDialog . . . 1037
Ui::AnimSettingDialog . . . 84
Ui_ExtraSettingsWidget . . . 1050
Ui::ExtraSettingsWidget . . . 158
Ui_FwUpgradeDialog . . . 1062
Ui::FwUpgradeDialog . . . 189
Ui_GradientDialog . . . 1068
Ui::GradientDialog . . . 201
Ui_KbAnimWidget . . . 1076
Ui::KbAnimWidget . . . 333
Ui_KbBindWidget . . . 1085
Ui::KbBindWidget . . . 391
Ui_KbLightWidget . . . 1091
Ui::KbLightWidget . . . 450
Ui_KbProfileDialog . . . 1099
Ui::KbProfileDialog . . . 560
Ui_KbWidget . . . 1103
Ui::KbWidget . . . 587
Ui_KPerfWidget . . . 1113
Ui::KPerfWidget . . . 679
Ui_LayoutDialog . . . 1132
Ui::LayoutDialog . . . 712
Ui_MainWindow . . . 1137
Ui::MainWindow . . . 728
Ui_ModeSelectDialog . . . 1141

Ui::ModeSelectDialog 747
Ui_MPerfWidget 1145
Ui::MPerfWidget 755
Ui_RebindWidget 1165
Ui::RebindWidget 934
Ui_SettingsWidget 1195
Ui::SettingsWidget 1018
unz64_file_pos_s 1436
unz64_s 1385
unz_file_info64_internal_s 1385
unz_file_info64_s 1436
unz_file_info_s 1436
unz_file_pos_s 1436
unz_global_info64_s 1436
unz_global_info_s 1436
UsbId 1207
zip64_internal 1456
zip_fileinfo 1512
zlib_filefunc64_32_def_s 1338
zlib_filefunc64_def_s 1338
zlib_filefunc_def_s 1338

Chapter 9

Data Structure Index

9.1 Data Structures

Here are the data structures with brief descriptions:

AnimAddDialog . . . 41
Ui::AnimAddDialog . . . 47
AnimScript . . . 52
Ui::AnimSettingDialog . . . 84
AnimSettingDialog . . . 97
AutoRun . . . 118
CkbSettings . . . 122
CkbSettingsWriter . . . 137
ColorButton . . . 139
ColorMap . . . 149
Ui::ExtraSettingsWidget . . . 158
ExtraSettingsWidget . . . 170
KbFirmware::FW . . . 179
FwUpgradeDialog . . . 180
Ui::FwUpgradeDialog . . . 189
GradientButton . . . 195
Ui::GradientDialog . . . 201
GradientDialog . . . 209
GradientDialogWidget . . . 222
JlCompress
    Utility class for typical operations . . . 238
Kb . . . 256
KbAnim . . . 306
Ui::KbAnimWidget . . . 333
KbAnimWidget . . . 343
KbBind . . . 361
Ui::KbBindWidget . . . 391
KbBindWidget . . . 398
KbFirmware . . . 409
KbLight . . . 421
Ui::KbLightWidget . . . 450
KbLightWidget . . . 459
KbManager . . . 472
KbMode . . . 484
KbPerf . . . 498
KbProfile . . . 533
KbProfileDialog . . . 551
Ui::KbProfileDialog . . . 560

KbWidget . . . 565
Ui::KbWidget . . . 587
Key . . . 597
KeyAction . . . 600
KeyMap . . . 631
KeyWidget . . . 656
Ui::KPerfWidget . . . 679
KPerfWidget . . . 700
Ui::LayoutDialog . . . 712
LayoutDialog . . . 717
MacroReader
Creates a worker thread object. It does a connect do delayed deletion of thread local variables in the case the worker thread terminates . . . 720
MacroReaderThread
Responsible for reading Macro Key Values. It is created as a separate thread (worker thread) for reading macro commands from an fresh opened notify channel. Standard notify channel for macro definitions is number 2 . . . 724
Ui::MainWindow . . . 728
MainWindow . . . 732
Ui::ModeSelectDialog . . . 747
ModeSelectDialog . . . 751
Ui::MPerfWidget . . . 755
MPerfWidget . . . 776
GradientDialog::Preset . . . 793
QDialog . . . 795
QIODevice . . . 796
QIODevice_descriptor . . . 797
QListWidget . . . 798
QMainWindow . . . 799
QObject . . . 800
QPushButton . . . 801
QThread . . . 802
QuaAdler32
Adler32 checksum . . . 803
QuaChecksum32
Checksum interface . . . 806
QuaCrc32
CRC32 checksum . . . 809
QuaGzipFile
GZIP file . . . 812
QuaZIODevice
A class to compress/decompress QIODevice . . . 819
QuaZip
ZIP archive . . . 827
QuaZipDir
Provides ZIP archive navigation . . . 859
QuaZipFile
A file inside ZIP archive . . . 877
QuaZipFileInfo
Information about a file inside archive . . . 900
QuaZipFileInfo64
Information about a file inside archive (with zip64 support) . . . 903
QuaZipFilePrivate
The implementation class for QuaZip . . . 911
QuaZipNewInfo
Information about a file to be created . . . 916
QuaZipPrivate
All the internal stuff for the QuaZip class . . . 924

QWidget . . . 933
Ui::RebindWidget . . . 934
RebindWidget . . . 965
RListWidget . . . 1001
SettingsWidget . . . 1006
Ui::SettingsWidget . . . 1018
SGroup . . . 1030
Ui_AnimAddDialog . . . 1033
Ui_AnimSettingDialog . . . 1037
Ui_ExtraSettingsWidget . . . 1050
Ui_FwUpgradeDialog . . . 1062
Ui_GradientDialog . . . 1068
Ui_KbAnimWidget . . . 1076
Ui_KbBindWidget . . . 1085
Ui_KbLightWidget . . . 1091
Ui_KbProfileDialog . . . 1099
Ui_KbWidget . . . 1103
Ui_KPerfWidget . . . 1113
Ui_LayoutDialog . . . 1132
Ui_MainWindow . . . 1137
Ui_ModeSelectDialog . . . 1141
Ui_MPerfWidget . . . 1145
Ui_RebindWidget . . . 1165
Ui_SettingsWidget . . . 1195
UsbId . . . 1207

Chapter 10

File Index

10.1 File List

Here is a list of all files with brief descriptions:

src/ckb/animadddialog.cpp . . . 1213
src/ckb/animadddialog.h . . . 1213
src/ckb/animscript.cpp . . . 1214
src/ckb/animscript.h . . . 1216
src/ckb/animsettingdialog.cpp . . . 1217
src/ckb/animsettingdialog.h . . . 1219
src/ckb/autorun.cpp . . . 1219
src/ckb/autorun.h . . . 1220
src/ckb/ckb-anim.h . . . 1221
src/ckb/ckbsettings.cpp . . . 1240
src/ckb/ckbsettings.h . . . 1242
src/ckb/ckbsettingswriter.cpp . . . 1243
src/ckb/ckbsettingswriter.h . . . 1245
src/ckb/colorbutton.cpp . . . 1245
src/ckb/colorbutton.h . . . 1246
src/ckb/colormap.cpp . . . 1247
src/ckb/colormap.h . . . 1248
src/ckb/extrasettingswidget.cpp . . . 1249
src/ckb/extrasettingswidget.h . . . 1250
src/ckb/fwupgradedialog.cpp . . . 1250
src/ckb/fwupgradedialog.h . . . 1253
src/ckb/gradientbutton.cpp . . . 1254
src/ckb/gradientbutton.h . . . 1255
src/ckb/gradientdialog.cpp . . . 1256
src/ckb/gradientdialog.h . . . 1256
src/ckb/gradientdialogwidget.cpp . . . 1257
src/ckb/gradientdialogwidget.h . . . 1257
src/ckb/kb.cpp . . . 1258
src/ckb/kb.h . . . 1259
src/ckb/kbanim.cpp . . . 1260
src/ckb/kbanim.h . . . 1262
src/ckb/kbanimwidget.cpp . . . 1263
src/ckb/kbanimwidget.h . . . 1263
src/ckb/kbbind.cpp . . . 1264
src/ckb/kbbind.h . . . 1265
src/ckb/kbbindwidget.cpp . . . 1266
src/ckb/kbbindwidget.h . . . 1266
src/ckb/kbfirmware.cpp . . . 1267

src/ckb/kbfirmware.h . . . 1268
src/ckb/kblight.cpp . . . 1269
src/ckb/kblight.h . . . 1272
src/ckb/kblightwidget.cpp . . . 1273
src/ckb/kblightwidget.h . . . 1273
src/ckb/kbmanager.cpp . . . 1274
src/ckb/kbmanager.h . . . 1275
src/ckb/kbmode.cpp . . . 1276
src/ckb/kbmode.h . . . 1276
src/ckb/kbperf.cpp . . . 1277
src/ckb/kbperf.h . . . 1277
src/ckb/kbprofile.cpp . . . 1278
src/ckb/kbprofile.h . . . 1279
src/ckb/kbprofiledialog.cpp . . . 1279
src/ckb/kbprofiledialog.h . . . 1280
src/ckb/kbwidget.cpp . . . 1281
src/ckb/kbwidget.h . . . 1281
src/ckb/keyaction.cpp . . . 1282
src/ckb/keyaction.h . . . 1283
src/ckb/keymap.cpp . . . 1284
src/ckb/keymap.h . . . 1306
src/ckb/keywidget.cpp . . . 1307
src/ckb/keywidget.h . . . 1309
src/ckb/kperfwidget.cpp . . . 1310
src/ckb/kperfwidget.h . . . 1310
src/ckb/layoutdialog.cpp . . . 1312
src/ckb/layoutdialog.h . . . 1313
src/ckb/macroreader.cpp . . . 1314
src/ckb/macroreader.h . . . 1314
src/ckb/main.cpp . . . 1315
src/ckb/mainwindow.cpp . . . 1321
src/ckb/mainwindow.h . . . 1322
src/ckb/media.h . . . 1324
src/ckb/media_linux.cpp . . . 1326
src/ckb/media_mac.m . . . 1327
src/ckb/modeselectdialog.cpp . . . 1328
src/ckb/modeselectdialog.h . . . 1329
src/ckb/mperfwidget.cpp . . . 1330
src/ckb/mperfwidget.h . . . 1332
src/ckb/rebindwidget.cpp . . . 1523
src/ckb/rebindwidget.h . . . 1523
src/ckb/rlistwidget.cpp . . . 1630
src/ckb/rlistwidget.h . . . 1630
src/ckb/settingswidget.cpp . . . 1631
src/ckb/settingswidget.h . . . 1633
src/ckb/ui_animadddialog.h . . . 1634
src/ckb/ui_animsettingdialog.h . . . 1635
src/ckb/ui_extrasettingswidget.h . . . 1635
src/ckb/ui_fwupgradedialog.h . . . 1636
src/ckb/ui_gradientdialog.h . . . 1637
src/ckb/ui_kbanimwidget.h . . . 1638
src/ckb/ui_kbbindwidget.h . . . 1639
src/ckb/ui_kblightwidget.h . . . 1640
src/ckb/ui_kbprofiledialog.h . . . 1641
src/ckb/ui_kbwidget.h . . . 1642
src/ckb/ui_kperfwidget.h . . . 1643
src/ckb/ui_layoutdialog.h . . . 1644
src/ckb/ui_mainwindow.h . . . 1645

src/ckb/ui_modeselectdialog.h . . . 1646
src/ckb/ui_mperfwidget.h . . . 1647
src/ckb/ui_rebindwidget.h . . . 1648
src/ckb/ui_settingswidget.h . . . 1649
src/ckb/quazip/crypt.h . . . 1334
src/ckb/quazip/ioapi.h . . . 1338
src/ckb/quazip/JlCompress.cpp . . . 1345
src/ckb/quazip/JlCompress.h . . . 1347
src/ckb/quazip/qioapi.cpp . . . 1347
src/ckb/quazip/quaadler32.cpp . . . 1359
src/ckb/quazip/quaadler32.h . . . 1360
src/ckb/quazip/quachecksum32.h . . . 1361
src/ckb/quazip/quacrc32.cpp . . . 1362
src/ckb/quazip/quacrc32.h . . . 1363
src/ckb/quazip/quagzipfile.cpp . . . 1364
src/ckb/quazip/quagzipfile.h . . . 1365
src/ckb/quazip/quaziodevice.cpp . . . 1366
src/ckb/quazip/quaziodevice.h . . . 1367
src/ckb/quazip/quazip.cpp . . . 1368
src/ckb/quazip/quazip.h . . . 1370
src/ckb/quazip/quazip_global.h . . . 1371
src/ckb/quazip/quazipdir.cpp . . . 1373
src/ckb/quazip/quazipdir.h . . . 1375
src/ckb/quazip/quazipfile.cpp . . . 1375
src/ckb/quazip/quazipfile.h . . . 1376
src/ckb/quazip/quazipfileinfo.cpp . . . 1377
src/ckb/quazip/quazipfileinfo.h . . . 1379
src/ckb/quazip/quazipnewinfo.cpp . . . 1380
src/ckb/quazip/quazipnewinfo.h . . . 1384
src/ckb/quazip/unzip.c . . . 1385
src/ckb/quazip/unzip.h . . . 1436
src/ckb/quazip/zip.c . . . 1456
src/ckb/quazip/zip.h . . . 1512
src/ckb/release/moc_animadddialog.cpp . . . 1524
src/ckb/release/moc_animscript.cpp . . . 1527
src/ckb/release/moc_animsettingdialog.cpp . . . 1530
src/ckb/release/moc_ckbsettingswriter.cpp . . . 1533
src/ckb/release/moc_colorbutton.cpp . . . 1535
src/ckb/release/moc_extrasettingswidget.cpp . . . 1537
src/ckb/release/moc_fwupgradedialog.cpp . . . 1540
src/ckb/release/moc_gradientbutton.cpp . . . 1543
src/ckb/release/moc_gradientdialog.cpp . . . 1546
src/ckb/release/moc_gradientdialogwidget.cpp . . . 1549
src/ckb/release/moc_kb.cpp . . . 1551
src/ckb/release/moc_kbanim.cpp . . . 1554
src/ckb/release/moc_kbanimwidget.cpp . . . 1556
src/ckb/release/moc_kbbind.cpp . . . 1559
src/ckb/release/moc_kbbindwidget.cpp . . . 1561
src/ckb/release/moc_kbfirmware.cpp . . . 1563
src/ckb/release/moc_kblight.cpp . . . 1566
src/ckb/release/moc_kblightwidget.cpp . . . 1569
src/ckb/release/moc_kbmanager.cpp . . . 1572
src/ckb/release/moc_kbmode.cpp . . . 1574
src/ckb/release/moc_kbperf.cpp . . . 1576
src/ckb/release/moc_kbprofile.cpp . . . 1579
src/ckb/release/moc_kbprofiledialog.cpp . . . 1581
src/ckb/release/moc_kbwidget.cpp . . . 1585
src/ckb/release/moc_keyaction.cpp . . . 1587

src/ckb/release/moc_keywidget.cpp . . . 1589
src/ckb/release/moc_kperfwidget.cpp . . . 1591
src/ckb/release/moc_layoutdialog.cpp . . . 1594
src/ckb/release/moc_macroreader.cpp . . . 1597
src/ckb/release/moc_mainwindow.cpp . . . 1601
src/ckb/release/moc_modeselectdialog.cpp . . . 1603
src/ckb/release/moc_mperfwidget.cpp . . . 1605
src/ckb/release/moc_quagzipfile.cpp . . . 1608
src/ckb/release/moc_quaziodevice.cpp . . . 1610
src/ckb/release/moc_quazipfile.cpp . . . 1612
src/ckb/release/moc_rebindwidget.cpp . . . 1614
src/ckb/release/moc_rlistwidget.cpp . . . 1617
src/ckb/release/moc_settingswidget.cpp . . . 1619
src/ckb/release/qrc_binary.cpp . . . 1621
src/ckb/release/qrc_image.cpp . . . 1624
src/ckb/release/qrc_text.cpp . . . 1627

Chapter 11

Namespace Documentation

11.1 Ui Namespace Reference

Data Structures

- class AnimAddDialog
- class AnimSettingDialog
- class ExtraSettingsWidget
- class FwUpgradeDialog
- class GradientDialog
- class KbAnimWidget
- class KbBindWidget
- class KbLightWidget
- class KbProfileDialog
- class KbWidget
- class KPerfWidget
- class LayoutDialog
- class MainWindow
- class ModeSelectDialog
- class MPerfWidget
- class RebindWidget
- class SettingsWidget

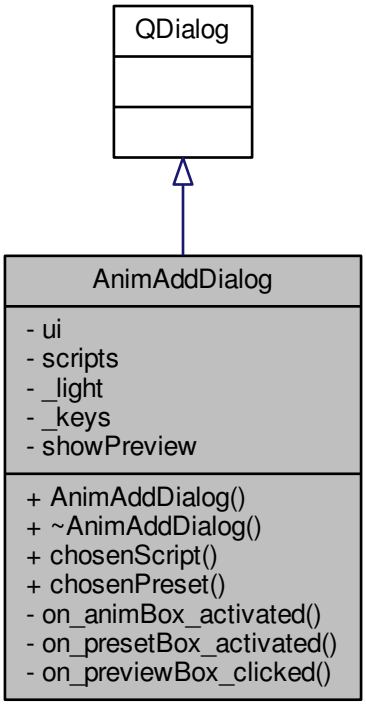
Chapter 12

Data Structure Documentation

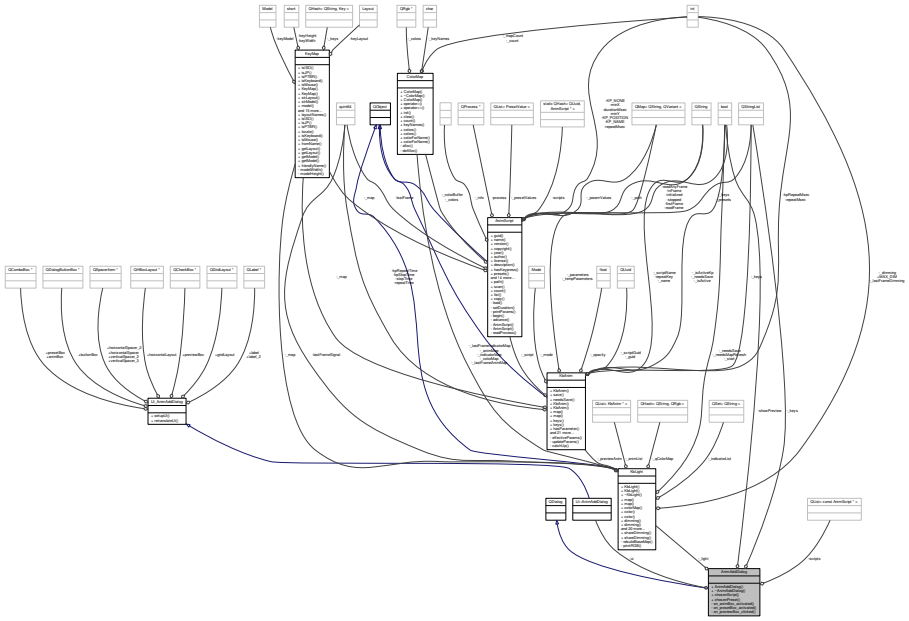
12.1 AnimAddDialog Class Reference

`#include <src/ckb/animadddialog.h>`

Inheritance diagram for AnimAddDialog:



Collaboration diagram for AnimAddDialog:



Public Member Functions

- AnimAddDialog (QWidget *parent, KbLight *light, const QStringList &keys)
- ~AnimAddDialog ()
- const AnimScript * chosenScript ()
- int chosenPreset ()

Private Slots

- void on_animBox_activated (int index)
- void on_presetBox_activated (int index)
- void on_previewBox_clicked (bool checked)

Private Attributes

- Ui::AnimAddDialog * ui
- QList< const AnimScript * > scripts
- KbLight * _light
- QStringList _keys
- bool showPreview

12.1.1 Detailed Description

Definition at line 12 of file animadddialog.h.

12.1.2 Constructor & Destructor Documentation

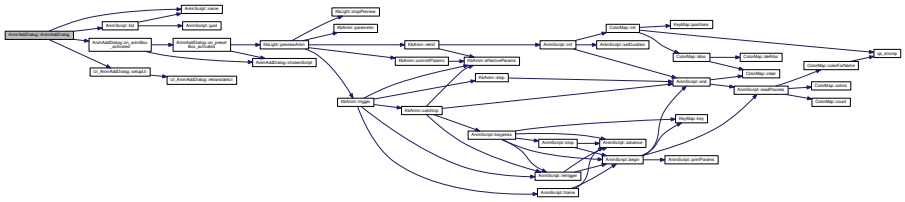
12.1.2.1 AnimAddDialog::AnimAddDialog (QWidget ∗ *parent*, KbLight ∗ *light*, const QStringList & *keys*) [explicit]

Definition at line 4 of file animadddialog.cpp.

References Ui_AnimAddDialog::animBox, AnimScript::list(), AnimScript::name(), on_animBox_activated(), scripts, Ui_AnimAddDialog::setupUi(), and ui.

```
4                                                                                    :
5     QDialog(parent),
6     ui(new Ui::AnimAddDialog), _light(light), _keys(keys),
      showPreview(true)
7 {
8     ui->setupUi(this);
9     scripts = AnimScript::list();
10     foreach(const AnimScript* script, scripts)
11         ui->animBox->addItem(script->name());
12     on_animBox_activated(0);
13 }
```

Here is the call graph for this function:



12.1.2.2 AnimAddDialog::∼AnimAddDialog ( )

Definition at line 15 of file animadddialog.cpp.

References _light, KbLight::stopPreview(), and ui.

```
15                                  {
16     delete ui;
17     _light->stopPreview();
18 }
```

Here is the call graph for this function:

12.1.3 Member Function Documentation

12.1.3.1 int AnimAddDialog::chosenPreset ( )

Definition at line 24 of file animadddialog.cpp.

References Ui_AnimAddDialog::presetBox, and ui.

Referenced by KbLightWidget::on_animButton_clicked().

```
24                                {
25     return ui->presetBox->currentIndex();
26 }
```

Here is the caller graph for this function:

12.1.3.2 const AnimScript * AnimAddDialog::chosenScript ( )

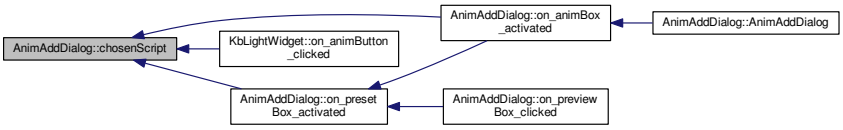
Definition at line 20 of file animadddialog.cpp.

References Ui_AnimAddDialog::animBox, scripts, and ui.

Referenced by on_animBox_activated(), KbLightWidget::on_animButton_clicked(), and on_presetBox_activated().

```
20                                          {
21     return scripts[ui->animBox->currentIndex()];
22 }
```

Here is the caller graph for this function:



12.1.3.3 void AnimAddDialog::on_animBox_activated (int *index*) [private], [slot]

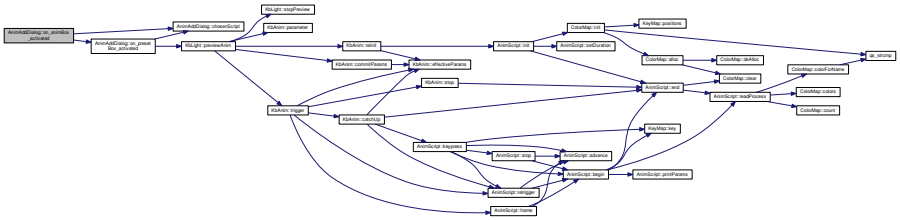
Definition at line 28 of file animadddialog.cpp.

References chosenScript(), on_presetBox_activated(), Ui_AnimAddDialog::presetBox, and ui.

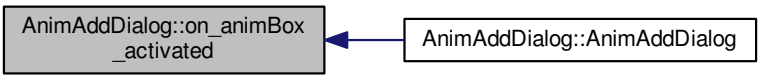
Referenced by AnimAddDialog().

```
28                                              {
29     // Update preset list
30     ui->presetBox->clear();
31     foreach(const QString& preset, chosenScript()->presets())
32         ui->presetBox->addItem(preset);
33     ui->presetBox->setCurrentIndex(0);
34     on_presetBox_activated(0);
35 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.1.3.4 void AnimAddDialog::on_presetBox_activated (int *index*) [private], [slot]

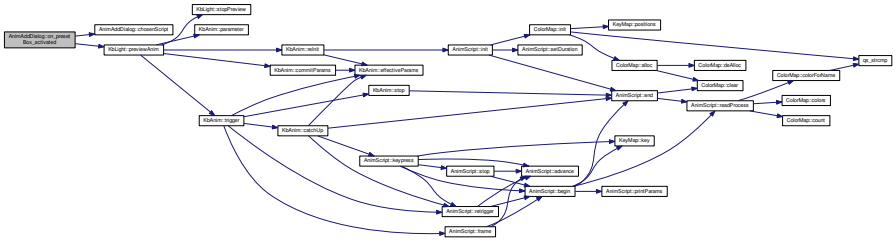
Definition at line 37 of file animadddialog.cpp.

References _keys, _light, chosenScript(), KbLight::previewAnim(), and showPreview.

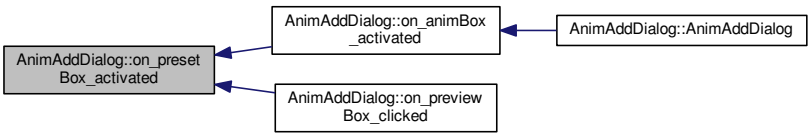
Referenced by on_animBox_activated(), and on_previewBox_clicked().

```
37                                                  {
38     // Update preview
39     if(showPreview)
40         _light->previewAnim(chosenScript(), _keys,
       chosenScript()->preset(index));
41 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



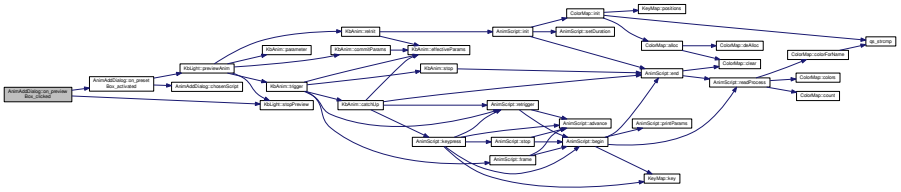
12.1.3.5 void AnimAddDialog::on_previewBox_clicked (bool *checked*) [private], [slot]

Definition at line 43 of file animadddialog.cpp.

References _light, on_presetBox_activated(), Ui_AnimAddDialog::presetBox, showPreview, KbLight::stopPreview(), and ui.

```
43                                                  {
44     showPreview = checked;
45     if(showPreview)
46         on_presetBox_activated(ui->presetBox->currentIndex());
47     else
48         _light->stopPreview();
49 }
```

Here is the call graph for this function:



12.1.4 Field Documentation

12.1.4.1 QStringList AnimAddDialog::_keys [private]

Definition at line 33 of file animadddialog.h.

Referenced by on_presetBox_activated().

12.1.4.2 KbLight* AnimAddDialog::_light [private]

Definition at line 32 of file animadddialog.h.

Referenced by on_presetBox_activated(), on_previewBox_clicked(), and ~AnimAddDialog().

12.1.4.3 QList<const AnimScript*> AnimAddDialog::scripts [private]

Definition at line 30 of file animadddialog.h.

Referenced by AnimAddDialog(), and chosenScript().

12.1.4.4 bool AnimAddDialog::showPreview [private]

Definition at line 34 of file animadddialog.h.

Referenced by on_presetBox_activated(), and on_previewBox_clicked().

12.1.4.5 Ui::AnimAddDialog* AnimAddDialog::ui [private]

Definition at line 29 of file animadddialog.h.

Referenced by AnimAddDialog(), chosenPreset(), chosenScript(), on_animBox_activated(), on_previewBox_clicked(), and ~AnimAddDialog().

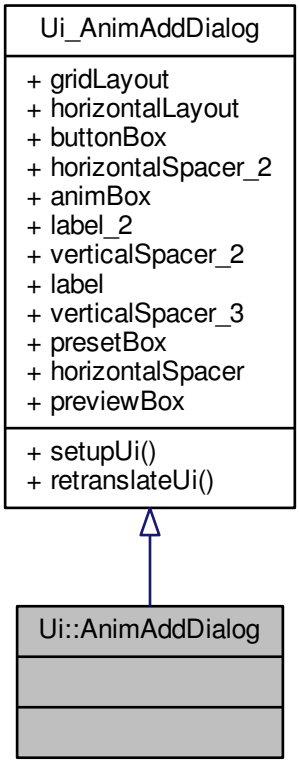
The documentation for this class was generated from the following files:

- src/ckb/animadddialog.h
- src/ckb/animadddialog.cpp

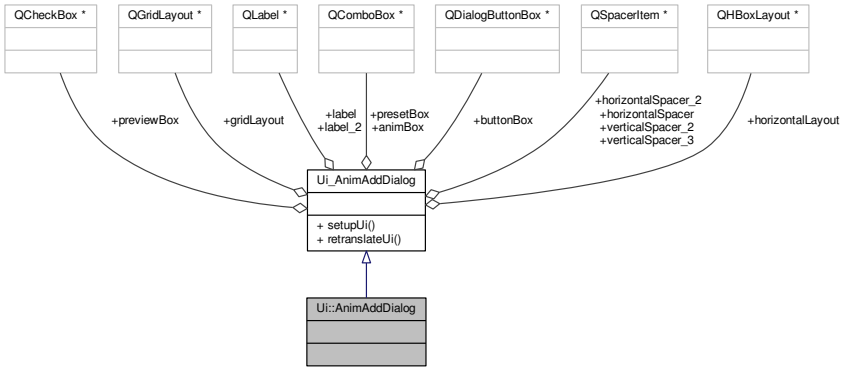
12.2 Ui::AnimAddDialog Class Reference

`#include <src/ckb/ui_animadddialog.h>`

Inheritance diagram for Ui::AnimAddDialog:



Collaboration diagram for Ui::AnimAddDialog:



Public Member Functions

- void setupUi (QDialog *AnimAddDialog)
- void retranslateUi (QDialog *AnimAddDialog)

Data Fields

- QGridLayout * gridLayout
- QHBoxLayout * horizontalLayout
- QDialogButtonBox * buttonBox
- QSpacerItem * horizontalSpacer_2
- QComboBox * animBox
- QLabel * label_2
- QSpacerItem * verticalSpacer_2
- QLabel * label
- QSpacerItem * verticalSpacer_3
- QComboBox * presetBox
- QSpacerItem * horizontalSpacer
- QCheckBox * previewBox

12.2.1 Detailed Description

Definition at line 146 of file ui_animadddialog.h.

12.2.2 Member Function Documentation

12.2.2.1 void Ui_AnimAddDialog::retranslateUi (QDialog * *AnimAddDialog*) [inline], [inherited]

Definition at line 135 of file ui_animadddialog.h.

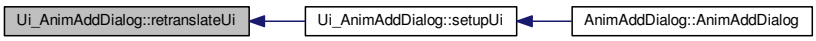
References Ui_AnimAddDialog::label, Ui_AnimAddDialog::label_2, and Ui_AnimAddDialog::previewBox.

Referenced by Ui_AnimAddDialog::setupUi().

```
136     {
137         AnimAddDialog->setWindowTitle(QApplication::translate("AnimAddDialog", "Add Animation", Q_NULLPTR))
    ;
138         label_2->setText(QApplication::translate("AnimAddDialog", "Preset:", Q_NULLPTR));
```

```
139         label->setText(QApplication::translate("AnimAddDialog", "Animation:", Q_NULLPTR));
140         previewBox->setText(QApplication::translate("AnimAddDialog", "Show preview", Q_NULLPTR));
141     } // retranslateUi
```

Here is the caller graph for this function:



12.2.2.2 void Ui_AnimAddDialog::setupUi (QDialog * *AnimAddDialog*) [inline],[inherited]

Definition at line 44 of file ui_animadddialog.h.

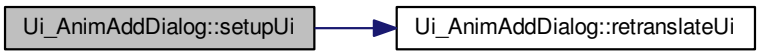
References Ui_AnimAddDialog::animBox, Ui_AnimAddDialog::buttonBox, Ui_AnimAddDialog::gridLayout, Ui_AnimAddDialog::horizontalLayout, Ui_AnimAddDialog::horizontalSpacer, Ui_AnimAddDialog::horizontalSpacer_2, Ui_AnimAddDialog::label, Ui_AnimAddDialog::label_2, Ui_AnimAddDialog::presetBox, Ui_AnimAddDialog::previewBox, Ui_AnimAddDialog::retranslateUi(), Ui_AnimAddDialog::verticalSpacer_2, and Ui_AnimAddDialog::verticalSpacer_3.

Referenced by AnimAddDialog::AnimAddDialog().

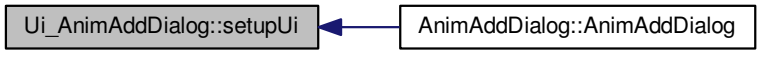
```
45     {
46         if (AnimAddDialog->objectName().isEmpty())
47             AnimAddDialog->setObjectName(QStringLiteral("AnimAddDialog"));
48         AnimAddDialog->resize(350, 205);
49         QSizePolicy sizePolicy(QSizePolicy::Fixed, QSizePolicy::Fixed);
50         sizePolicy.setHorizontalStretch(0);
51         sizePolicy.setVerticalStretch(0);
52         sizePolicy.setHeightForWidth(AnimAddDialog->sizePolicy().hasHeightForWidth());
53         AnimAddDialog->setSizePolicy(sizePolicy);
54         gridLayout = new QGridLayout(AnimAddDialog);
55         gridLayout->setObjectName(QStringLiteral("gridLayout"));
56         horizontalLayout = new QHBoxLayout();
57         horizontalLayout->setObjectName(QStringLiteral("horizontalLayout"));
58         buttonBox = new QDialogButtonBox(AnimAddDialog);
59         buttonBox->setObjectName(QStringLiteral("buttonBox"));
60         buttonBox->setOrientation(Qt::Horizontal);
61         buttonBox->setStandardButtons(QDialogButtonBox::Cancel|QDialogButtonBox::Ok);
62         buttonBox->setCenterButtons(false);
63
64         horizontalLayout->addWidget(buttonBox);
65
66
67         gridLayout->addLayout(horizontalLayout, 5, 0, 1, 4);
68
69         horizontalSpacer_2 = new QSpacerItem(50, 20, QSizePolicy::Fixed,
    QSizePolicy::Minimum);
70
71         gridLayout->addItem(horizontalSpacer_2, 1, 3, 1, 1);
72
73         animBox = new QComboBox(AnimAddDialog);
74         animBox->setObjectName(QStringLiteral("animBox"));
75         QSizePolicy sizePolicy1(QSizePolicy::Expanding, QSizePolicy::Fixed);
76         sizePolicy1.setHorizontalStretch(0);
77         sizePolicy1.setVerticalStretch(0);
78         sizePolicy1.setHeightForWidth(animBox->sizePolicy().hasHeightForWidth());
79         animBox->setSizePolicy(sizePolicy1);
80
81         gridLayout->addWidget(animBox, 1, 2, 1, 1);
82
83         label_2 = new QLabel(AnimAddDialog);
84         label_2->setObjectName(QStringLiteral("label_2"));
85         QSizePolicy sizePolicy2(QSizePolicy::Preferred, QSizePolicy::Preferred);
86         sizePolicy2.setHorizontalStretch(0);
87         sizePolicy2.setVerticalStretch(0);
88         sizePolicy2.setHeightForWidth(label_2->sizePolicy().hasHeightForWidth());
89         label_2->setSizePolicy(sizePolicy2);
90         label_2->setAlignment(Qt::AlignRight|Qt::AlignTrailing|Qt::AlignVCenter);
91
92         gridLayout->addWidget(label_2, 2, 1, 1, 1);
93
```

```
94          verticalSpacer_2 = new QSpacerItem(20, 0, QSizePolicy::Minimum,
    QSizePolicy::Expanding);
95
96          gridLayout->addItem(verticalSpacer_2, 4, 2, 1, 1);
97
98          label = new QLabel(AnimAddDialog);
99          label->setObjectName(QStringLiteral("label"));
100          sizePolicy2.setHeightForWidth(label->sizePolicy().hasHeightForWidth());
101          label->setSizePolicy(sizePolicy2);
102          label->setAlignment(Qt::AlignRight|Qt::AlignTrailing|Qt::AlignVCenter);
103
104          gridLayout->addWidget(label, 1, 1, 1, 1);
105
106          verticalSpacer_3 = new QSpacerItem(20, 0, QSizePolicy::Minimum,
    QSizePolicy::Expanding);
107
108          gridLayout->addItem(verticalSpacer_3, 0, 2, 1, 1);
109
110          presetBox = new QComboBox(AnimAddDialog);
111          presetBox->setObjectName(QStringLiteral("presetBox"));
112          sizePolicy1.setHeightForWidth(presetBox->sizePolicy().hasHeightForWidth());
113          presetBox->setSizePolicy(sizePolicy1);
114
115          gridLayout->addWidget(presetBox, 2, 2, 1, 1);
116
117          horizontalSpacer = new QSpacerItem(50, 20, QSizePolicy::Fixed, QSizePolicy::Minimum
    );
118
119          gridLayout->addItem(horizontalSpacer, 1, 0, 1, 1);
120
121          previewBox = new QCheckBox(AnimAddDialog);
122          previewBox->setObjectName(QStringLiteral("previewBox"));
123          previewBox->setChecked(true);
124
125          gridLayout->addWidget(previewBox, 3, 2, 1, 1);
126
127
128          retranslateUi(AnimAddDialog);
129          QObject::connect(buttonBox, SIGNAL(accepted()), AnimAddDialog, SLOT(accept()));
130          QObject::connect(buttonBox, SIGNAL(rejected()), AnimAddDialog, SLOT(reject()));
131
132          QMetaObject::connectSlotsByName(AnimAddDialog);
133      } // setupUi
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.2.3 Field Documentation

12.2.3.1 QComboBox* Ui_AnimAddDialog::animBox [inherited]

Definition at line 35 of file ui_animadddialog.h.

Referenced by AnimAddDialog::AnimAddDialog(), AnimAddDialog::chosenScript(), and Ui_AnimAddDialog::setupUi().

12.2.3.2 QDialogButtonBox* Ui_AnimAddDialog::buttonBox [inherited]

Definition at line 33 of file ui_animadddialog.h.

Referenced by Ui_AnimAddDialog::setupUi().

12.2.3.3 QGridLayout* Ui_AnimAddDialog::gridLayout [inherited]

Definition at line 31 of file ui_animadddialog.h.

Referenced by Ui_AnimAddDialog::setupUi().

12.2.3.4 QHBoxLayout* Ui_AnimAddDialog::horizontalLayout [inherited]

Definition at line 32 of file ui_animadddialog.h.

Referenced by Ui_AnimAddDialog::setupUi().

12.2.3.5 QSpacerItem* Ui_AnimAddDialog::horizontalSpacer [inherited]

Definition at line 41 of file ui_animadddialog.h.

Referenced by Ui_AnimAddDialog::setupUi().

12.2.3.6 QSpacerItem* Ui_AnimAddDialog::horizontalSpacer_2 [inherited]

Definition at line 34 of file ui_animadddialog.h.

Referenced by Ui_AnimAddDialog::setupUi().

12.2.3.7 QLabel* Ui_AnimAddDialog::label [inherited]

Definition at line 38 of file ui_animadddialog.h.

Referenced by Ui_AnimAddDialog::retranslateUi(), and Ui_AnimAddDialog::setupUi().

12.2.3.8 QLabel* Ui_AnimAddDialog::label_2 [inherited]

Definition at line 36 of file ui_animadddialog.h.

Referenced by Ui_AnimAddDialog::retranslateUi(), and Ui_AnimAddDialog::setupUi().

12.2.3.9 QComboBox* Ui_AnimAddDialog::presetBox [inherited]

Definition at line 40 of file ui_animadddialog.h.

Referenced by AnimAddDialog::chosenPreset(), AnimAddDialog::on_animBox_activated(), AnimAddDialog::on_previewBox_clicked(), and Ui_AnimAddDialog::setupUi().

12.2.3.10 QCheckBox* Ui_AnimAddDialog::previewBox [inherited]

Definition at line 42 of file ui_animadddialog.h.

Referenced by Ui_AnimAddDialog::retranslateUi(), and Ui_AnimAddDialog::setupUi().

12.2.3.11 QSpacerItem* Ui_AnimAddDialog::verticalSpacer_2 [inherited]

Definition at line 37 of file ui_animadddialog.h.

Referenced by Ui_AnimAddDialog::setupUi().

12.2.3.12 QSpacerItem* Ui_AnimAddDialog::verticalSpacer_3 [inherited]

Definition at line 39 of file ui_animadddialog.h.

Referenced by Ui_AnimAddDialog::setupUi().

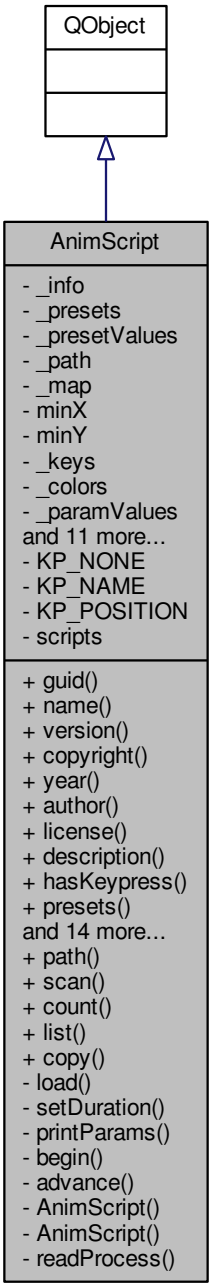
The documentation for this class was generated from the following file:

- src/ckb/ui_animadddialog.h

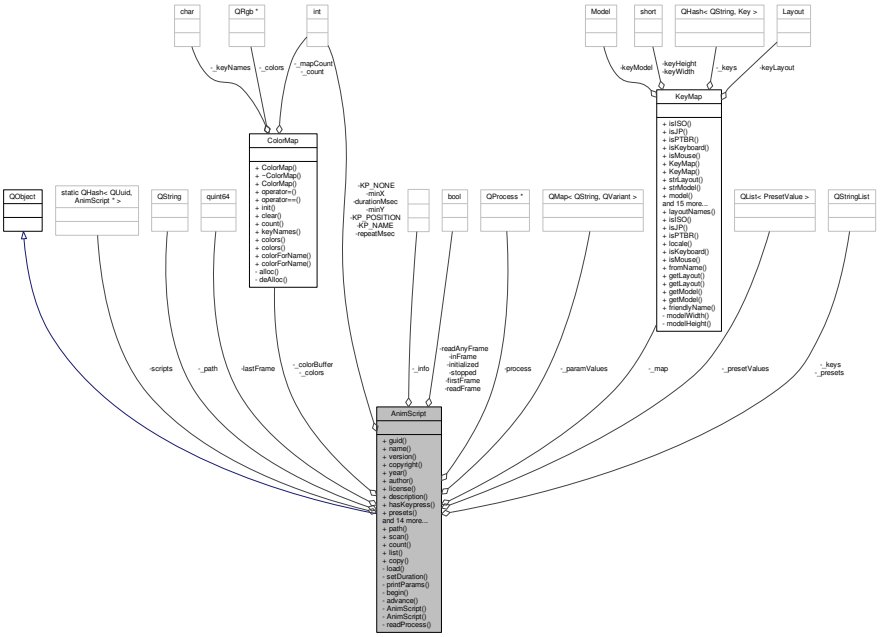
12.3 AnimScript Class Reference

`#include <src/ckb/animscript.h>`

Inheritance diagram for AnimScript:



Collaboration diagram for AnimScript:



Data Structures

- struct Param

Public Types

- typedef QMap< QString, QVariant > PresetValue

Public Member Functions

- const QUuid & guid () const
- const QString & name () const
- const QString & version () const
- QString copyright () const
- const QString & year () const
- const QString & author () const
- const QString & license () const
- const QString & description () const
- bool hasKeypress () const
- const QStringList & presets () const
- const PresetValue & preset (int index) const
- QListIterator< Param > paramIterator () const
- Param param (const QString &name) const
- bool hasParam (const QString &name) const
- void init (const KeyMap &map, const QStringList &keys, const QMap< QString, QVariant > &paramValues)
- void parameters (const QMap< QString, QVariant > &paramValues)
- void retrigger (quint64 timestamp, bool allowPreempt=false)
- void stop (quint64 timestamp)
- void keypress (const QString &key, bool pressed, quint64 timestamp)

- void frame (quint64 timestamp)
- void end ()
- bool hasFrame () const
- const ColorMap & colors () const
- ~AnimScript ()

Static Public Member Functions

- static QString path ()
- static void scan ()
- static int count ()
- static QList< const AnimScript * > list ()
- static AnimScript * copy (QObject *parent, const QUuid &id)

Private Slots

- void readProcess ()

Private Member Functions

- bool load ()
- void setDuration ()
- void printParams ()
- void begin (quint64 timestamp)
- void advance (quint64 timestamp)
- AnimScript (QObject *parent, const QString &path)
- AnimScript (QObject *parent, const AnimScript &base)

Private Attributes

- struct {
  QUuid **guid**
  QString **name**
  QString **version**
  QString **year**
  QString **author**
  QString **license**
  QString **description**
  QList< Param > **params**
  int **kpMode**:3
  bool **absoluteTime**:1
  bool **repeat**:1
  bool **preempt**:1
  bool **liveParams**:1
  } _info

- QStringList _presets
- QList< PresetValue > _presetValues
- QString _path
- KeyMap _map
- int minX
- int minY
- QStringList _keys

- ColorMap _colors
- PresetValue _paramValues
- quint64 lastFrame
- int durationMsec
- int repeatMsec
- bool initialized:1
- bool firstFrame:1
- bool readFrame:1
- bool readAnyFrame:1
- bool stopped:1
- bool inFrame:1
- QProcess * process
- ColorMap _colorBuffer

Static Private Attributes

- static const int KP_NONE = 0
- static const int KP_NAME = 1
- static const int KP_POSITION = 2
- static QHash< QUuid, AnimScript * > scripts

12.3.1 Detailed Description

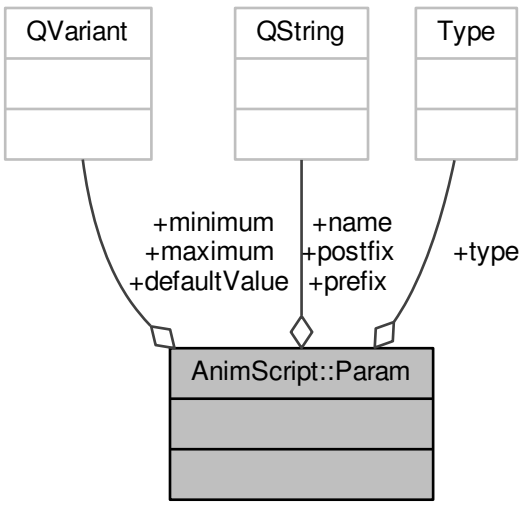
Definition at line 16 of file animscript.h.

12.3.2 Data Structure Documentation

12.3.2.1 struct AnimScript::Param

Definition at line 21 of file animscript.h.

Collaboration diagram for AnimScript::Param:



Data Fields

	Type	

Data Fields

QVariant	defaultValue	
QVariant	maximum	
QVariant	minimum	
QString	name	
QString	postfix	
QString	prefix	
Type	type	

12.3.3 Member Typedef Documentation

12.3.3.1 typedef QMap<QString, QVariant> AnimScript::PresetValue

Definition at line 58 of file animscript.h.

12.3.4 Constructor & Destructor Documentation

12.3.4.1 AnimScript::~AnimScript ( )

Definition at line 21 of file animscript.cpp.

References process.

```
21                          {
22     if(process){
23         process->kill();
24         process->waitForFinished(1000);
25         delete process;
26     }
27 }
```

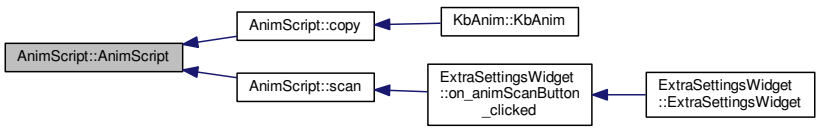
12.3.4.2 AnimScript::AnimScript (QObject * *parent*, const QString & *path*) [private]

Definition at line 11 of file animscript.cpp.

Referenced by copy(), and scan().

```
11                                                          :
12     QObject(parent), _path(path), initialized(false),
       process(0)
13 {
14 }
```

Here is the caller graph for this function:



12.3.4.3 AnimScript::AnimScript (QObject * *parent*, const AnimScript & *base*) [private]

Definition at line 16 of file animscript.cpp.

```
16                                                                  :
17     QObject(parent), _info(base._info), _path(base._path),
       initialized(false), process(0)
18 {
19 }
```

12.3.5 Member Function Documentation

12.3.5.1 void AnimScript::advance (quint64 *timestamp*) [private]

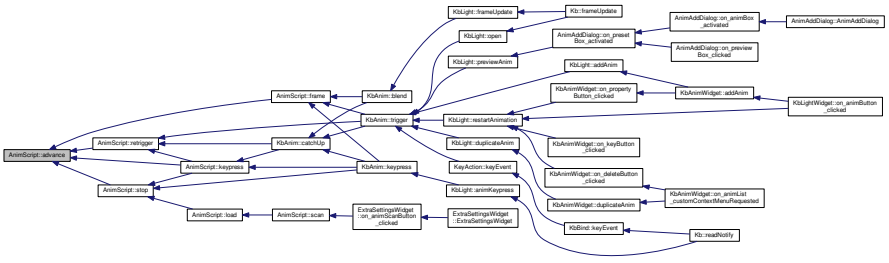
Definition at line 445 of file animscript.cpp.

References _info, durationMsec, lastFrame, and process.

Referenced by frame(), keypress(), retrigger(), and stop().

```
445                                          {
446     if(timestamp <= lastFrame || !process)
447         // Don't do anything if the time hasn't actually advanced.
448         return;
449     double delta = (timestamp - lastFrame) / (double)durationMsec;
450     if(!_info.absoluteTime){
451         // Skip any complete durations
452         while(delta > 1.){
453             process->write("time 1\n");
454             delta--;
455         }
456     }
457     process->write(QString("time %1\n").arg(delta).toLatin1());
458     lastFrame = timestamp;
459 }
```

Here is the caller graph for this function:



12.3.5.2 const QString& AnimScript::author ( ) const [inline]

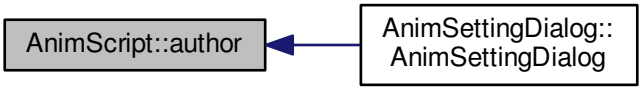
Definition at line 64 of file animscript.h.

References _info.

Referenced by AnimSettingDialog::AnimSettingDialog().

```
64 { return _info.author; }
```

Here is the caller graph for this function:



12.3.5.3 void AnimScript::begin (quint64 *timestamp*) [private]

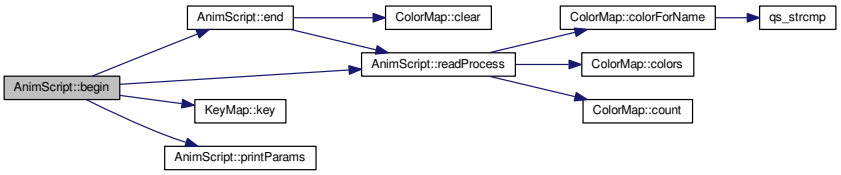
Definition at line 287 of file animscript.cpp.

References _keys, _map, _path, end(), firstFrame, inFrame, initialized, KeyMap::key(), lastFrame, minX, minY, printParams(), process, readAnyFrame, readFrame, readProcess(), stopped, Key::x, and Key::y.

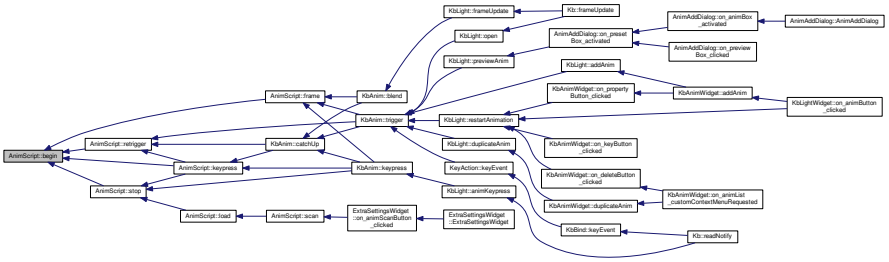
Referenced by frame(), keypress(), retrigger(), and stop().

```
287                                     {
288     if(!initialized)
289         return;
290     end();
291     stopped = firstFrame = readFrame = readAnyFrame =
      inFrame = false;
292     // Determine the upper left corner of the given keys
293     QStringList keysCopy = _keys;
294     minX = INT_MAX;
295     minY = INT_MAX;
296     foreach(const QString& key, keysCopy){
297         const Key& pos = _map.key(key);
298         if(!pos){
299             keysCopy.removeAll(key);
300             continue;
301         }
302         if(pos.x < minX)
303             minX = pos.x;
304         if(pos.y < minY)
305             minY = pos.y;
306     }
307     if(keysCopy.isEmpty()){
308         // If the key list is empty, don't actually start the animation but pretend it's running anyway
309         firstFrame = readFrame = readAnyFrame = true;
310         return;
311     }
312     process = new QProcess(this);
313     connect(process, SIGNAL(readyRead()), this, SLOT(readProcess()));
314     process->start(_path, QStringList("--ckb-run"));
315     qDebug() << "Starting " << _path;
316     // Write the keymap to the process
317     process->write("begin keymap\n");
318     process->write(QString("keycount %1\n").arg(keysCopy.count()).toLatin1());
319     foreach(const QString& key, keysCopy){
320         const Key& pos = _map.key(key);
321         process->write(QString("key %1 %2,%3\n").arg(key).arg(pos.x -
      minX).arg(pos.y - minY).toLatin1());
322     }
323     process->write("end keymap\n");
324     // Write parameters
325     printParams();
326     // Begin animating
327     process->write("begin run\n");
328     lastFrame = timestamp;
329 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.3.5.4 const ColorMap& AnimScript::colors ( ) const [inline]

Definition at line 99 of file animscript.h.

References _colors.

Referenced by KbAnim::blend().

```
99 { return _colors; }
```

Here is the caller graph for this function:

12.3.5.5 AnimScript * AnimScript::copy (QObject * *parent*, const QUuid & *id*) [static]

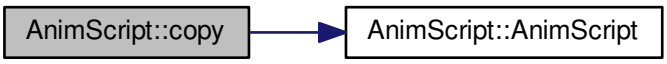
Definition at line 67 of file animscript.cpp.

References AnimScript(), and scripts.

Referenced by KbAnim::KbAnim().

```
67                                                      {
68     return scripts.contains(id) ? new AnimScript(parent, *
    scripts.value(id)) : 0;
69 }
```

Here is the call graph for this function:



Here is the caller graph for this function:

12.3.5.6 QString AnimScript::copyright ( ) const [inline]

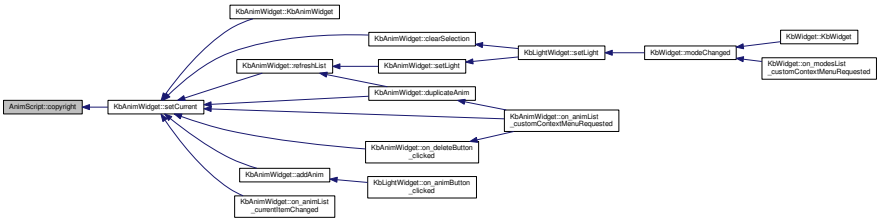
Definition at line 62 of file animscript.h.

References _info.

Referenced by KbAnimWidget::setCurrent().

```
62 { return "Copyright © " + _info.year + " " + _info.author; }
```

Here is the caller graph for this function:



12.3.5.7 static int AnimScript::count ( ) [inline], [static]

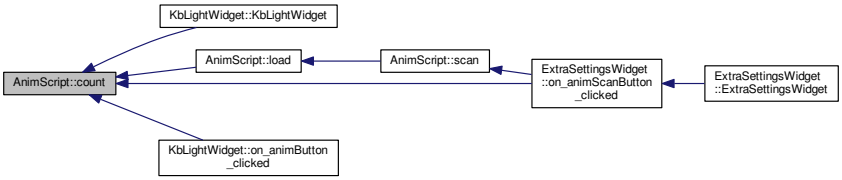
Definition at line 54 of file animscript.h.

References scripts.

Referenced by KbLightWidget::KbLightWidget(), load(), KbLightWidget::on_animButton_clicked(), and ExtraSettingsWidget::on_animScanButton_clicked().

```
54 { return scripts.count(); }
```

Here is the caller graph for this function:



12.3.5.8 const QString& AnimScript::description ( ) const [inline]

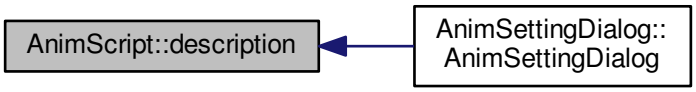
Definition at line 66 of file animscript.h.

References _info.

Referenced by AnimSettingDialog::AnimSettingDialog().

```
66 { return _info.description; }
```

Here is the caller graph for this function:



12.3.5.9 void AnimScript::end ( )

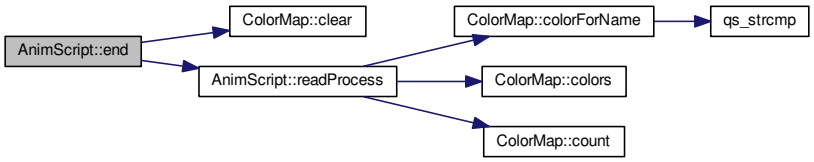
Definition at line 387 of file animscript.cpp.

References _colors, ColorMap::clear(), process, and readProcess().

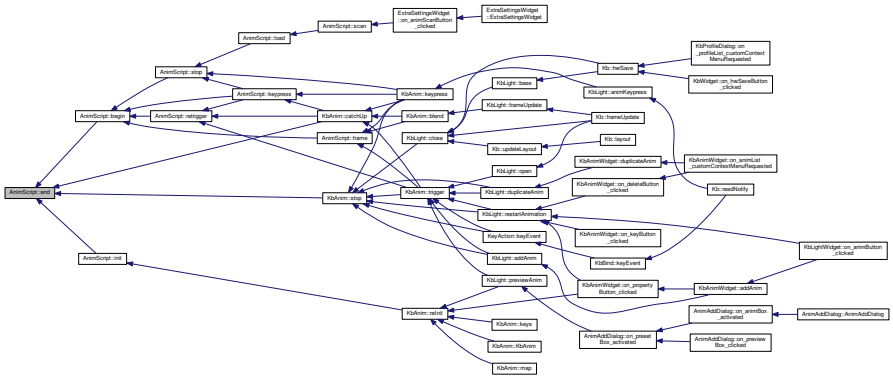
Referenced by begin(), KbAnim::catchUp(), init(), and KbAnim::stop().

```
387                     {
388     _colors.clear();
389     if(process){
390         process->kill();
391         connect(process, SIGNAL(finished(int)), process, SLOT(deleteLater()));
392         disconnect(process, SIGNAL(readyRead()), this, SLOT(readProcess()));
393         process = 0;
394     }
395 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.3.5.10 void AnimScript::frame (quint64 *timestamp*)

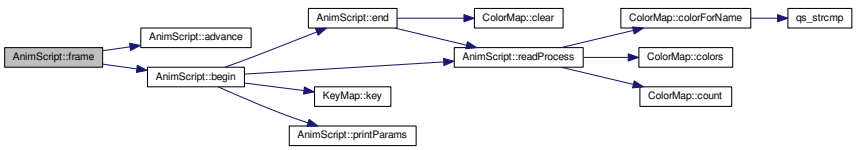
Definition at line 430 of file animscript.cpp.

References advance(), begin(), firstFrame, initialized, process, readFrame, and stopped.

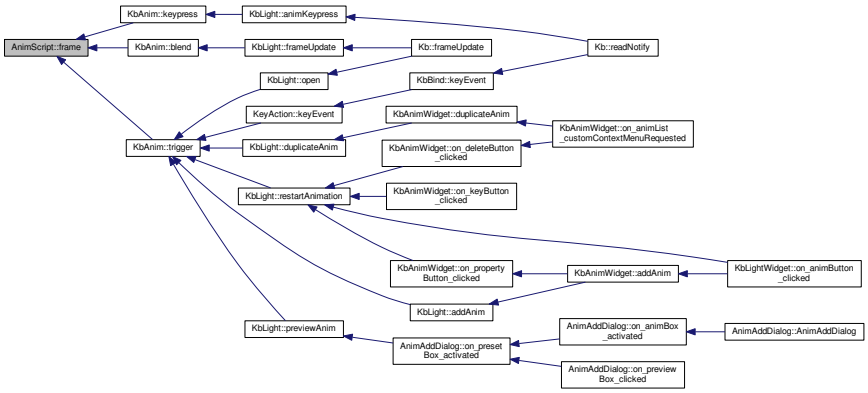
Referenced by KbAnim::blend(), KbAnim::keypress(), and KbAnim::trigger().

```
430                                     {
431     if(!initialized || stopped)
432         return;
433     // Start the animation if it's not running yet
434     if(!process)
435         begin(timestamp);
436
437     advance(timestamp);
438     if((readFrame || !firstFrame) && process)
439         // Don't ask for a new frame if the animation hasn't delivered the last one yet
440         process->write("frame\n");
441     firstFrame = true;
442     readFrame = false;
443 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.3.5.11 const QUuid& AnimScript::guid ( ) const [inline]

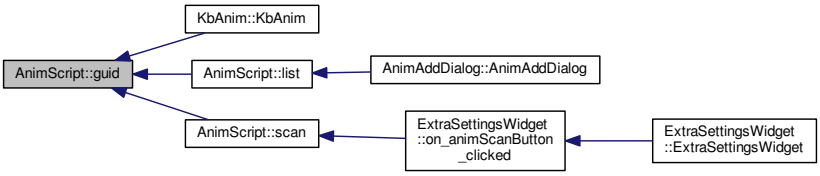
Definition at line 59 of file animscript.h.

References _info.

Referenced by KbAnim::KbAnim(), list(), and scan().

```
59 { return _info.guid; }
```

Here is the caller graph for this function:



12.3.5.12 bool AnimScript::hasFrame ( ) const [inline]

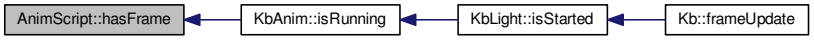
Definition at line 96 of file animscript.h.

References initialized, and readAnyFrame.

Referenced by KbAnim::isRunning().

```
96 { return initialized && readAnyFrame; }
```

Here is the caller graph for this function:



12.3.5.13 bool AnimScript::hasKeypress ( ) const [inline]

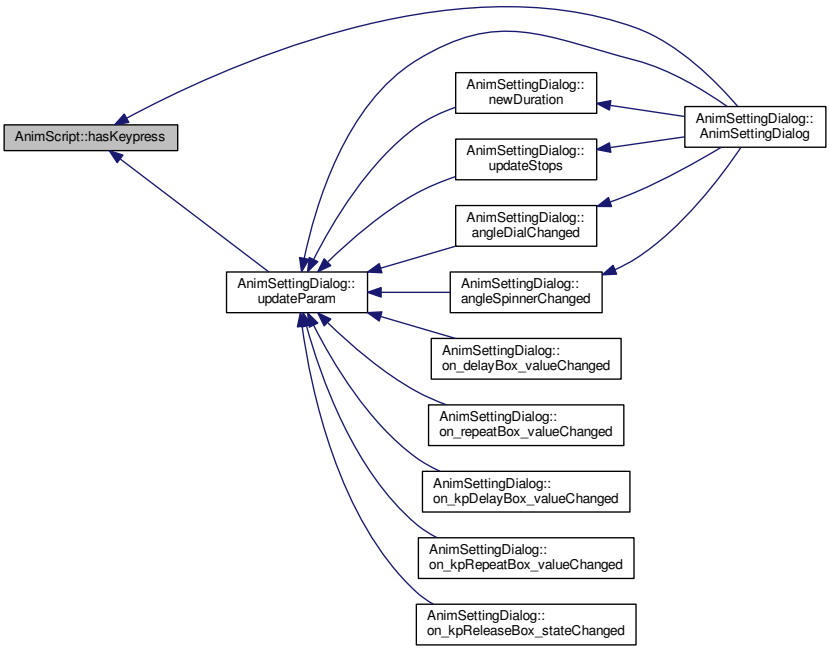
Definition at line 67 of file animscript.h.

References _info, and KP_NONE.

Referenced by AnimSettingDialog::AnimSettingDialog(), and AnimSettingDialog::updateParam().

```
67 { return _info.kpMode != KP_NONE; }
```

Here is the caller graph for this function:



12.3.5.14 bool AnimScript::hasParam (const QString & *name*) const [inline]

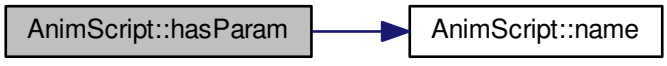
Definition at line 74 of file animscript.h.

References _info, and name().

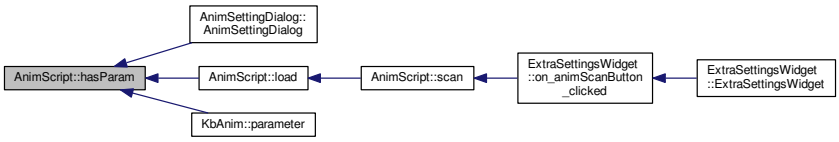
Referenced by AnimSettingDialog::AnimSettingDialog(), load(), and KbAnim::parameter().

```
74 { QListIterator<Param> i(_info.params); while(i.hasNext()){ if(i.next().name ==
     name) return true; } return false; }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.3.5.15 void AnimScript::init (const KeyMap & *map,* const QStringList & *keys,* const QMap< QString, QVariant > & *paramValues*)

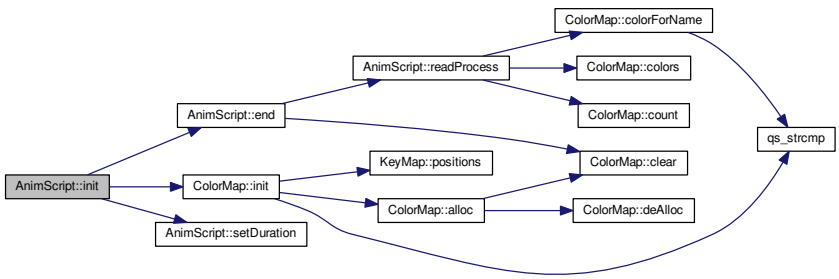
Definition at line 239 of file animscript.cpp.

References _colorBuffer, _colors, _keys, _map, _paramValues, _path, end(), firstFrame, ColorMap::init(), initialized, setDuration(), and stopped.

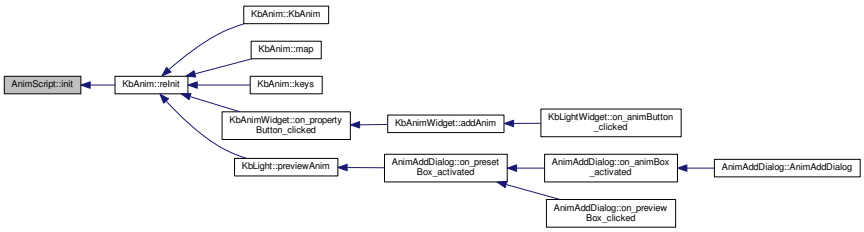
Referenced by KbAnim::reInit().

```
239
    {
240     if(_path == "")
241         return;
242     end();
243     _map = map;
244     _colors.init(map);
245     _colorBuffer.init(map);
246     _keys = keys;
247     _paramValues = paramValues;
248     setDuration();
249     stopped = firstFrame = false;
250     initialized = true;
251 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.3.5.16 void AnimScript::keypress (const QString & *key,* bool *pressed,* quint64 *timestamp*)

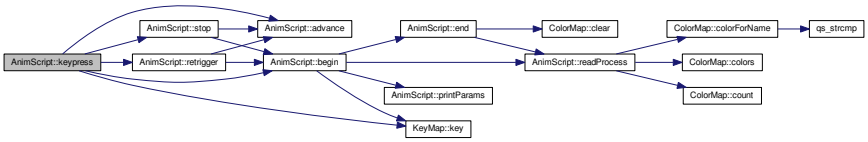
Definition at line 354 of file animscript.cpp.

References _info, _map, _paramValues, advance(), begin(), initialized, KeyMap::key(), KP_NAME, KP_NONE, KP-_POSITION, minX, minY, process, retrigger(), stop(), Key::x, and Key::y.

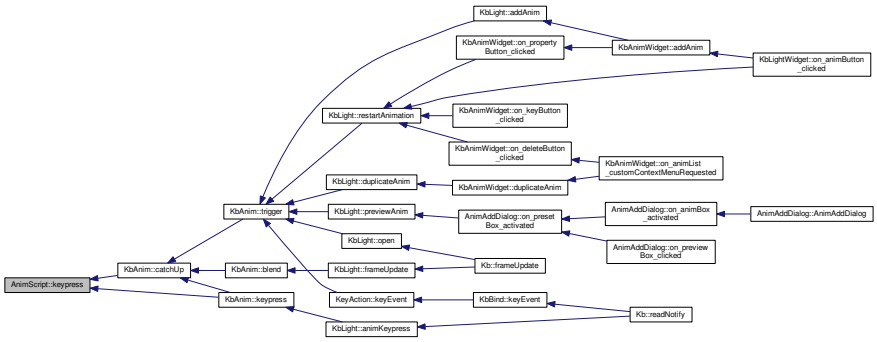
Referenced by KbAnim::catchUp(), and KbAnim::keypress().

```
354                                                                    {
355     if(!initialized)
356         return;
357     if(!process)
358         begin(timestamp);
359     int kpMode = _info.kpMode;
360     if(_paramValues.value("kpmode", 0).toInt() != 0)
361         // Disable KP mode according to user preferences
362         kpMode = KP_NONE;
363     switch(kpMode){
364     case KP_NONE:
365         // If KPs aren't allowed, call retrigger/stop instead
366         if(pressed)
367             retrigger(timestamp);
368         else if(_paramValues.value("kprelease", false).toBool())
369             stop(timestamp);
370         break;
371     case KP_NAME:
372         // Print keypress by name
373         advance(timestamp);
374         process->write(("key " + key + (pressed ? " down\n" : " up\n")).toLatin1());
375         break;
376     case KP_POSITION:
377         // Print keypress by position
378         const Key& kp = _map.key(key);
379         if(!kp)
380             return;
381         advance(timestamp);
382         process->write(("key " + QString("%1,%2").arg(kp.x - minX).arg(kp.
      y - minY) + (pressed ? " down\n" : " up\n")).toLatin1());
383         break;
384     }
385 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.3.5.17 const QString& AnimScript::license ( ) const [inline]

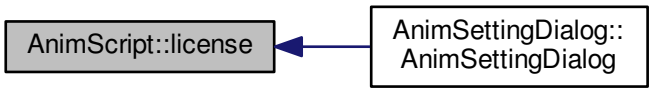
Definition at line 65 of file animscript.h.

References _info.

Referenced by AnimSettingDialog::AnimSettingDialog().

```
65 { return _info.license; }
```

Here is the caller graph for this function:



12.3.5.18 QList< const AnimScript * > AnimScript::list ( ) [static]

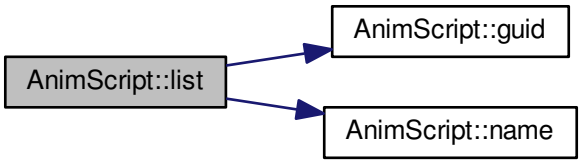
Definition at line 51 of file animscript.cpp.

References _info, guid(), name(), and scripts.

Referenced by AnimAddDialog::AnimAddDialog().

```
51                                      {
52     // Gather the animations into an alphabetically-sorted list
53     QMap<QString, const AnimScript*> result;
54     foreach(AnimScript* script, scripts.values()){
55         QString name = script->name();
56         if(result.contains(name)){
57             // If duplicate names exist, make them unique by including their GUIDs
58             AnimScript* last = (AnimScript*)result[name];
59             last->_info.name += " " + last->guid().toString().toUpper();
60             script->_info.name += " " + script->guid().toString().toUpper();
61         }
62         result[script->name()] = script;
63     }
64     return result.values();
65 }
```

Here is the call graph for this function:



Here is the caller graph for this function:

12.3.5.19 bool AnimScript::load ( ) [private]

Definition at line 77 of file animscript.cpp.

References _info, _path, _presets, _presetValues, AnimScript::Param::AGRADIENT, AnimScript::Param::ANGLE, AnimScript::Param::ARGB, AnimScript::Param::BOOL, count(), AnimScript::Param::defaultValue, AnimScript::Param::DOUBLE, AnimScript::Param::GRADIENT, hasParam(), AnimScript::Param::INVALID, KP_NAME, KP_NONE, KP_POSITION, AnimScript::Param::LABEL, AnimScript::Param::LONG, name(), ONE_DAY, param(), preset(), AnimScript::Param::RGB, stop(), AnimScript::Param::STRING, and urlParam().

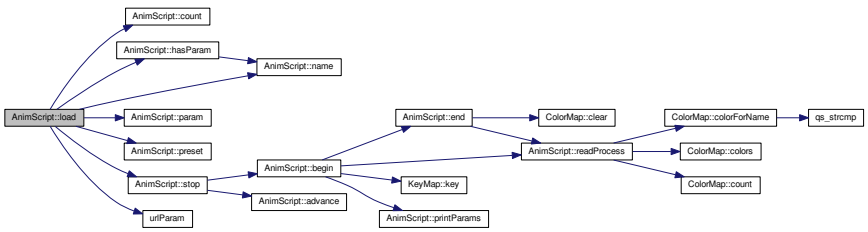
Referenced by scan().

```
77                      {
78     // Run the process to get script info
79     QProcess infoProcess;
80     infoProcess.start(_path, QStringList("--ckb-info"));
81     qDebug() << "Scanning " << _path;
82     infoProcess.waitForFinished(1000);
83     if(infoProcess.state() == QProcess::Running){
84         // Kill the process if it takes more than 1s
85         infoProcess.kill();
86         return false;
87     }
88     // Set defaults for performance info
89     _info.kpMode = KP_NONE;
90     _info.absoluteTime = _info.preempt = _info.liveParams = false;
91     _info.repeat = true;
92     // Read output
93     QString line;
94     while((line = infoProcess.readLine()) != ""){
95         line = line.trimmed();
96         QStringList components = line.split(" ");
97         int count = components.count();
98         if(count < 2)
99             continue;
100         QString param = components[0].trimmed();
101         if(param == "guid")
102             _info.guid = QUuid(urlParam(components[1]));
```

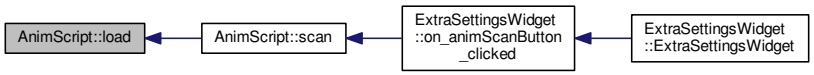
```
103         else if(param == "name")
104             _info.name = urlParam(components[1]);
105         else if(param == "version")
106             _info.version = urlParam(components[1]);
107         else if(param == "year")
108             _info.year = urlParam(components[1]);
109         else if(param == "author")
110             _info.author = urlParam(components[1]);
111         else if(param == "license")
112             _info.license = urlParam(components[1]);
113         else if(param == "description")
114             _info.description = urlParam(components[1]);
115         else if(param == "kpmode")
116             _info.kpMode = (components[1] == "position") ? KP_POSITION : (components[1] ==
    "name") ? KP_NAME : KP_NONE;
117         else if(param == "time")
118             _info.absoluteTime = (components[1] == "absolute");
119         else if(param == "repeat")
120             _info.repeat = (components[1] == "on");
121         else if(param == "preempt")
122             _info.preempt = (components[1] == "on");
123         else if(param == "parammode")
124             _info.liveParams = (components[1] == "live");
125         else if(param == "param"){
126             // Read parameter
127             if(count < 3)
128                 continue;
129             while(components.count() < 8)
130                 components.append("");
131             Param::Type type = Param::INVALID;
132             QString sType = components[1].toLower();
133             if(sType == "long")
134                 type = Param::LONG;
135             else if(sType == "double")
136                 type = Param::DOUBLE;
137             else if(sType == "bool")
138                 type = Param::BOOL;
139             else if(sType == "rgb")
140                 type = Param::RGB;
141             else if(sType == "argb")
142                 type = Param::ARGB;
143             else if(sType == "gradient")
144                 type = Param::GRADIENT;
145             else if(sType == "agradient")
146                 type = Param::AGRADIENT;
147             else if(sType == "angle")
148                 type = Param::ANGLE;
149             else if(sType == "string")
150                 type = Param::STRING;
151             else if(sType == "label")
152                 type = Param::LABEL;
153             else
154                 continue;
155             // "param <type> <name> <prefix> <postfix> <default>"
156             QString name = components[2].toLower();
157             // Make sure it's not present already
158             if(hasParam(name))
159                 continue;
160             QString prefix = urlParam(components[3]), postfix = urlParam(components[4]);
161             QVariant def = urlParam(components[5]), minimum = urlParam(components[6]),
    maximum = urlParam(components[7]);
162             // Don't allow predefined params
163             if(name == "trigger" || name == "kptrigger" || name == "kpmode" || name == "duration" || name
    == "delay" || name == "kpdelay" || name == "repeat" || name == "kprepeat" || name == "stop" || name == "
    kpstop" || name == "kpmodestop" || name == "kprelease")
164                 continue;
165             Param param = { type, name, prefix, postfix, def, minimum, maximum };
166             _info.params.append(param);
167         } else if(param == "preset"){
168             // Add preset
169             QString name = urlParam(components.at(1));
170             QMap<QString, QVariant> preset;
171             for(int i = 2; i < count; i++){
172                 // Scan name/value setting pairs
173                 QString setting = components.at(i);
174                 QStringList sComponents = setting.split("=");
175                 if(sComponents.count() != 2)
176                     continue;
177                 QString param = sComponents.first().trimmed();
178                 QString value = urlParam(sComponents.last());
179                 preset[param] = value;
180             }
181             // If the preset contains a duration, set the repeat time to the same value
182             if(preset.contains("duration")){
183                 QVariant duration = preset.value("duration");
184                 preset["repeat"] = duration;
185                 preset["kprepeat"] = duration;
```

```
186             }
187             _presets.append(name);
188             _presetValues.append(preset);
189         }
190     }
191     // Make sure the required parameters are filled out
192     if(_info.guid.isNull() || _info.name == "" || _info.version == "" ||
    _info.year == "" || _info.author == "" || _info.license == "")
193         return false;
194     // Add timing parameters
195     double defaultDuration = -1.;
196     if(!_info.absoluteTime){
197         defaultDuration = 1.;
198         Param duration = { Param::DOUBLE, "duration", "", "", defaultDuration, 0.1,
    ONE_DAY };
199         _info.params.append(duration);
200     }
201     Param trigger = { Param::BOOL, "trigger", "", "", true, 0, 0 };
202     _info.params.append(trigger);
203     Param kptrigger = { Param::BOOL, "kptrigger", "", "", false, 0, 0 };
204     _info.params.append(kptrigger);
205     if(_info.absoluteTime || !_info.repeat)
206         _info.preempt = false;
207     Param kpmode = { Param::LONG, "kpmode", "", "", 1, 0, 0 };
208     if(_info.kpMode)
209         kpmode.defaultValue = 0;
210     _info.params.append(kpmode);
211     Param delay = { Param::DOUBLE, "delay", "", "", 0., 0., ONE_DAY };
212     Param kpdelay = { Param::DOUBLE, "kpdelay", "", "", 0., 0.,
    ONE_DAY };
213     _info.params.append(delay);
214     _info.params.append(kpdelay);
215     Param kpmodestop = { Param::BOOL, "kpmodestop", "", "", false, 0, 0 };
216     Param kprelease = { Param::BOOL, "kprelease", "", "", false, 0, 0 };
217     _info.params.append(kpmodestop);
218     _info.params.append(kprelease);
219     if(_info.repeat){
220         Param repeat = { Param::DOUBLE, "repeat", "", "", defaultDuration, 0.1,
    ONE_DAY };
221         Param kprepeat = { Param::DOUBLE, "kprepeat", "", "", defaultDuration, 0.1,
    ONE_DAY };
222         // When repeats are enabled, stop and kpstop are LONG values (number of repeats)
223         Param stop = { Param::LONG, "stop", "", "", -1, 0, 1000 };
224         Param kpstop = { Param::LONG, "kpstop", "", "", 0, 0, 1000 };
225         _info.params.append(repeat);
226         _info.params.append(kprepeat);
227         _info.params.append(stop);
228         _info.params.append(kpstop);
229     } else {
230         // When repeats are disabled, stop and kpstop are DOUBLE values (seconds)
231         Param stop = { Param::DOUBLE, "stop", "", "", -1., 0.1,
    ONE_DAY };
232         Param kpstop = { Param::DOUBLE, "kpstop", "", "", -1., 0.1,
    ONE_DAY };
233         _info.params.append(stop);
234         _info.params.append(kpstop);
235     }
236     return true;
237 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.3.5.20 const QString& AnimScript::name ( ) const [inline]

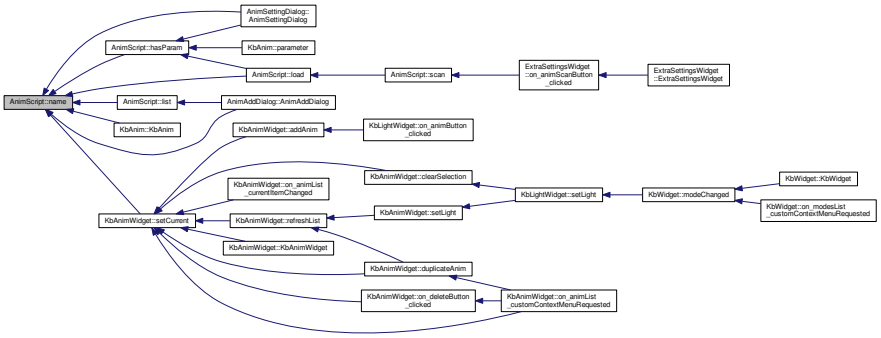
Definition at line 60 of file animscript.h.

References _info.

Referenced by AnimAddDialog::AnimAddDialog(), AnimSettingDialog::AnimSettingDialog(), hasParam(), KbAnim::-KbAnim(), list(), load(), and KbAnimWidget::setCurrent().

```
60 { return _info.name; }
```

Here is the caller graph for this function:



12.3.5.21 Param AnimScript::param (const QString & *name*) const [inline]

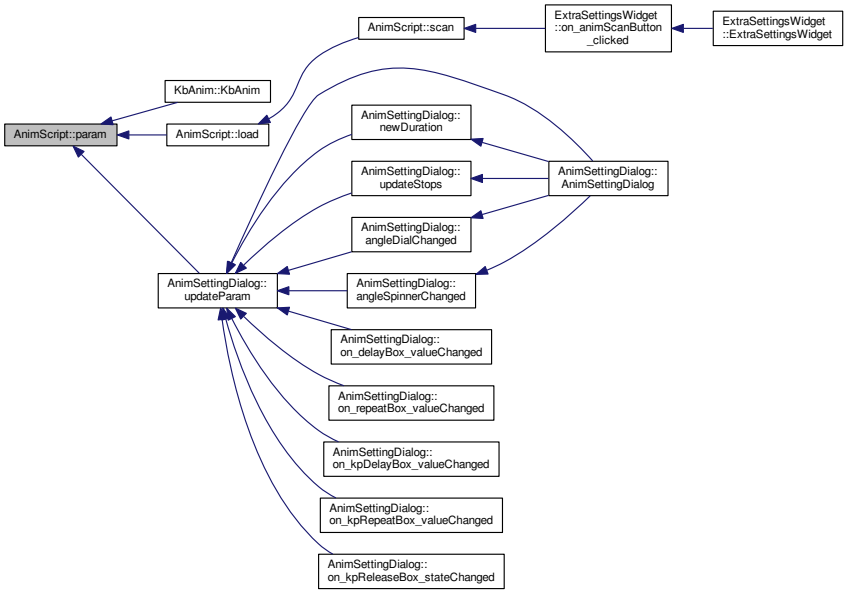
Definition at line 73 of file animscript.h.

References _info, AnimScript::Param::INVALID, and AnimScript::Param::name.

Referenced by KbAnim::KbAnim(), load(), and AnimSettingDialog::updateParam().

```
73 { QListIterator<Param> i(_info.params); while(i.hasNext()){ Param p = i.next(); if(p.name ==
     name) return p; } return ((Param[]){ { Param::INVALID, "", "", "", 0, 0, 0 } })[0]; }
```

Here is the caller graph for this function:



12.3.5.22 void AnimScript::parameters (const QMap< QString, QVariant > & *paramValues*)

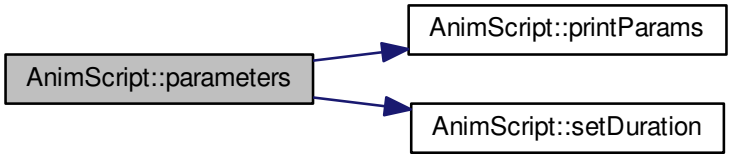
Definition at line 265 of file animscript.cpp.

References _info, _paramValues, initialized, printParams(), process, and setDuration().

Referenced by KbAnim::updateParams().

```
265                                                                    {
266     if(!initialized || !process || !_info.liveParams)
267         return;
268     _paramValues = paramValues;
269     setDuration();
270     printParams();
271 }
```

Here is the call graph for this function:



Here is the caller graph for this function:

12.3.5.23 QListIterator<Param> AnimScript::paramIterator ( ) const [inline]

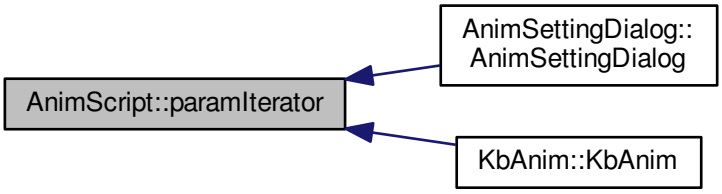
Definition at line 72 of file animscript.h.

References _info.

Referenced by AnimSettingDialog::AnimSettingDialog(), and KbAnim::KbAnim().

```
72 { return _info.params; }
```

Here is the caller graph for this function:



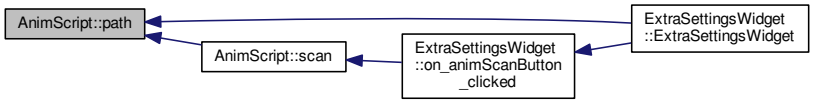
12.3.5.24 QString AnimScript::path ( ) [static]

Definition at line 29 of file animscript.cpp.

Referenced by ExtraSettingsWidget::ExtraSettingsWidget(), and scan().

```
29                       {
30 #ifdef __APPLE__
31     return QDir(QApplication::applicationDirPath() + "/../Resources").absoluteFilePath("ckb-animations");
32 #else
33     return QDir("/usr/lib").absoluteFilePath("ckb-animations");
34 #endif
35 }
```

Here is the caller graph for this function:



12.3.5.25 const PresetValue& AnimScript::preset (int *index*) const [inline]

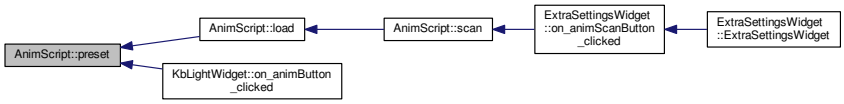
Definition at line 69 of file animscript.h.

References _presetValues.

Referenced by load(), and KbLightWidget::on_animButton_clicked().

```
69 { return _presetValues[index]; }
```

Here is the caller graph for this function:



12.3.5.26 const QStringList& AnimScript::presets ( ) const [inline]

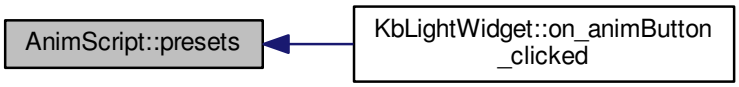
Definition at line 68 of file animscript.h.

References _presets.

Referenced by KbLightWidget::on_animButton_clicked().

```
68 { return _presets; }
```

Here is the caller graph for this function:



12.3.5.27 void AnimScript::printParams ( ) [private]

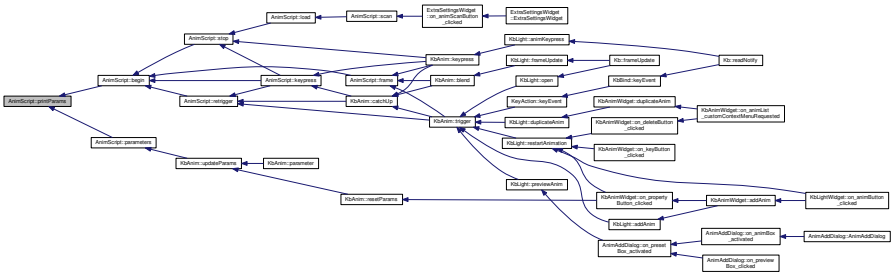
Definition at line 273 of file animscript.cpp.

References _paramValues, and process.

Referenced by begin(), and parameters().

```
273                             {
274     process->write("begin params\n");
275     QMapIterator<QString, QVariant> i(_paramValues);
276     while(i.hasNext()){
277         i.next();
278         process->write("param ");
279         process->write(i.key().toLatin1());
280         process->write(" ");
281         process->write(QUrl::toPercentEncoding(i.value().toString()));
282         process->write("\n");
283     }
284     process->write("end params\n");
285 }
```

Here is the caller graph for this function:



12.3.5.28 void AnimScript::readProcess ( ) [private], [slot]

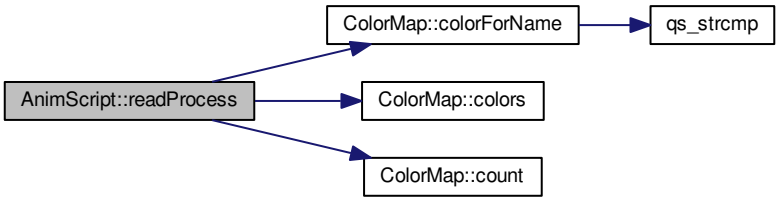
Definition at line 397 of file animscript.cpp.

References _colorBuffer, _colors, ColorMap::colorForName(), ColorMap::colors(), ColorMap::count(), inFrame, process, readAnyFrame, readFrame, and stopped.

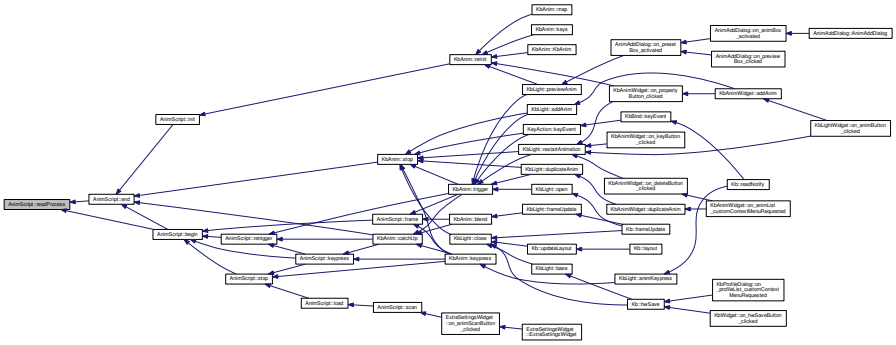
Referenced by begin(), and end().

```
397                             {
398     while(process->canReadLine()){
399         QByteArray line = process->readLine().trimmed();
400         if(!inFrame){
401             // Ignore anything not between "begin frame" and "end frame", except for "end run", which
      indicates that the program is done.
402             if(line == "begin frame")
403                 inFrame = true;
404             else if(line == "end run"){
405                 stopped = true;
406                 return;
407             }
408             continue;
409         }
410         if(line.startsWith("argb ")){
411             // Add a color to the buffer
412             char keyName[31];
413             QRgb keyColor = 0;
414             if(sscanf(line, "argb %30s %x", keyName, &keyColor) != 2)
415                 continue;
416             QRgb* inMap = _colorBuffer.colorForName(keyName);
417             if(!inMap)
418                 continue;
419             *inMap = keyColor;
420         }
421         if(line == "end frame"){
422             // Frame is finished. Copy color buffer back to the atomic map
423             memcpy(_colors.colors(), _colorBuffer.
      colors(), sizeof(QRgb) * _colors.count());
424             inFrame = false;
425             readFrame = readAnyFrame = true;
426         }
427     }
428 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.3.5.29 void AnimScript::retrigger (quint64 *timestamp,* bool *allowPreempt* = `false`)

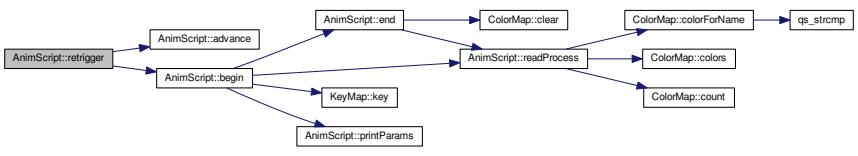
Definition at line 331 of file animscript.cpp.

References _info, advance(), begin(), initialized, process, and repeatMsec.

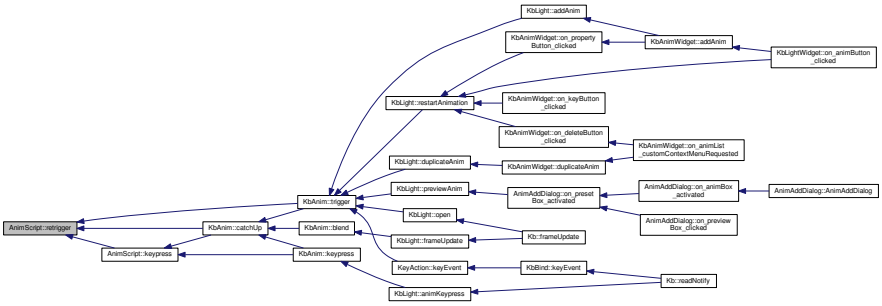
Referenced by KbAnim::catchUp(), keypress(), and KbAnim::trigger().

```
331                                                         {
332     if(!initialized)
333         return;
334     if(allowPreempt && _info.preempt && repeatMsec > 0)
335         // If preemption is wanted, trigger the animation 1 duration in the past first
336         retrigger(timestamp - repeatMsec);
337     if(!process)
338         begin(timestamp);
339     advance(timestamp);
340     if(process)
341         process->write("start\n");
342 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.3.5.30 void AnimScript::scan ( ) [static]

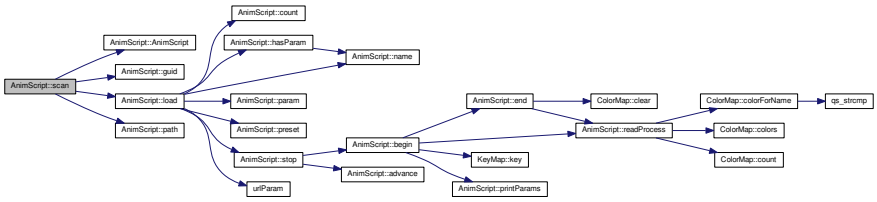
Definition at line 37 of file animscript.cpp.

References _info, AnimScript(), guid(), load(), path(), and scripts.

Referenced by ExtraSettingsWidget::on_animScanButton_clicked().

```
37                     {
38     QDir dir(path());
39     foreach(AnimScript* script, scripts)
40         delete script;
41     scripts.clear();
42     foreach(QString file, dir.entryList(QDir::Files | QDir::Executable)){
43         AnimScript* script = new AnimScript(qApp, dir.absoluteFilePath(file));
44         if(script->load() && !scripts.contains(script->_info.
       guid))
45             scripts[script->_info.guid] = script;
46         else
47             delete script;
48     }
49 }
```

Here is the call graph for this function:



Here is the caller graph for this function:

12.3.5.31 void AnimScript::setDuration ( ) [private]

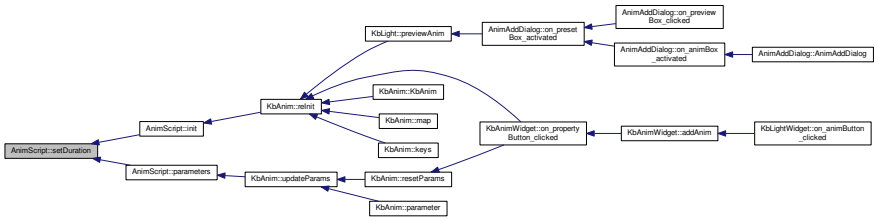
Definition at line 253 of file animscript.cpp.

References _info, _paramValues, durationMsec, and repeatMsec.

Referenced by init(), and parameters().

```
253                             {
254     if(_info.absoluteTime){
255         durationMsec = 1000;
256         repeatMsec = 0;
257     } else {
258         durationMsec = round(_paramValues.value("duration").toDouble() * 1000.);
259         if(durationMsec <= 0)
260             durationMsec = -1;
261         repeatMsec = round(_paramValues.value("repeat").toDouble() * 1000.);
262     }
263 }
```

Here is the caller graph for this function:



12.3.5.32 void AnimScript::stop (quint64 *timestamp*)

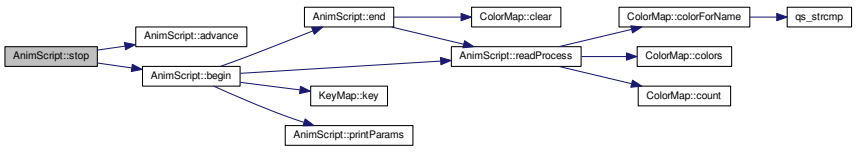
Definition at line 344 of file animscript.cpp.

References advance(), begin(), initialized, and process.

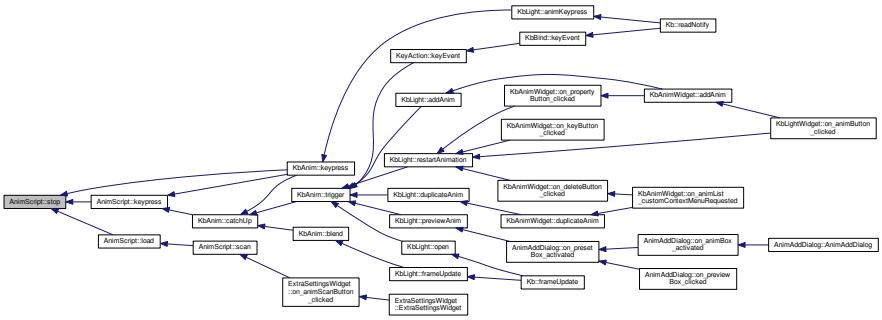
Referenced by KbAnim::keypress(), keypress(), and load().

```
344                                     {
345     if(!initialized)
346         return;
347     if(!process)
348         begin(timestamp);
349     advance(timestamp);
350     if(process)
351         process->write("stop\n");
352 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.3.5.33 const QString& AnimScript::version ( ) const [inline]

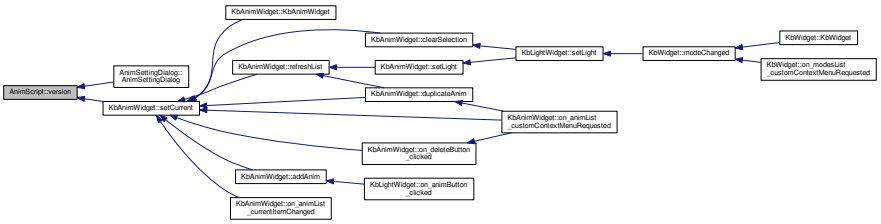
Definition at line 61 of file animscript.h.

References _info.

Referenced by AnimSettingDialog::AnimSettingDialog(), and KbAnimWidget::setCurrent().

```
61 { return _info.version; }
```

Here is the caller graph for this function:



12.3.5.34 const QString& AnimScript::year ( ) const [inline]

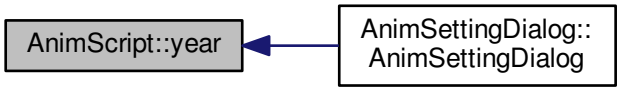
Definition at line 63 of file animscript.h.

References _info.

Referenced by AnimSettingDialog::AnimSettingDialog().

```
63 { return _info.year; }
```

Here is the caller graph for this function:



12.3.6 Field Documentation

12.3.6.1 ColorMap AnimScript::_colorBuffer [private]

Definition at line 143 of file animscript.h.

Referenced by init(), and readProcess().

12.3.6.2 ColorMap AnimScript::_colors [private]

Definition at line 135 of file animscript.h.

Referenced by colors(), end(), init(), and readProcess().

12.3.6.3 struct { ... } AnimScript::_info [private]

Referenced by advance(), author(), copyright(), description(), guid(), hasKeypress(), hasParam(), keypress(), license(), list(), load(), name(), param(), parameters(), paramIterator(), retrigger(), scan(), setDuration(), version(), and year().

12.3.6.4 QStringList AnimScript::_keys [private]

Definition at line 133 of file animscript.h.

Referenced by begin(), and init().

12.3.6.5 KeyMap AnimScript::_map [private]

Definition at line 130 of file animscript.h.

Referenced by begin(), init(), and keypress().

12.3.6.6 PresetValue AnimScript::_paramValues [private]

Definition at line 136 of file animscript.h.

Referenced by init(), keypress(), parameters(), printParams(), and setDuration().

12.3.6.7 QString AnimScript::_path [private]

Definition at line 128 of file animscript.h.

Referenced by begin(), init(), and load().

12.3.6.8 QStringList AnimScript::_presets [private]

Definition at line 125 of file animscript.h.

Referenced by load(), and presets().

12.3.6.9 QList<PresetValue> AnimScript::_presetValues [private]

Definition at line 126 of file animscript.h.

Referenced by load(), and preset().

12.3.6.10 int AnimScript::durationMsec [private]

Definition at line 140 of file animscript.h.

Referenced by advance(), and setDuration().

12.3.6.11 bool AnimScript::firstFrame [private]

Definition at line 141 of file animscript.h.

Referenced by begin(), frame(), and init().

12.3.6.12 bool AnimScript::inFrame [private]

Definition at line 141 of file animscript.h.

Referenced by begin(), and readProcess().

12.3.6.13 bool AnimScript::initialized [private]

Definition at line 141 of file animscript.h.

Referenced by begin(), frame(), hasFrame(), init(), keypress(), parameters(), retrigger(), and stop().

12.3.6.14 const int AnimScript::KP_NAME = 1 [static], [private]

Definition at line 124 of file animscript.h.

Referenced by keypress(), and load().

12.3.6.15 const int AnimScript::KP_NONE = 0 [static], [private]

Definition at line 124 of file animscript.h.

Referenced by hasKeypress(), keypress(), and load().

12.3.6.16 const int AnimScript::KP_POSITION = 2 [static], [private]

Definition at line 124 of file animscript.h.

Referenced by keypress(), and load().

12.3.6.17 quint64 AnimScript::lastFrame [private]

Definition at line 139 of file animscript.h.

Referenced by advance(), and begin().

12.3.6.18 int AnimScript::minX [private]

Definition at line 131 of file animscript.h.

Referenced by begin(), and keypress().

12.3.6.19 int AnimScript::minY [private]

Definition at line 131 of file animscript.h.

Referenced by begin(), and keypress().

12.3.6.20 QProcess* AnimScript::process [private]

Definition at line 142 of file animscript.h.

Referenced by advance(), begin(), end(), frame(), keypress(), parameters(), printParams(), readProcess(), retrigger(), stop(), and ~AnimScript().

12.3.6.21 bool AnimScript::readAnyFrame [private]

Definition at line 141 of file animscript.h.

Referenced by begin(), hasFrame(), and readProcess().

12.3.6.22 bool AnimScript::readFrame [private]

Definition at line 141 of file animscript.h.

Referenced by begin(), frame(), and readProcess().

12.3.6.23 int AnimScript::repeatMsec [private]

Definition at line 140 of file animscript.h.

Referenced by retrigger(), and setDuration().

12.3.6.24 QHash< QUuid, AnimScript * > AnimScript::scripts [static], [private]

Definition at line 152 of file animscript.h.

Referenced by copy(), count(), list(), and scan().

12.3.6.25 bool AnimScript::stopped [private]

Definition at line 141 of file animscript.h.

Referenced by begin(), frame(), init(), and readProcess().

The documentation for this class was generated from the following files:

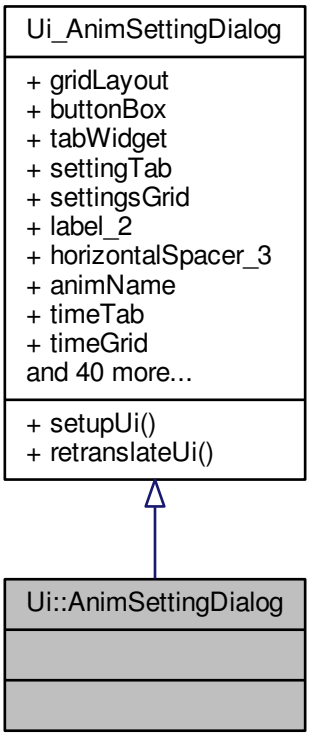
- src/ckb/animscript.h

- src/ckb/animscript.cpp

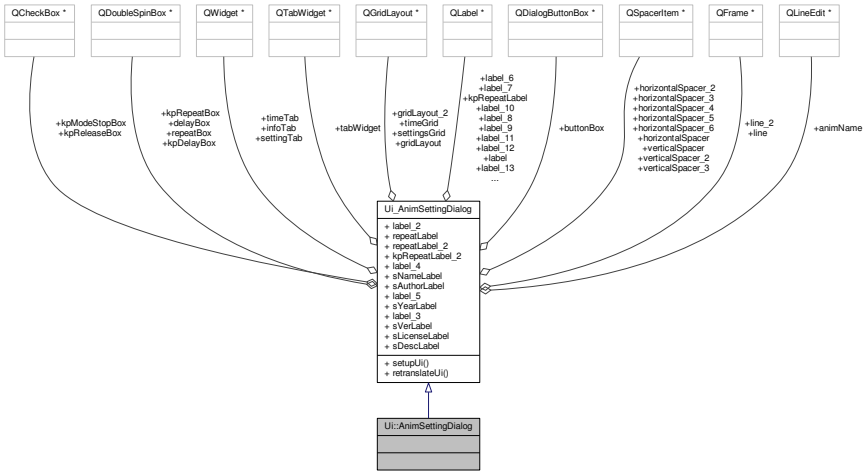
12.4 Ui::AnimSettingDialog Class Reference

`#include <src/ckb/ui_animsettingdialog.h>`

Inheritance diagram for Ui::AnimSettingDialog:



Collaboration diagram for Ui::AnimSettingDialog:



Public Member Functions

- void setupUi (QDialog *AnimSettingDialog)
- void retranslateUi (QDialog *AnimSettingDialog)

Data Fields

- QGridLayout * gridLayout
- QDialogButtonBox * buttonBox
- QTabWidget * tabWidget
- QWidget * settingTab
- QGridLayout * settingsGrid
- QLabel * label_2
- QSpacerItem * horizontalSpacer_3
- QLineEdit * animName
- QWidget * timeTab
- QGridLayout * timeGrid
- QSpacerItem * horizontalSpacer_5
- QLabel * kpRepeatLabel
- QLabel * repeatLabel
- QDoubleSpinBox * repeatBox
- QLabel * repeatLabel_2
- QFrame * line
- QLabel * kpRepeatLabel_2
- QLabel * label_4
- QLabel * label_8
- QDoubleSpinBox * kpDelayBox
- QLabel * label_13
- QLabel * label_12
- QDoubleSpinBox * delayBox
- QCheckBox * kpModeStopBox
- QLabel * label_10
- QSpacerItem * verticalSpacer_3
- QSpacerItem * horizontalSpacer_4

- QCheckBox * kpReleaseBox
- QLabel * label_6
- QDoubleSpinBox * kpRepeatBox
- QSpacerItem * verticalSpacer_2
- QFrame * line_2
- QSpacerItem * horizontalSpacer_6
- QWidget * infoTab
- QGridLayout * gridLayout_2
- QLabel * sNameLabel
- QLabel * label_7
- QLabel * label_11
- QLabel * sAuthorLabel
- QLabel * label_9
- QLabel * label_5
- QLabel * sYearLabel
- QLabel * label_3
- QSpacerItem * horizontalSpacer
- QSpacerItem * horizontalSpacer_2
- QLabel * sVerLabel
- QLabel * sLicenseLabel
- QLabel * label
- QSpacerItem * verticalSpacer
- QLabel * sDescLabel

12.4.1 Detailed Description

Definition at line 392 of file ui_animsettingdialog.h.

12.4.2 Member Function Documentation

12.4.2.1 void Ui_AnimSettingDialog::retranslateUi (QDialog * *AnimSettingDialog*) [inline], [inherited]

Definition at line 356 of file ui_animsettingdialog.h.

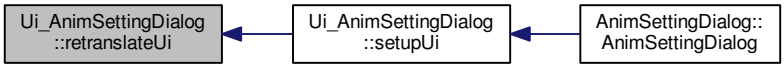
References Ui_AnimSettingDialog::infoTab, Ui_AnimSettingDialog::kpModeStopBox, Ui_AnimSettingDialog::kpReleaseBox, Ui_AnimSettingDialog::kpRepeatLabel, Ui_AnimSettingDialog::kpRepeatLabel_2, Ui_AnimSettingDialog::label, Ui_AnimSettingDialog::label_10, Ui_AnimSettingDialog::label_11, Ui_AnimSettingDialog::label_12, Ui_AnimSettingDialog::label_13, Ui_AnimSettingDialog::label_2, Ui_AnimSettingDialog::label_3, Ui_AnimSettingDialog::label_4, Ui_AnimSettingDialog::label_5, Ui_AnimSettingDialog::label_6, Ui_AnimSettingDialog::label_7, Ui_AnimSettingDialog::label_8, Ui_AnimSettingDialog::label_9, Ui_AnimSettingDialog::repeatLabel, Ui_AnimSettingDialog::repeatLabel_2, Ui_AnimSettingDialog::sAuthorLabel, Ui_AnimSettingDialog::sDescLabel, Ui_AnimSettingDialog::settingTab, Ui_AnimSettingDialog::sLicenseLabel, Ui_AnimSettingDialog::sNameLabel, Ui_AnimSettingDialog::sVerLabel, Ui_AnimSettingDialog::sYearLabel, Ui_AnimSettingDialog::tabWidget, and Ui_AnimSettingDialog::timeTab.

Referenced by Ui_AnimSettingDialog::setupUi().

```
357     {
358         AnimSettingDialog->setWindowTitle(QApplication::translate("AnimSettingDialog", "Dialog", Q_NULLPTR)
    );
359         label_2->setText(QApplication::translate("AnimSettingDialog", "Name:", Q_NULLPTR));
360         tabWidget->setTabText(tabWidget->indexOf(settingTab),
    QApplication::translate("AnimSettingDialog", "Settings", Q_NULLPTR));
361         kpRepeatLabel->setText(QApplication::translate("AnimSettingDialog", "Repeat every:",
    Q_NULLPTR));
362         repeatLabel->setText(QApplication::translate("AnimSettingDialog", "Repeat every:",
    Q_NULLPTR));
363         repeatLabel_2->setText(QApplication::translate("AnimSettingDialog", "seconds",
    Q_NULLPTR));
364         kpRepeatLabel_2->setText(QApplication::translate("AnimSettingDialog", "seconds",
```

```
    Q_NULLPTR));
365         label_4->setText(QApplication::translate("AnimSettingDialog", "<b>On mode start</b>",
    Q_NULLPTR));
366         label_8->setText(QApplication::translate("AnimSettingDialog", "seconds", Q_NULLPTR));
367         label_13->setText(QApplication::translate("AnimSettingDialog", "seconds", Q_NULLPTR));
368         label_12->setText(QApplication::translate("AnimSettingDialog", "Start delay:", Q_NULLPTR));
369         kpModeStopBox->setText(QApplication::translate("AnimSettingDialog", "Stop on key press
    ", Q_NULLPTR));
370         label_10->setText(QApplication::translate("AnimSettingDialog", "Start delay:", Q_NULLPTR));
371         kpReleaseBox->setText(QApplication::translate("AnimSettingDialog", "Stop on key release
    ", Q_NULLPTR));
372         label_6->setText(QApplication::translate("AnimSettingDialog", "<b>On key press</b>",
    Q_NULLPTR));
373         tabWidget->setTabText(tabWidget->indexOf(timeTab), QApplication::translate
    ("AnimSettingDialog", "Timing", Q_NULLPTR));
374         sNameLabel->setText(QApplication::translate("AnimSettingDialog", "Name", Q_NULLPTR));
375         label_7->setText(QApplication::translate("AnimSettingDialog", "Year:", Q_NULLPTR));
376         label_11->setText(QApplication::translate("AnimSettingDialog", "Description:", Q_NULLPTR));
377         sAuthorLabel->setText(QApplication::translate("AnimSettingDialog", "Author", Q_NULLPTR)
    );
378         label_9->setText(QApplication::translate("AnimSettingDialog", "License:", Q_NULLPTR));
379         label_5->setText(QApplication::translate("AnimSettingDialog", "Version:", Q_NULLPTR));
380         sYearLabel->setText(QApplication::translate("AnimSettingDialog", "Year", Q_NULLPTR));
381         label_3->setText(QApplication::translate("AnimSettingDialog", "Author:", Q_NULLPTR));
382         sVerLabel->setText(QApplication::translate("AnimSettingDialog", "Version", Q_NULLPTR));
383         sLicenseLabel->setText(QApplication::translate("AnimSettingDialog", "License",
    Q_NULLPTR));
384         label->setText(QApplication::translate("AnimSettingDialog", "Script name:", Q_NULLPTR));
385         sDescLabel->setText(QApplication::translate("AnimSettingDialog", "Description", Q_NULLPTR
    ));
386         tabWidget->setTabText(tabWidget->indexOf(infoTab), QApplication::translate
    ("AnimSettingDialog", "About", Q_NULLPTR));
387     } // retranslateUi
```

Here is the caller graph for this function:



12.4.2.2 void Ui_AnimSettingDialog::setupUi (QDialog * *AnimSettingDialog*) [inline], [inherited]

Definition at line 85 of file ui_animsettingdialog.h.

References Ui_AnimSettingDialog::animName, Ui_AnimSettingDialog::buttonBox, Ui_AnimSettingDialog::delayBox, Ui_AnimSettingDialog::gridLayout, Ui_AnimSettingDialog::gridLayout_2, Ui_AnimSettingDialog::horizontalSpacer, Ui_AnimSettingDialog::horizontalSpacer_2, Ui_AnimSettingDialog::horizontalSpacer_3, Ui_AnimSettingDialog::horizontalSpacer_4, Ui_AnimSettingDialog::horizontalSpacer_5, Ui_AnimSettingDialog::horizontalSpacer_6, Ui_AnimSettingDialog::infoTab, Ui_AnimSettingDialog::kpDelayBox, Ui_AnimSettingDialog::kpModeStopBox, Ui_AnimSettingDialog::kpReleaseBox, Ui_AnimSettingDialog::kpRepeatBox, Ui_AnimSettingDialog::kpRepeatLabel, Ui_AnimSettingDialog::kpRepeatLabel_2, Ui_AnimSettingDialog::label, Ui_AnimSettingDialog::label_10, Ui_AnimSettingDialog::label_11, Ui_AnimSettingDialog::label_12, Ui_AnimSettingDialog::label_13, Ui_AnimSettingDialog::label_2, Ui_AnimSettingDialog::label_3, Ui_AnimSettingDialog::label_4, Ui_AnimSettingDialog::label_5, Ui_AnimSettingDialog::label_6, Ui_AnimSettingDialog::label_7, Ui_AnimSettingDialog::label_8, Ui_AnimSettingDialog::label_9, Ui_AnimSettingDialog::line, Ui_AnimSettingDialog::line_2, Ui_AnimSettingDialog::repeatBox, Ui_AnimSettingDialog::repeatLabel, Ui_AnimSettingDialog::repeatLabel_2, Ui_AnimSettingDialog::retranslateUi(), Ui_AnimSettingDialog::sAuthorLabel, Ui_AnimSettingDialog::sDescLabel, Ui_AnimSettingDialog::settingsGrid, Ui_AnimSettingDialog::settingTab, Ui_AnimSettingDialog::sLicenseLabel, Ui_AnimSettingDialog::sNameLabel, Ui_AnimSettingDialog::sVerLabel, Ui_AnimSettingDialog::sYearLabel, Ui_AnimSettingDialog::tabWidget, Ui_AnimSettingDialog::timeGrid, Ui_AnimSettingDialog::timeTab, Ui_AnimSettingDialog::verticalSpacer, Ui_AnimSettingDialog::verticalSpacer_2, and Ui_AnimSettingDialog::verticalSpacer_3.

Referenced by AnimSettingDialog::AnimSettingDialog().

```
86      {
87          if (AnimSettingDialog->objectName().isEmpty())
88              AnimSettingDialog->setObjectName(QStringLiteral("AnimSettingDialog"));
89          AnimSettingDialog->resize(480, 457);
90          AnimSettingDialog->setMinimumSize(QSize(480, 0));
91          AnimSettingDialog->setFocusPolicy(Qt::ClickFocus);
92          gridLayout = new QGridLayout(AnimSettingDialog);
93          gridLayout->setObjectName(QStringLiteral("gridLayout"));
94          buttonBox = new QDialogButtonBox(AnimSettingDialog);
95          buttonBox->setObjectName(QStringLiteral("buttonBox"));
96          buttonBox->setOrientation(Qt::Horizontal);
97          buttonBox->setStandardButtons(QDialogButtonBox::Cancel|QDialogButtonBox::Ok);
98
99          gridLayout->addWidget(buttonBox, 2, 0, 1, 1);
100
101          tabWidget = new QTabWidget(AnimSettingDialog);
102          tabWidget->setObjectName(QStringLiteral("tabWidget"));
103          settingTab = new QWidget();
104          settingTab->setObjectName(QStringLiteral("settingTab"));
105          settingsGrid = new QGridLayout(settingTab);
106          settingsGrid->setSpacing(12);
107          settingsGrid->setObjectName(QStringLiteral("settingsGrid"));
108          settingsGrid->setContentsMargins(12, 12, 12, 12);
109          label_2 = new QLabel(settingTab);
110          label_2->setObjectName(QStringLiteral("label_2"));
111
112          settingsGrid->addWidget(label_2, 0, 1, 1, 1);
113
114          horizontalSpacer_3 = new QSpacerItem(5, 0, QSizePolicy::Fixed,
    QSizePolicy::Minimum);
115
116          settingsGrid->addItem(horizontalSpacer_3, 0, 2, 1, 1);
117
118          animName = new QLineEdit(settingTab);
119          animName->setObjectName(QStringLiteral("animName"));
120
121          settingsGrid->addWidget(animName, 0, 3, 1, 3);
122
123          tabWidget->addTab(settingTab, QString());
124          timeTab = new QWidget();
125          timeTab->setObjectName(QStringLiteral("timeTab"));
126          timeGrid = new QGridLayout(timeTab);
127          timeGrid->setSpacing(12);
128          timeGrid->setObjectName(QStringLiteral("timeGrid"));
129          timeGrid->setContentsMargins(12, 12, 12, 12);
130          horizontalSpacer_5 = new QSpacerItem(15, 0, QSizePolicy::Fixed,
    QSizePolicy::Minimum);
131
132          timeGrid->addItem(horizontalSpacer_5, 3, 2, 1, 1);
133
134          kpRepeatLabel = new QLabel(timeTab);
135          kpRepeatLabel->setObjectName(QStringLiteral("kpRepeatLabel"));
136
137          timeGrid->addWidget(kpRepeatLabel, 11, 1, 1, 1);
138
139          repeatLabel = new QLabel(timeTab);
140          repeatLabel->setObjectName(QStringLiteral("repeatLabel"));
141
142          timeGrid->addWidget(repeatLabel, 3, 1, 1, 1);
143
144          repeatBox = new QDoubleSpinBox(timeTab);
145          repeatBox->setObjectName(QStringLiteral("repeatBox"));
146          repeatBox->setDecimals(1);
147          repeatBox->setMinimum(0.1);
148          repeatBox->setMaximum(86400);
149
150          timeGrid->addWidget(repeatBox, 3, 3, 1, 1);
151
152          repeatLabel_2 = new QLabel(timeTab);
153          repeatLabel_2->setObjectName(QStringLiteral("repeatLabel_2"));
154
155          timeGrid->addWidget(repeatLabel_2, 3, 4, 1, 1);
156
157          line = new QFrame(timeTab);
158          line->setObjectName(QStringLiteral("line"));
159          line->setFrameShape(QFrame::HLine);
160          line->setFrameShadow(QFrame::Sunken);
161
162          timeGrid->addWidget(line, 1, 0, 1, 6);
163
164          kpRepeatLabel_2 = new QLabel(timeTab);
165          kpRepeatLabel_2->setObjectName(QStringLiteral("kpRepeatLabel_2"));
166
167          timeGrid->addWidget(kpRepeatLabel_2, 11, 4, 1, 1);
168
169          label_4 = new QLabel(timeTab);
170          label_4->setObjectName(QStringLiteral("label_4"));
```

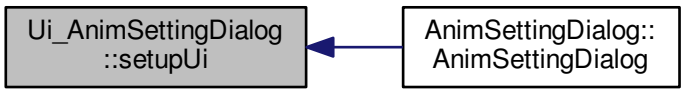
```
171
172         timeGrid->addWidget(label_4, 0, 0, 1, 6);
173
174         label_8 = new QLabel(timeTab);
175         label_8->setObjectName(QStringLiteral("label_8"));
176
177         timeGrid->addWidget(label_8, 2, 4, 1, 1);
178
179         kpDelayBox = new QDoubleSpinBox(timeTab);
180         kpDelayBox->setObjectName(QStringLiteral("kpDelayBox"));
181         kpDelayBox->setDecimals(1);
182         kpDelayBox->setMaximum(86400);
183
184         timeGrid->addWidget(kpDelayBox, 10, 3, 1, 1);
185
186         label_13 = new QLabel(timeTab);
187         label_13->setObjectName(QStringLiteral("label_13"));
188
189         timeGrid->addWidget(label_13, 10, 4, 1, 1);
190
191         label_12 = new QLabel(timeTab);
192         label_12->setObjectName(QStringLiteral("label_12"));
193
194         timeGrid->addWidget(label_12, 10, 1, 1, 1);
195
196         delayBox = new QDoubleSpinBox(timeTab);
197         delayBox->setObjectName(QStringLiteral("delayBox"));
198         delayBox->setDecimals(1);
199         delayBox->setMaximum(86400);
200
201         timeGrid->addWidget(delayBox, 2, 3, 1, 1);
202
203         kpModeStopBox = new QCheckBox(timeTab);
204         kpModeStopBox->setObjectName(QStringLiteral("kpModeStopBox"));
205
206         timeGrid->addWidget(kpModeStopBox, 5, 1, 1, 3);
207
208         label_10 = new QLabel(timeTab);
209         label_10->setObjectName(QStringLiteral("label_10"));
210
211         timeGrid->addWidget(label_10, 2, 1, 1, 1);
212
213         verticalSpacer_3 = new QSpacerItem(0, 10, QSizePolicy::Minimum, QSizePolicy::Fixed)
    ;
214
215         timeGrid->addItem(verticalSpacer_3, 7, 1, 1, 1);
216
217         horizontalSpacer_4 = new QSpacerItem(40, 20, QSizePolicy::Expanding,
    QSizePolicy::Minimum);
218
219         timeGrid->addItem(horizontalSpacer_4, 2, 5, 1, 1);
220
221         kpReleaseBox = new QCheckBox(timeTab);
222         kpReleaseBox->setObjectName(QStringLiteral("kpReleaseBox"));
223
224         timeGrid->addWidget(kpReleaseBox, 14, 1, 1, 3);
225
226         label_6 = new QLabel(timeTab);
227         label_6->setObjectName(QStringLiteral("label_6"));
228
229         timeGrid->addWidget(label_6, 8, 0, 1, 6);
230
231         kpRepeatBox = new QDoubleSpinBox(timeTab);
232         kpRepeatBox->setObjectName(QStringLiteral("kpRepeatBox"));
233         kpRepeatBox->setDecimals(1);
234         kpRepeatBox->setMinimum(0.1);
235         kpRepeatBox->setMaximum(86400);
236
237         timeGrid->addWidget(kpRepeatBox, 11, 3, 1, 1);
238
239         verticalSpacer_2 = new QSpacerItem(20, 40, QSizePolicy::Minimum,
    QSizePolicy::Expanding);
240
241         timeGrid->addItem(verticalSpacer_2, 15, 1, 1, 1);
242
243         line_2 = new QFrame(timeTab);
244         line_2->setObjectName(QStringLiteral("line_2"));
245         line_2->setFrameShape(QFrame::HLine);
246         line_2->setFrameShadow(QFrame::Sunken);
247
248         timeGrid->addWidget(line_2, 9, 0, 1, 6);
249
250         horizontalSpacer_6 = new QSpacerItem(0, 0, QSizePolicy::Fixed,
    QSizePolicy::Minimum);
251
252         timeGrid->addItem(horizontalSpacer_6, 12, 2, 1, 1);
253
```

```
254         tabWidget->addTab(timeTab, QString());
255         infoTab = new QWidget();
256         infoTab->setObjectName(QStringLiteral("infoTab"));
257         gridLayout_2 = new QGridLayout(infoTab);
258         gridLayout_2->setSpacing(12);
259         gridLayout_2->setObjectName(QStringLiteral("gridLayout_2"));
260         gridLayout_2->setContentsMargins(12, 12, 12, 12);
261         sNameLabel = new QLabel(infoTab);
262         sNameLabel->setObjectName(QStringLiteral("sNameLabel"));
263
264         gridLayout_2->addWidget(sNameLabel, 0, 2, 1, 1);
265
266         label_7 = new QLabel(infoTab);
267         label_7->setObjectName(QStringLiteral("label_7"));
268
269         gridLayout_2->addWidget(label_7, 3, 0, 1, 1);
270
271         label_11 = new QLabel(infoTab);
272         label_11->setObjectName(QStringLiteral("label_11"));
273
274         gridLayout_2->addWidget(label_11, 5, 0, 1, 1);
275
276         sAuthorLabel = new QLabel(infoTab);
277         sAuthorLabel->setObjectName(QStringLiteral("sAuthorLabel"));
278
279         gridLayout_2->addWidget(sAuthorLabel, 2, 2, 1, 1);
280
281         label_9 = new QLabel(infoTab);
282         label_9->setObjectName(QStringLiteral("label_9"));
283
284         gridLayout_2->addWidget(label_9, 4, 0, 1, 1);
285
286         label_5 = new QLabel(infoTab);
287         label_5->setObjectName(QStringLiteral("label_5"));
288
289         gridLayout_2->addWidget(label_5, 1, 0, 1, 1);
290
291         sYearLabel = new QLabel(infoTab);
292         sYearLabel->setObjectName(QStringLiteral("sYearLabel"));
293
294         gridLayout_2->addWidget(sYearLabel, 3, 2, 1, 1);
295
296         label_3 = new QLabel(infoTab);
297         label_3->setObjectName(QStringLiteral("label_3"));
298
299         gridLayout_2->addWidget(label_3, 2, 0, 1, 1);
300
301         horizontalSpacer = new QSpacerItem(40, 20, QSizePolicy::Expanding,
    QSizePolicy::Minimum);
302
303         gridLayout_2->addItem(horizontalSpacer, 0, 3, 1, 1);
304
305         horizontalSpacer_2 = new QSpacerItem(20, 20, QSizePolicy::Fixed,
    QSizePolicy::Minimum);
306
307         gridLayout_2->addItem(horizontalSpacer_2, 0, 1, 1, 1);
308
309         sVerLabel = new QLabel(infoTab);
310         sVerLabel->setObjectName(QStringLiteral("sVerLabel"));
311
312         gridLayout_2->addWidget(sVerLabel, 1, 2, 1, 1);
313
314         sLicenseLabel = new QLabel(infoTab);
315         sLicenseLabel->setObjectName(QStringLiteral("sLicenseLabel"));
316
317         gridLayout_2->addWidget(sLicenseLabel, 4, 2, 1, 1);
318
319         label = new QLabel(infoTab);
320         label->setObjectName(QStringLiteral("label"));
321
322         gridLayout_2->addWidget(label, 0, 0, 1, 1);
323
324         verticalSpacer = new QSpacerItem(20, 40, QSizePolicy::Minimum, QSizePolicy::Expanding
    );
325
326         gridLayout_2->addItem(verticalSpacer, 6, 0, 1, 1);
327
328         sDescLabel = new QLabel(infoTab);
329         sDescLabel->setObjectName(QStringLiteral("sDescLabel"));
330         sDescLabel->setAlignment(Qt::AlignLeading|Qt::AlignLeft|Qt::AlignTop);
331
332         gridLayout_2->addWidget(sDescLabel, 5, 2, 2, 2);
333
334         tabWidget->addTab(infoTab, QString());
335
336         gridLayout->addWidget(tabWidget, 0, 0, 1, 1);
337
```

```
338         QWidget::setTabOrder(tabWidget, animName);
339         QWidget::setTabOrder(animName, delayBox);
340         QWidget::setTabOrder(delayBox, repeatBox);
341         QWidget::setTabOrder(repeatBox, kpModeStopBox);
342         QWidget::setTabOrder(kpModeStopBox, kpDelayBox);
343         QWidget::setTabOrder(kpDelayBox, kpRepeatBox);
344         QWidget::setTabOrder(kpRepeatBox, kpReleaseBox);
345
346         retranslateUi(AnimSettingDialog);
347         QObject::connect(buttonBox, SIGNAL(accepted()), AnimSettingDialog, SLOT(accept()));
348         QObject::connect(buttonBox, SIGNAL(rejected()), AnimSettingDialog, SLOT(reject()));
349
350         tabWidget->setCurrentIndex(0);
351
352
353         QMetaObject::connectSlotsByName(AnimSettingDialog);
354     } // setupUi
```

Here is the call graph for this function:

Here is the caller graph for this function:



12.4.3 Field Documentation

12.4.3.1 QLineEdit* Ui_AnimSettingDialog::animName [inherited]

Definition at line 41 of file ui_animsettingdialog.h.

Referenced by AnimSettingDialog::AnimSettingDialog(), AnimSettingDialog::name(), and Ui_AnimSettingDialog-::setupUi().

12.4.3.2 QDialogButtonBox* Ui_AnimSettingDialog::buttonBox [inherited]

Definition at line 35 of file ui_animsettingdialog.h.

Referenced by Ui_AnimSettingDialog::setupUi().

12.4.3.3 QDoubleSpinBox* Ui_AnimSettingDialog::delayBox [inherited]

Definition at line 56 of file ui_animsettingdialog.h.

Referenced by AnimSettingDialog::AnimSettingDialog(), and Ui_AnimSettingDialog::setupUi().

12.4.3.4 QGridLayout* Ui_AnimSettingDialog::gridLayout [inherited]

Definition at line 34 of file ui_animsettingdialog.h.

Referenced by Ui_AnimSettingDialog::setupUi().

12.4.3.5 QGridLayout* Ui_AnimSettingDialog::gridLayout_2 [inherited]

Definition at line 68 of file ui_animsettingdialog.h.

Referenced by Ui_AnimSettingDialog::setupUi().

12.4.3.6 QSpacerItem* Ui_AnimSettingDialog::horizontalSpacer [inherited]

Definition at line 77 of file ui_animsettingdialog.h.

Referenced by Ui_AnimSettingDialog::setupUi().

12.4.3.7 QSpacerItem* Ui_AnimSettingDialog::horizontalSpacer_2 [inherited]

Definition at line 78 of file ui_animsettingdialog.h.

Referenced by Ui_AnimSettingDialog::setupUi().

12.4.3.8 QSpacerItem* Ui_AnimSettingDialog::horizontalSpacer_3 [inherited]

Definition at line 40 of file ui_animsettingdialog.h.

Referenced by Ui_AnimSettingDialog::setupUi().

12.4.3.9 QSpacerItem* Ui_AnimSettingDialog::horizontalSpacer_4 [inherited]

Definition at line 60 of file ui_animsettingdialog.h.

Referenced by Ui_AnimSettingDialog::setupUi().

12.4.3.10 QSpacerItem* Ui_AnimSettingDialog::horizontalSpacer_5 [inherited]

Definition at line 44 of file ui_animsettingdialog.h.

Referenced by Ui_AnimSettingDialog::setupUi().

12.4.3.11 QSpacerItem* Ui_AnimSettingDialog::horizontalSpacer_6 [inherited]

Definition at line 66 of file ui_animsettingdialog.h.

Referenced by Ui_AnimSettingDialog::setupUi().

12.4.3.12 QWidget* Ui_AnimSettingDialog::infoTab [inherited]

Definition at line 67 of file ui_animsettingdialog.h.

Referenced by Ui_AnimSettingDialog::retranslateUi(), and Ui_AnimSettingDialog::setupUi().

12.4.3.13 QDoubleSpinBox* Ui_AnimSettingDialog::kpDelayBox [inherited]

Definition at line 53 of file ui_animsettingdialog.h.

Referenced by AnimSettingDialog::AnimSettingDialog(), and Ui_AnimSettingDialog::setupUi().

12.4.3.14 QCheckBox* Ui_AnimSettingDialog::kpModeStopBox [inherited]

Definition at line 57 of file ui_animsettingdialog.h.

Referenced by AnimSettingDialog::AnimSettingDialog(), Ui_AnimSettingDialog::retranslateUi(), and Ui_AnimSettingDialog::setupUi().

12.4.3.15 QCheckBox* Ui_AnimSettingDialog::kpReleaseBox [inherited]

Definition at line 61 of file ui_animsettingdialog.h.

Referenced by AnimSettingDialog::AnimSettingDialog(), Ui_AnimSettingDialog::retranslateUi(), and Ui_AnimSettingDialog::setupUi().

12.4.3.16 QDoubleSpinBox* Ui_AnimSettingDialog::kpRepeatBox [inherited]

Definition at line 63 of file ui_animsettingdialog.h.

Referenced by AnimSettingDialog::AnimSettingDialog(), and Ui_AnimSettingDialog::setupUi().

12.4.3.17 QLabel* Ui_AnimSettingDialog::kpRepeatLabel [inherited]

Definition at line 45 of file ui_animsettingdialog.h.

Referenced by AnimSettingDialog::AnimSettingDialog(), Ui_AnimSettingDialog::retranslateUi(), and Ui_AnimSettingDialog::setupUi().

12.4.3.18 QLabel* Ui_AnimSettingDialog::kpRepeatLabel_2 [inherited]

Definition at line 50 of file ui_animsettingdialog.h.

Referenced by AnimSettingDialog::AnimSettingDialog(), Ui_AnimSettingDialog::retranslateUi(), and Ui_AnimSettingDialog::setupUi().

12.4.3.19 QLabel* Ui_AnimSettingDialog::label [inherited]

Definition at line 81 of file ui_animsettingdialog.h.

Referenced by Ui_AnimSettingDialog::retranslateUi(), and Ui_AnimSettingDialog::setupUi().

12.4.3.20 QLabel* Ui_AnimSettingDialog::label_10 [inherited]

Definition at line 58 of file ui_animsettingdialog.h.

Referenced by Ui_AnimSettingDialog::retranslateUi(), and Ui_AnimSettingDialog::setupUi().

12.4.3.21 QLabel* Ui_AnimSettingDialog::label_11 [inherited]

Definition at line 71 of file ui_animsettingdialog.h.

Referenced by Ui_AnimSettingDialog::retranslateUi(), and Ui_AnimSettingDialog::setupUi().

12.4.3.22 QLabel* Ui_AnimSettingDialog::label_12 [inherited]

Definition at line 55 of file ui_animsettingdialog.h.

Referenced by Ui_AnimSettingDialog::retranslateUi(), and Ui_AnimSettingDialog::setupUi().

12.4.3.23 QLabel* Ui_AnimSettingDialog::label_13 [inherited]

Definition at line 54 of file ui_animsettingdialog.h.

Referenced by Ui_AnimSettingDialog::retranslateUi(), and Ui_AnimSettingDialog::setupUi().

12.4.3.24 QLabel* Ui_AnimSettingDialog::label_2 [inherited]

Definition at line 39 of file ui_animsettingdialog.h.

Referenced by Ui_AnimSettingDialog::retranslateUi(), and Ui_AnimSettingDialog::setupUi().

12.4.3.25 QLabel* Ui_AnimSettingDialog::label_3 [inherited]

Definition at line 76 of file ui_animsettingdialog.h.

Referenced by Ui_AnimSettingDialog::retranslateUi(), and Ui_AnimSettingDialog::setupUi().

12.4.3.26 QLabel* Ui_AnimSettingDialog::label_4 [inherited]

Definition at line 51 of file ui_animsettingdialog.h.

Referenced by Ui_AnimSettingDialog::retranslateUi(), and Ui_AnimSettingDialog::setupUi().

12.4.3.27 QLabel* Ui_AnimSettingDialog::label_5 [inherited]

Definition at line 74 of file ui_animsettingdialog.h.

Referenced by Ui_AnimSettingDialog::retranslateUi(), and Ui_AnimSettingDialog::setupUi().

12.4.3.28 QLabel* Ui_AnimSettingDialog::label_6 [inherited]

Definition at line 62 of file ui_animsettingdialog.h.

Referenced by Ui_AnimSettingDialog::retranslateUi(), and Ui_AnimSettingDialog::setupUi().

12.4.3.29 QLabel* Ui_AnimSettingDialog::label_7 [inherited]

Definition at line 70 of file ui_animsettingdialog.h.

Referenced by Ui_AnimSettingDialog::retranslateUi(), and Ui_AnimSettingDialog::setupUi().

12.4.3.30 QLabel* Ui_AnimSettingDialog::label_8 [inherited]

Definition at line 52 of file ui_animsettingdialog.h.

Referenced by Ui_AnimSettingDialog::retranslateUi(), and Ui_AnimSettingDialog::setupUi().

12.4.3.31 QLabel* Ui_AnimSettingDialog::label_9 [inherited]

Definition at line 73 of file ui_animsettingdialog.h.

Referenced by Ui_AnimSettingDialog::retranslateUi(), and Ui_AnimSettingDialog::setupUi().

12.4.3.32 QFrame* Ui_AnimSettingDialog::line [inherited]

Definition at line 49 of file ui_animsettingdialog.h.

Referenced by Ui_AnimSettingDialog::setupUi().

12.4.3.33 QFrame* Ui_AnimSettingDialog::line_2 [inherited]

Definition at line 65 of file ui_animsettingdialog.h.

Referenced by Ui_AnimSettingDialog::setupUi().

12.4.3.34 QDoubleSpinBox* Ui_AnimSettingDialog::repeatBox [inherited]

Definition at line 47 of file ui_animsettingdialog.h.

Referenced by AnimSettingDialog::AnimSettingDialog(), and Ui_AnimSettingDialog::setupUi().

12.4.3.35 QLabel* Ui_AnimSettingDialog::repeatLabel [inherited]

Definition at line 46 of file ui_animsettingdialog.h.

Referenced by AnimSettingDialog::AnimSettingDialog(), Ui_AnimSettingDialog::retranslateUi(), and Ui_AnimSettingDialog::setupUi().

12.4.3.36 QLabel* Ui_AnimSettingDialog::repeatLabel_2 [inherited]

Definition at line 48 of file ui_animsettingdialog.h.

Referenced by AnimSettingDialog::AnimSettingDialog(), Ui_AnimSettingDialog::retranslateUi(), and Ui_AnimSettingDialog::setupUi().

12.4.3.37 QLabel* Ui_AnimSettingDialog::sAuthorLabel [inherited]

Definition at line 72 of file ui_animsettingdialog.h.

Referenced by AnimSettingDialog::AnimSettingDialog(), Ui_AnimSettingDialog::retranslateUi(), and Ui_AnimSettingDialog::setupUi().

12.4.3.38 QLabel* Ui_AnimSettingDialog::sDescLabel [inherited]

Definition at line 83 of file ui_animsettingdialog.h.

Referenced by AnimSettingDialog::AnimSettingDialog(), Ui_AnimSettingDialog::retranslateUi(), and Ui_AnimSettingDialog::setupUi().

12.4.3.39 QGridLayout* Ui_AnimSettingDialog::settingsGrid [inherited]

Definition at line 38 of file ui_animsettingdialog.h.

Referenced by AnimSettingDialog::AnimSettingDialog(), and Ui_AnimSettingDialog::setupUi().

12.4.3.40 QWidget* Ui_AnimSettingDialog::settingTab [inherited]

Definition at line 37 of file ui_animsettingdialog.h.

Referenced by Ui_AnimSettingDialog::retranslateUi(), and Ui_AnimSettingDialog::setupUi().

12.4.3.41 QLabel* Ui_AnimSettingDialog::sLicenseLabel [inherited]

Definition at line 80 of file ui_animsettingdialog.h.

Referenced by AnimSettingDialog::AnimSettingDialog(), Ui_AnimSettingDialog::retranslateUi(), and Ui_AnimSettingDialog::setupUi().

12.4.3.42 QLabel* Ui_AnimSettingDialog::sNameLabel [inherited]

Definition at line 69 of file ui_animsettingdialog.h.

Referenced by AnimSettingDialog::AnimSettingDialog(), Ui_AnimSettingDialog::retranslateUi(), and Ui_AnimSettingDialog::setupUi().

12.4.3.43 QLabel* Ui_AnimSettingDialog::sVerLabel [inherited]

Definition at line 79 of file ui_animsettingdialog.h.

Referenced by AnimSettingDialog::AnimSettingDialog(), Ui_AnimSettingDialog::retranslateUi(), and Ui_AnimSettingDialog::setupUi().

12.4.3.44 QLabel* Ui_AnimSettingDialog::sYearLabel [inherited]

Definition at line 75 of file ui_animsettingdialog.h.

Referenced by AnimSettingDialog::AnimSettingDialog(), Ui_AnimSettingDialog::retranslateUi(), and Ui_AnimSettingDialog::setupUi().

12.4.3.45 QTabWidget* Ui_AnimSettingDialog::tabWidget [inherited]

Definition at line 36 of file ui_animsettingdialog.h.

Referenced by Ui_AnimSettingDialog::retranslateUi(), and Ui_AnimSettingDialog::setupUi().

12.4.3.46 QGridLayout* Ui_AnimSettingDialog::timeGrid [inherited]

Definition at line 43 of file ui_animsettingdialog.h.

Referenced by AnimSettingDialog::AnimSettingDialog(), and Ui_AnimSettingDialog::setupUi().

12.4.3.47 QWidget* Ui_AnimSettingDialog::timeTab [inherited]

Definition at line 42 of file ui_animsettingdialog.h.

Referenced by Ui_AnimSettingDialog::retranslateUi(), and Ui_AnimSettingDialog::setupUi().

12.4.3.48 QSpacerItem* Ui_AnimSettingDialog::verticalSpacer [inherited]

Definition at line 82 of file ui_animsettingdialog.h.

Referenced by Ui_AnimSettingDialog::setupUi().

12.4.3.49 QSpacerItem* Ui_AnimSettingDialog::verticalSpacer_2 [inherited]

Definition at line 64 of file ui_animsettingdialog.h.

Referenced by Ui_AnimSettingDialog::setupUi().

12.4.3.50 QSpacerItem* Ui_AnimSettingDialog::verticalSpacer_3 [inherited]

Definition at line 59 of file ui_animsettingdialog.h.

Referenced by Ui_AnimSettingDialog::setupUi().

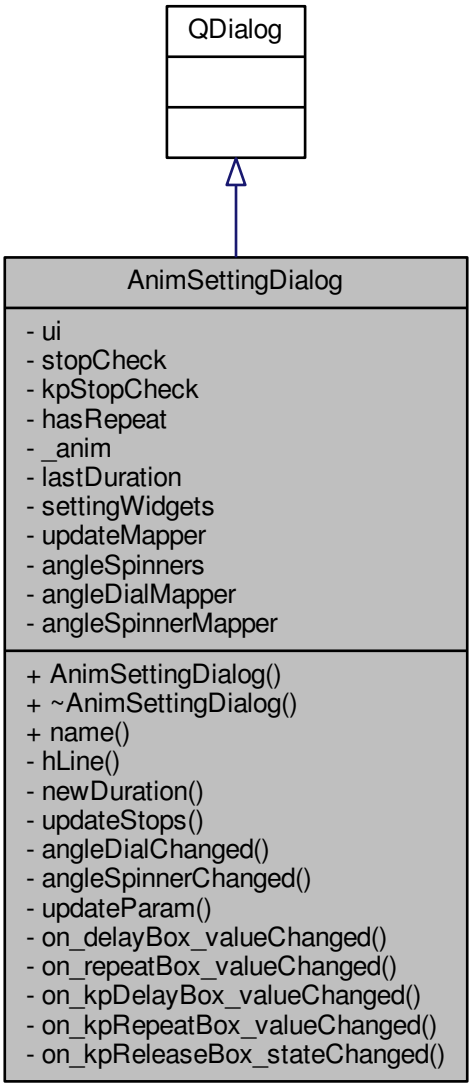
The documentation for this class was generated from the following file:

- src/ckb/ui_animsettingdialog.h

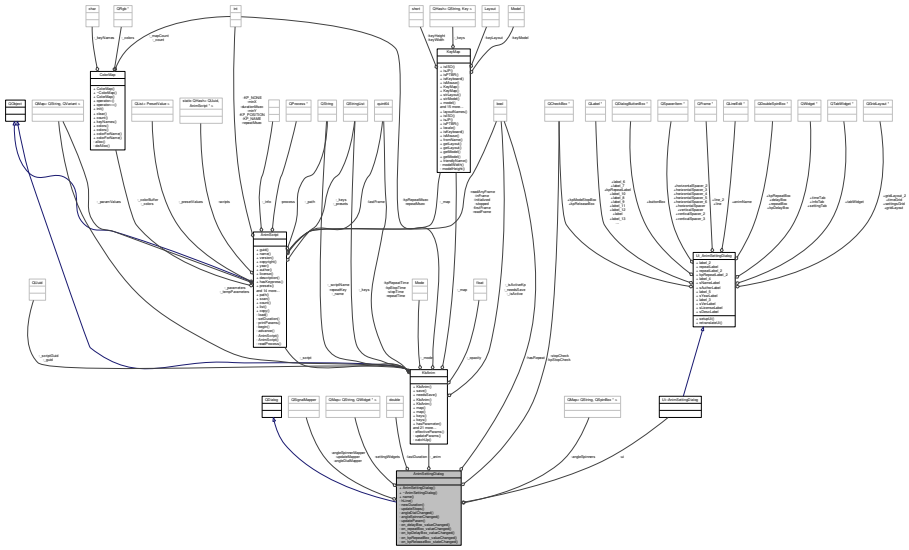
12.5 AnimSettingDialog Class Reference

`#include <src/ckb/animsettingdialog.h>`

Inheritance diagram for AnimSettingDialog:



Collaboration diagram for AnimSettingDialog:



Public Member Functions

- AnimSettingDialog (QWidget ∗parent, KbAnim ∗anim)
- ∼AnimSettingDialog ()
- QString name () const

Private Slots

- void newDuration (double duration)
- void updateStops ()
- void angleDialChanged (QString name)
- void angleSpinnerChanged (QString name)
- void updateParam (QString name)
- void on_delayBox_valueChanged (double arg1)
- void on_repeatBox_valueChanged (double arg1)
- void on_kpDelayBox_valueChanged (double arg1)
- void on_kpRepeatBox_valueChanged (double arg1)
- void on_kpReleaseBox_stateChanged (int arg1)

Private Member Functions

- QFrame ∗ hLine ()

Private Attributes

- Ui::AnimSettingDialog ∗ ui
- QCheckBox ∗ stopCheck
- QCheckBox ∗ kpStopCheck
- bool hasRepeat
- KbAnim ∗ _anim
- double lastDuration
- QMap< QString, QWidget ∗ > settingWidgets

- QSignalMapper updateMapper
- QMap< QString, QSpinBox * > angleSpinners
- QSignalMapper angleDialMapper
- QSignalMapper angleSpinnerMapper

12.5.1 Detailed Description

Definition at line 15 of file animsettingdialog.h.

12.5.2 Constructor & Destructor Documentation

12.5.2.1 AnimSettingDialog::AnimSettingDialog (QWidget * *parent*, KbAnim * *anim*) [explicit]

Definition at line 19 of file animsettingdialog.cpp.

References AnimScript::Param::AGRADIENT, AnimScript::Param::ANGLE, angleDialChanged(), angleDialMapper, angleFlip(), angleSpinnerChanged(), angleSpinnerMapper, angleSpinners, Ui_AnimSettingDialog::animName, AnimScript::Param::ARGB, AnimScript::author(), AnimScript::Param::BOOL, Ui_AnimSettingDialog::delayBox, AnimScript::description(), AnimScript::Param::DOUBLE, AnimScript::Param::GRADIENT, AnimScript::hasKeypress(), AnimScript::hasParam(), KbAnim::hasParameter(), hasRepeat, hLine(), Ui_AnimSettingDialog::kpDelayBox, Ui_AnimSettingDialog::kpModeStopBox, Ui_AnimSettingDialog::kpReleaseBox, Ui_AnimSettingDialog::kpRepeatBox, Ui_AnimSettingDialog::kpRepeatLabel, Ui_AnimSettingDialog::kpRepeatLabel_2, kpStopCheck, AnimScript::Param::LABEL, lastDuration, AnimScript::license(), AnimScript::Param::LONG, AnimScript::Param::maximum, AnimScript::Param::minimum, AnimScript::Param::name, AnimScript::name(), KbAnim::name(), newDuration(), KbAnim::parameter(), AnimScript::paramIterator(), AnimScript::Param::postfix, AnimScript::Param::prefix, Ui_AnimSettingDialog::repeatBox, Ui_AnimSettingDialog::repeatLabel, Ui_AnimSettingDialog::repeatLabel_2, AnimScript::Param::RGB, Ui_AnimSettingDialog::sAuthorLabel, KbAnim::script(), KbAnim::scriptName(), Ui_AnimSettingDialog::sDescLabel, Ui_AnimSettingDialog::settingsGrid, settingWidgets, Ui_AnimSettingDialog::setupUi(), Ui_AnimSettingDialog::sLicenseLabel, Ui_AnimSettingDialog::sNameLabel, stopCheck, AnimScript::Param::STRING, Ui_AnimSettingDialog::sVerLabel, Ui_AnimSettingDialog::sYearLabel, Ui_AnimSettingDialog::timeGrid, AnimScript::Param::type, ui, updateMapper, updateParam(), updateStops(), AnimScript::version(), and AnimScript::year().

```
19                                                                          :
20     QDialog(parent),
21     ui(new Ui::AnimSettingDialog), stopCheck(0),
       kpStopCheck(0),
22     _anim(anim), lastDuration(1.0)
23 {
24     ui->setupUi(this);
25     setWindowTitle(anim->scriptName() + " Animation");
26     ui->animName->setText(anim->name());
27     const AnimScript* script = anim->script();
28
29     connect(&updateMapper, SIGNAL(mapped(QString)), this, SLOT(
       updateParam(QString)));
30     connect(&angleDialMapper, SIGNAL(mapped(QString)), this, SLOT(
       angleDialChanged(QString)));
31     connect(&angleSpinnerMapper, SIGNAL(mapped(QString)), this, SLOT(
       angleSpinnerChanged(QString)));
32
33     // Build settings UI
34     int row = 1;
35     ui->settingsGrid->addItem(new QSpacerItem(0, 10, QSizePolicy::Fixed, QSizePolicy::Fixed),
       row++, 6);
36     ui->settingsGrid->addWidget(new QLabel("<b>Animation</b>", this), row++, 0, 1, 7);
37     ui->settingsGrid->addWidget(hLine(), row++, 0, 1, 7);
38     bool rSpacePlaced = false;
39     QListIterator<AnimScript::Param> i(script->paramIterator());
40     while(i.hasNext()){
41         const AnimScript::Param& param = i.next();
42         // Skip timing/playback params
43         if(param.name == "duration"
44                 || param.name == "trigger" || param.name == "kptrigger"
45                 || param.name == "kpmode"
46                 || param.name == "delay" || param.name == "kpdelay"
47                 || param.name == "repeat" || param.name == "kprepeat"
48                 || param.name == "stop" || param.name == "kpstop"
```

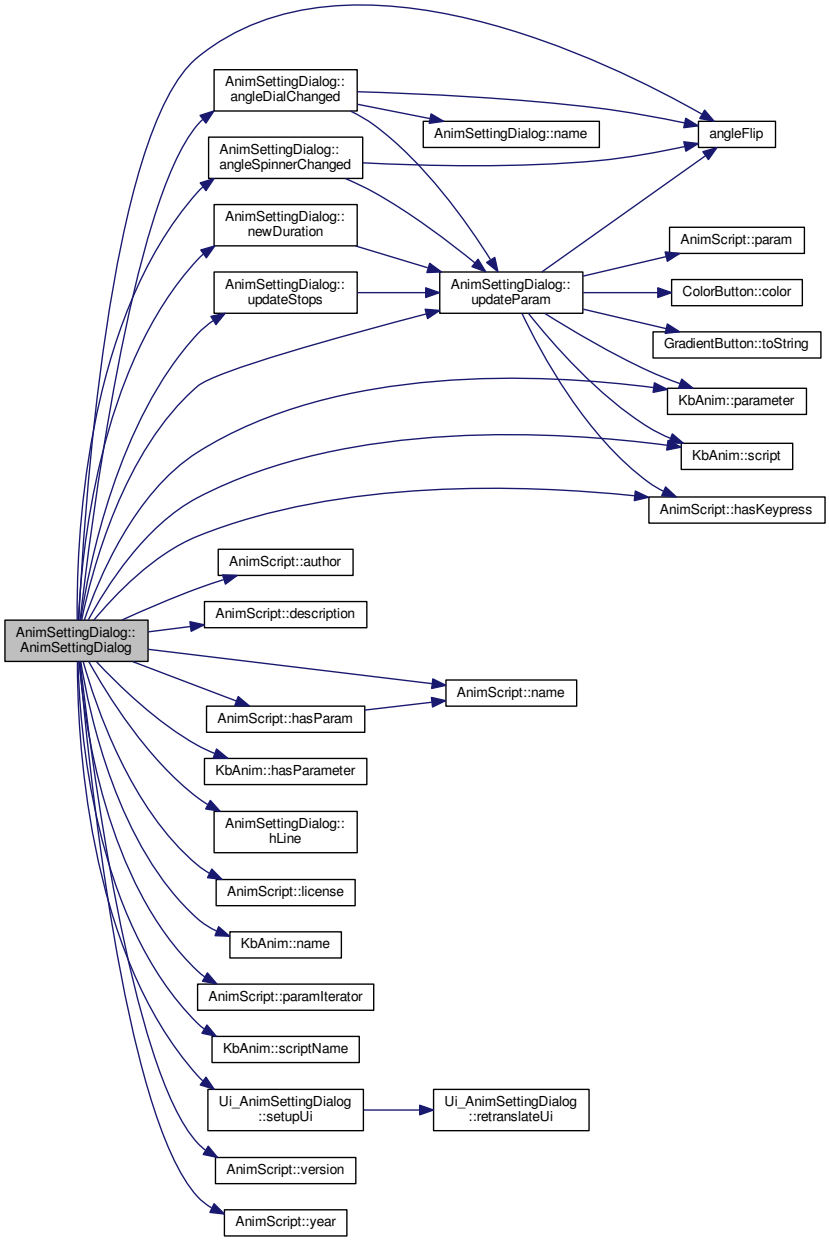
```
49                  || param.name == "kpmodestop" || param.name == "kprelease")
50              continue;
51          QVariant value = anim->parameter(param.name);
52          // Display prefix label on the left (except for booleans and labels)
53          if(param.type != AnimScript::Param::BOOL && param.
    type != AnimScript::Param::LABEL)
54              ui->settingsGrid->addWidget(new QLabel(param.prefix, this), row, 1);
55          // Configure and display main widget
56          QWidget* widget = 0;
57          int colSpan = 1;
58          QString postfix = param.postfix;
59          switch(param.type){
60          case AnimScript::Param::BOOL:
61              widget = new QCheckBox(param.prefix, this);
62              ((QCheckBox*)widget)->setChecked(value.toBool());
63              colSpan = 4;
64              connect(widget, SIGNAL(stateChanged(int)), &updateMapper, SLOT(map()));
65              break;
66          case AnimScript::Param::LONG:
67              widget = new QSpinBox(this);
68              ((QSpinBox*)widget)->setMinimum(param.minimum.toInt());
69              ((QSpinBox*)widget)->setMaximum(param.maximum.toInt());
70              ((QSpinBox*)widget)->setValue(value.toInt());
71              if(postfix.length() <= 3){
72                  ((QSpinBox*)widget)->setSuffix(postfix);
73                  postfix = "";
74              }
75              connect(widget, SIGNAL(valueChanged(int)), &updateMapper, SLOT(map()));
76              break;
77          case AnimScript::Param::DOUBLE:
78              widget = new QDoubleSpinBox(this);
79              ((QDoubleSpinBox*)widget)->setDecimals(1);
80              ((QDoubleSpinBox*)widget)->setMinimum(param.minimum.toDouble());
81              ((QDoubleSpinBox*)widget)->setMaximum(param.maximum.toDouble());
82              ((QDoubleSpinBox*)widget)->setValue(value.toDouble());
83              if(postfix.length() <= 3){
84                  ((QDoubleSpinBox*)widget)->setSuffix(postfix);
85                  postfix = "";
86              }
87              connect(widget, SIGNAL(valueChanged(double)), &updateMapper, SLOT(map()));
88              break;
89          case AnimScript::Param::RGB:
90              widget = new ColorButton(this);
91              ((ColorButton*)widget)->color(QColor("#" + value.toString()));
92              colSpan = 3;
93              connect(widget, SIGNAL(colorChanged(QColor)), &updateMapper, SLOT(map()));
94              break;
95          case AnimScript::Param::ARGB:{
96              widget = new ColorButton(this, true);
97              QString val = value.toString();
98              QColor color;
99              if(val.length() == 8){
100                 color = "#" + val.right(6);
101                 color.setAlpha(val.left(2).toInt(0, 16));
102             } else
103                 color = "#" + val;
104             ((ColorButton*)widget)->color(color);
105             colSpan = 3;
106             connect(widget, SIGNAL(colorChanged(QColor)), &updateMapper, SLOT(map()));
107             break;
108         }
109         case AnimScript::Param::GRADIENT:
110             widget = new GradientButton(this);
111             ((GradientButton*)widget)->fromString(value.toString());
112             colSpan = 3;
113             connect(widget, SIGNAL(gradientChanged()), &updateMapper, SLOT(map()));
114             break;
115         case AnimScript::Param::AGRADIENT:
116             widget = new GradientButton(this, true);
117             ((GradientButton*)widget)->fromString(value.toString());
118             colSpan = 3;
119             connect(widget, SIGNAL(gradientChanged()), &updateMapper, SLOT(map()));
120             break;
121         case AnimScript::Param::ANGLE:
122             widget = new QDial(this);
123             ((QDial*)widget)->setFixedSize(60, 60);
124             ((QDial*)widget)->setMinimum(0);
125             ((QDial*)widget)->setMaximum(360);
126             // this is NOT a typo...     ^
127             // Even though 360 shouldn't be valid, the wheel/arrow keys will lose 1 degree at the bottom
    unless 360 is set as max
128             ((QDial*)widget)->setSingleStep(5);
129             ((QDial*)widget)->setPageStep(15);
130             ((QDial*)widget)->setNotchTarget(6);
131             ((QDial*)widget)->setNotchesVisible(true);
132             ((QDial*)widget)->setWrapping(true);
133             ((QDial*)widget)->setInvertedAppearance(true);
```

```
134             ((QDial*)widget)->setValue(angleFlip(value.toInt()));
135             angleDialMapper.setMapping(widget, param.name);
136             connect(widget, SIGNAL(valueChanged(int)), &angleDialMapper, SLOT(map()));
137             break;
138         case AnimScript::Param::STRING:
139             widget = new QLineEdit(this);
140             ((QLineEdit*)widget)->setText(value.toString());
141             colSpan = 3;
142             connect(widget, SIGNAL(textEdited(const QString&)), &updateMapper, SLOT(map()));
143             break;
144         case AnimScript::Param::LABEL:
145             widget = new QLabel(this);
146             ((QLabel*)widget)->setText(param.prefix);
147             colSpan = 4;
148             break;
149         default:
150             break;
151         }
152         if(widget)
153             updateMapper.setMapping(widget, param.name);
154         if(param.type == AnimScript::Param::BOOL || param.
    type == AnimScript::Param::LABEL){
155             // Boolean values are placed on the left with no prefix or postfix
156             settingWidgets[param.name] = widget;
157             ui->settingsGrid->addWidget(widget, row, 3, 1, colSpan);
158         } else {
159             // Display the widget
160             if(widget){
161                 settingWidgets[param.name] = widget;
162                 ui->settingsGrid->addWidget(widget, row, 3, 1, colSpan);
163             }
164             // Angles additionally have a spin box
165             if(param.type == AnimScript::Param::ANGLE){
166                 QSpinBox* spinner = new QSpinBox(this);
167                 spinner->setMinimum(0);
168                 spinner->setMaximum(359);
169                 spinner->setWrapping(true);
170                 spinner->setSuffix("°");
171                 spinner->setValue(value.toInt());
172                 angleSpinners[param.name] = spinner;
173                 angleSpinnerMapper.setMapping(spinner, param.
    name);
174                 connect(spinner, SIGNAL(valueChanged(int)), &angleSpinnerMapper, SLOT(map
    ()));
175                 ui->settingsGrid->addWidget(spinner, row, 4);
176                 colSpan = 2;
177             }
178             // Display postfix label on the right
179             ui->settingsGrid->addWidget(new QLabel(postfix, this), row, 3 + colSpan, 1, 4 -
    colSpan);
180             if(colSpan < 3 && !rSpacePlaced){
181                 // Add a spacer to compress short elements to the left
182                 ui->settingsGrid->addItem(new QSpacerItem(0, 0, QSizePolicy::Minimum), row, 4
     + colSpan);
183                 rSpacePlaced = true;
184             }
185         }
186         row++;
187     }
188     // Add playback info at bottom
189     ui->settingsGrid->addItem(new QSpacerItem(0, 10, QSizePolicy::Fixed, QSizePolicy::Fixed),
     row++, 6);
190     ui->settingsGrid->addWidget(new QLabel("<b>Playback</b>", this), row++, 0, 1, 7);
191     ui->settingsGrid->addWidget(hLine(), row++, 0, 1, 7);
192     if(script->hasParam("duration")){
193         // Show duration spinner (if allowed)
194         lastDuration = anim->parameter("duration").toDouble();
195         ui->settingsGrid->addWidget(new QLabel("Duration:", this), row, 1);
196         QDoubleSpinBox* spinner = new QDoubleSpinBox(this);
197         spinner->setDecimals(1);
198         spinner->setMinimum(0.1);
199         spinner->setValue(lastDuration);
200         connect(spinner, SIGNAL(valueChanged(double)), this, SLOT(newDuration(double)));
201         settingWidgets["duration"] = spinner;
202         ui->settingsGrid->addWidget(spinner, row, 3, 1, 1);
203         ui->settingsGrid->addWidget(new QLabel("seconds", this), row, 4, 1, 2);
204         row++;
205     }
206     // Show boxes for start with mode/with keypress
207     QCheckBox* check = new QCheckBox("Start with mode", this);
208     check->setChecked(anim->parameter("trigger").toBool());
209     ui->settingsGrid->addWidget(check, row, 3, 1, 4);
210     settingWidgets["trigger"] = check;
211     connect(check, SIGNAL(stateChanged(int)), &updateMapper, SLOT(map()));
212     updateMapper.setMapping(check, "trigger");
213     row++;
214     check = new QCheckBox("Start with key press", this);
```

```
215     check->setChecked(anim->parameter("kptrigger").toBool());
216     ui->settingsGrid->addWidget(check, row, 3, 1, 2);
217     settingWidgets["kptrigger"] = check;
218     connect(check, SIGNAL(stateChanged(int)), &updateMapper, SLOT(map()));
219     updateMapper.setMapping(check, "kptrigger");
220     // Add an option allowing the user to select keypress mode
221     QComboBox* combo = new QComboBox(this);
222     int selected = anim->parameter("kpmode").toInt();
223     if(script->hasKeypress()){
224         // If the script supports keypresses, show the option to handle them that way (default)
225         combo->addItem("on pressed key");
226         combo->addItem("on whole keyboard");
227         combo->addItem("on keyboard (once)");
228     } else {
229         selected--;
230         // Otherwise, just show the choice of whether to start it every time or just once
231         combo->addItem("every time");
232         combo->addItem("only once");
233     }
234     if(selected < 0 || selected > combo->count())
235         selected = 0;
236     combo->setCurrentIndex(selected);
237     ui->settingsGrid->addWidget(combo, row, 5, 1, 2);
238     settingWidgets["kpmode"] = combo;
239     connect(combo, SIGNAL(activated(int)), &updateMapper, SLOT(map()));
240     updateMapper.setMapping(combo, "kpmode");
241     row++;
242
243     // Add horizontal spacer to compress content to left
244     ui->settingsGrid->addItem(new QSpacerItem(0, 0, QSizePolicy::Expanding), 0, 6);
245     // Add vertical spacer to compress content to top
246     ui->settingsGrid->addItem(new QSpacerItem(0, 0, QSizePolicy::Minimum,
    QSizePolicy::Expanding), row, 0);
247
248     // Add timing fields
249     settingWidgets["delay"] = ui->delayBox;
250     ui->delayBox->setValue(anim->parameter("delay").toDouble());
251     settingWidgets["kpdelay"] = ui->kpDelayBox;
252     ui->kpDelayBox->setValue(anim->parameter("kpdelay").toDouble());
253     settingWidgets["kpmodestop"] = ui->kpModeStopBox;
254     ui->kpModeStopBox->setChecked(anim->parameter("kpmodestop").toBool());
255     connect(ui->kpModeStopBox, SIGNAL(clicked(bool)), &
    updateMapper, SLOT(map()));
256     updateMapper.setMapping(ui->kpModeStopBox, "kpmodestop");
257     settingWidgets["kprelease"] = ui->kpReleaseBox;
258     ui->kpReleaseBox->setChecked(anim->parameter("kprelease").toBool());
259     if(anim->hasParameter("repeat")){
260         hasRepeat = true;
261         settingWidgets["repeat"] = ui->repeatBox;
262         ui->repeatBox->setValue(anim->parameter("repeat").toDouble());
263         settingWidgets["kprepeat"] = ui->kpRepeatBox;
264         ui->kpRepeatBox->setValue(anim->parameter("kprepeat").toDouble());
265         // If repeat is enabled, add repeat counts as integer values
266         // Mode repeat
267         ui->timeGrid->addWidget(new QLabel("Repeat:", this), 4, 1);
268         QSpinBox* spinner = new QSpinBox(this);
269         spinner->setMinimum(0);
270         spinner->setMaximum(1000000);
271         spinner->setValue(anim->parameter("stop").toInt());
272         settingWidgets["stop"] = spinner;
273         connect(spinner, SIGNAL(valueChanged(int)), &updateMapper, SLOT(map()));
274         updateMapper.setMapping(spinner, "stop");
275         ui->timeGrid->addWidget(spinner, 4, 3);
276         ui->timeGrid->addWidget(new QLabel("times", this), 4, 4);
277         // KP repeat
278         ui->timeGrid->addWidget(new QLabel("Repeat:", this), 12, 1);
279         spinner = new QSpinBox(this);
280         spinner->setMinimum(0);
281         spinner->setMaximum(1000000);
282         spinner->setValue(anim->parameter("kpstop").toInt());
283         settingWidgets["kpstop"] = spinner;
284         connect(spinner, SIGNAL(valueChanged(int)), &updateMapper, SLOT(map()));
285         updateMapper.setMapping(spinner, "kpstop");
286         ui->timeGrid->addWidget(spinner, 12, 3);
287         ui->timeGrid->addWidget(new QLabel("times", this), 12, 4);
288         // Infinite repeat toggles
289         stopCheck = new QCheckBox("Forever", this);
290         stopCheck->setChecked(anim->parameter("stop").toInt() < 0);
291         ui->timeGrid->addWidget(stopCheck, 4, 5);
292         kpStopCheck = new QCheckBox("Forever", this);
293         connect(stopCheck, SIGNAL(clicked()), this, SLOT(updateStops()));
294         kpStopCheck->setChecked(anim->parameter("kpstop").toInt() < 0);
295         ui->timeGrid->addWidget(kpStopCheck, 12, 5);
296         connect(kpStopCheck, SIGNAL(clicked()), this, SLOT(
    updateStops()));
297     } else {
298         hasRepeat = false;
```

```
299         // If repeat is not enabled, hide repeat-related fields
300         ui->repeatBox->setHidden(true);
301         ui->repeatLabel->setHidden(true);
302         ui->repeatLabel_2->setHidden(true);
303         ui->kpRepeatBox->setHidden(true);
304         ui->kpRepeatLabel->setHidden(true);
305         ui->kpRepeatLabel_2->setHidden(true);
306         // Add stop times as double values
307         // Stop time
308         QDoubleSpinBox* spinner = new QDoubleSpinBox(this);
309         spinner->setDecimals(1);
310         spinner->setMinimum(0.1);
311         spinner->setMaximum(24. * 60. * 60.);
312         double stop = anim->parameter("stop").toDouble();
313         if(stop <= 0.)
314             spinner->setValue(lastDuration);
315         else
316             spinner->setValue(stop);
317         settingWidgets["stop"] = spinner;
318         connect(spinner, SIGNAL(valueChanged(double)), &updateMapper, SLOT(map()));
319         updateMapper.setMapping(spinner, "stop");
320         ui->timeGrid->addWidget(spinner, 4, 3);
321         ui->timeGrid->addWidget(new QLabel("seconds", this), 4, 4);
322         // KP stop time
323         spinner = new QDoubleSpinBox(this);
324         spinner->setDecimals(1);
325         spinner->setMinimum(0.1);
326         spinner->setMaximum(24. * 60. * 60.);
327         double kpstop = anim->parameter("kpstop").toDouble();
328         if(kpstop <= 0.)
329             spinner->setValue(lastDuration);
330         else
331             spinner->setValue(kpstop);
332         settingWidgets["kpstop"] = spinner;
333         connect(spinner, SIGNAL(valueChanged(double)), &updateMapper, SLOT(map()));
334         updateMapper.setMapping(spinner, "kpstop");
335         ui->timeGrid->addWidget(spinner, 12, 3);
336         ui->timeGrid->addWidget(new QLabel("seconds", this), 12, 4);
337         // Infinite run toggles
338         stopCheck = new QCheckBox("Stop after:", this);
339         stopCheck->setChecked(stop > 0.);
340         ui->timeGrid->addWidget(stopCheck, 4, 1);
341         connect(stopCheck, SIGNAL(clicked()), this, SLOT(updateStops()));
342         kpStopCheck = new QCheckBox("Stop after:", this);
343         kpStopCheck->setChecked(kpstop > 0.);
344         ui->timeGrid->addWidget(kpStopCheck, 12, 1);
345         connect(kpStopCheck, SIGNAL(clicked()), this, SLOT(
    updateStops()));
346     }
347     updateStops();
348
349     // Set script info UI
350     ui->sNameLabel->setText(script->name());
351     ui->sVerLabel->setText(script->version());
352     ui->sAuthorLabel->setText(script->author());
353     ui->sYearLabel->setText(script->year());
354     ui->sLicenseLabel->setText(script->license());
355     ui->sDescLabel->setText(script->description());
356
357     // Lock dialog size
358     setFixedSize(minimumSize());
359 }
```

Here is the call graph for this function:



12.5.2.2 AnimSettingDialog::~AnimSettingDialog ( )

Definition at line 476 of file animsettingdialog.cpp.

References ui.

```
476                                        {
477     delete ui;
478 }
```

12.5.3 Member Function Documentation

12.5.3.1 void AnimSettingDialog::angleDialChanged (QString *name*) [private], [slot]

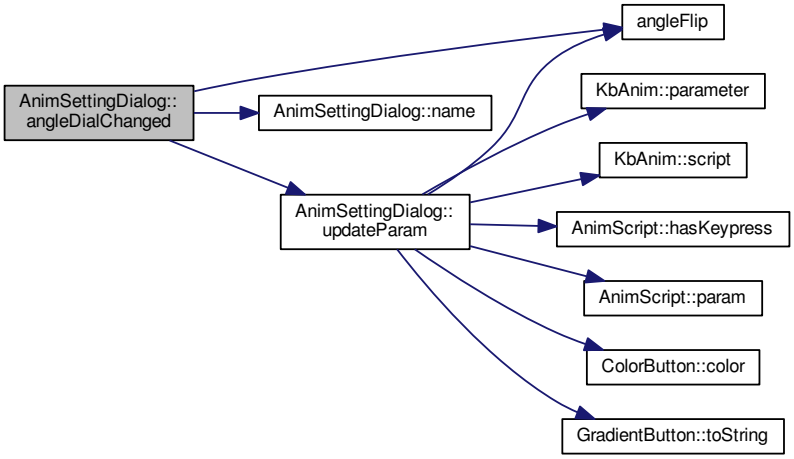
Definition at line 389 of file animsettingdialog.cpp.

References angleFlip(), angleSpinners, name(), settingWidgets, and updateParam().

Referenced by AnimSettingDialog().

```
389                                                 {
390     // Dial changed; update spinner value
391     angleSpinners[name]->setValue(angleFlip(((QDial*)
      settingWidgets[name])->value()));
392     updateParam(name);
393 }
```

Here is the call graph for this function:



Here is the caller graph for this function:

12.5.3.2 void AnimSettingDialog::angleSpinnerChanged (QString *name*) [private], [slot]

Definition at line 395 of file animsettingdialog.cpp.

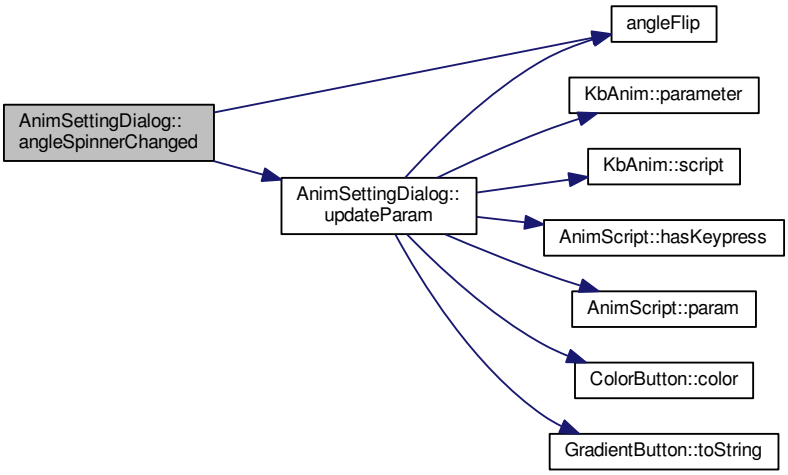
References angleFlip(), angleSpinners, settingWidgets, and updateParam().

Referenced by AnimSettingDialog().

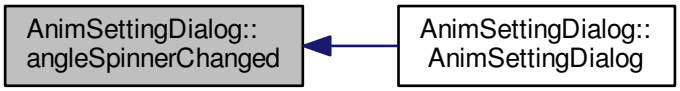
```
395                                                 {
```

```
396     // Spinner changed; update dial value
397     ((QDial*)settingWidgets[name])->setValue(angleFlip(
      angleSpinners[name]->value()));
398     updateParam(name);
399 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.5.3.3 QFrame * AnimSettingDialog::hLine ( ) [private]

Definition at line 480 of file animsettingdialog.cpp.

Referenced by AnimSettingDialog().

```
480                                 {
481     QFrame* frame = new QFrame(this);
482     frame->setFrameShape(QFrame::HLine);
483     frame->setFrameShadow(QFrame::Sunken);
484     return frame;
485 }
```

Here is the caller graph for this function:

12.5.3.4 QString AnimSettingDialog::name ( ) const

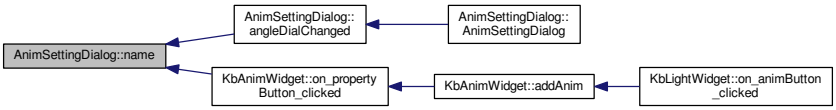
Definition at line 472 of file animsettingdialog.cpp.

References Ui_AnimSettingDialog::animName, and ui.

Referenced by angleDialChanged(), and KbAnimWidget::on_propertyButton_clicked().

```
472                                         {
473     return ui->animName->text();
474 }
```

Here is the caller graph for this function:



12.5.3.5 void AnimSettingDialog::newDuration (double *duration*) [private], [slot]

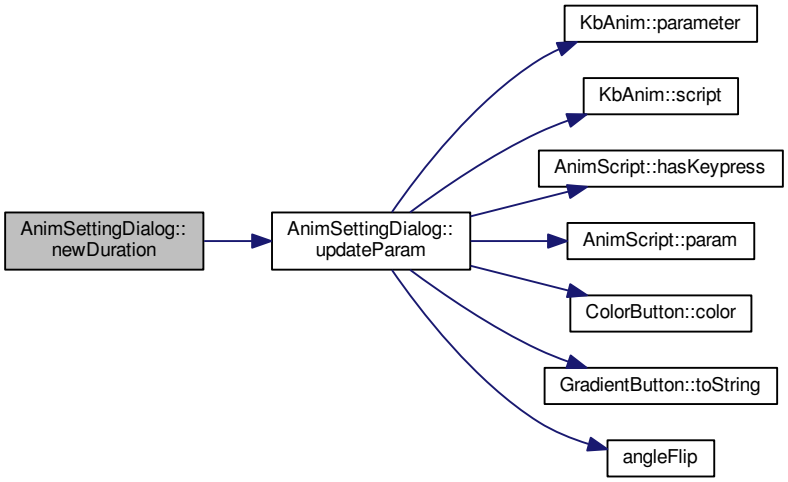
Definition at line 361 of file animsettingdialog.cpp.

References lastDuration, settingWidgets, and updateParam().

Referenced by AnimSettingDialog().

```
361                                                 {
362     // Duration changed. Automatically fix Repeat and KP Repeat to the same values, if they matched before
363     QDoubleSpinBox* rep = (QDoubleSpinBox*)settingWidgets.value("repeat");
364     QDoubleSpinBox* kpRep = (QDoubleSpinBox*)settingWidgets.value("kprepeat");
365     if(rep && rep->value() == lastDuration)
366         rep->setValue(duration);
367     if(kpRep && kpRep->value() == lastDuration)
368         kpRep->setValue(duration);
369     lastDuration = duration;
370     updateParam("duration");
371     updateParam("repeat");
372     updateParam("kprepeat");
373 }
```

Here is the call graph for this function:



Here is the caller graph for this function:

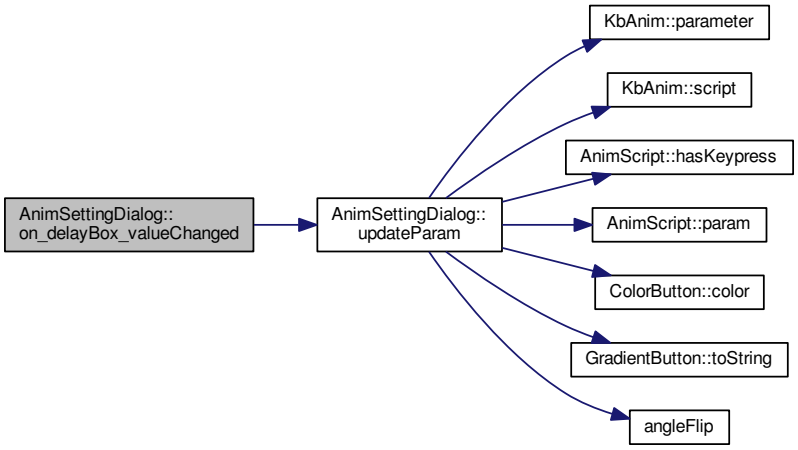
12.5.3.6 void AnimSettingDialog::on_delayBox_valueChanged (double *arg1*) [private], [slot]

Definition at line 487 of file animsettingdialog.cpp.

References updateParam().

```
487                                                    {
488     updateParam("delay");
489 }
```

Here is the call graph for this function:



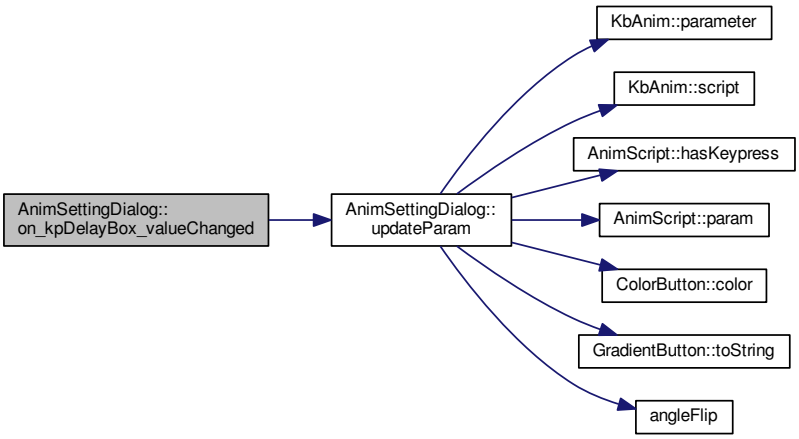
12.5.3.7 void AnimSettingDialog::on_kpDelayBox_valueChanged (double *arg1*) [private], [slot]

Definition at line 495 of file animsettingdialog.cpp.

References updateParam().

```
495                                                    {
496     updateParam("kpdelay");
497 }
```

Here is the call graph for this function:



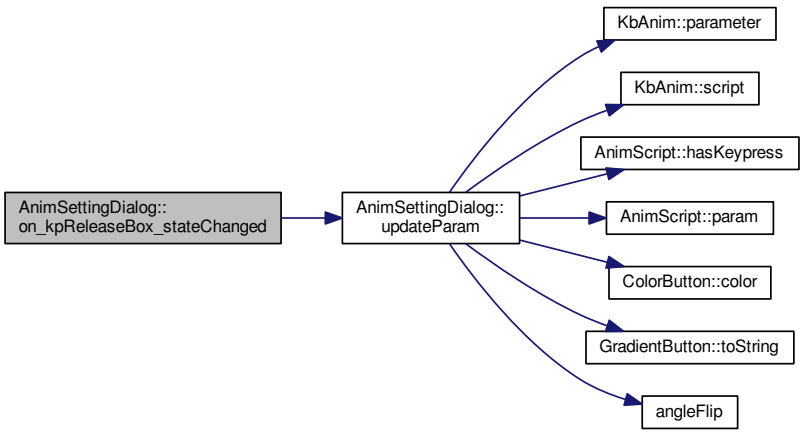
12.5.3.8 void AnimSettingDialog::on_kpReleaseBox_stateChanged (int *arg1*) [private], [slot]

Definition at line 503 of file animsettingdialog.cpp.

References updateParam().

```
503                                                     {
504     updateParam("kprelease");
505 }
```

Here is the call graph for this function:



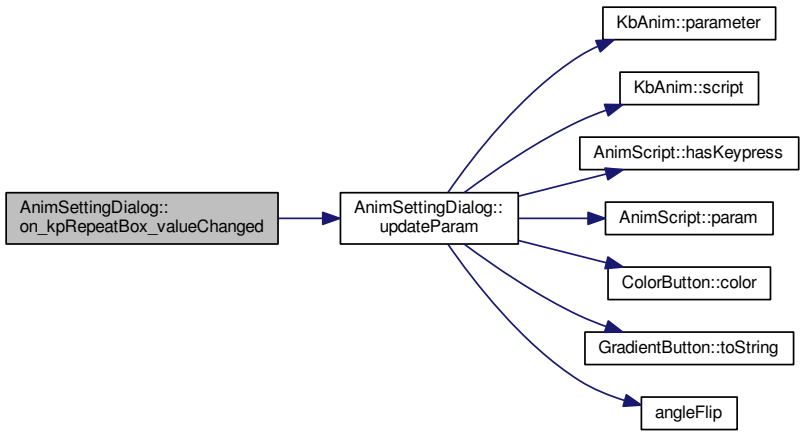
12.5.3.9 void AnimSettingDialog::on_kpRepeatBox_valueChanged (double *arg1*) [private], [slot]

Definition at line 499 of file animsettingdialog.cpp.

References updateParam().

```
499                                                     {
500     updateParam("kprepeat");
501 }
```

Here is the call graph for this function:



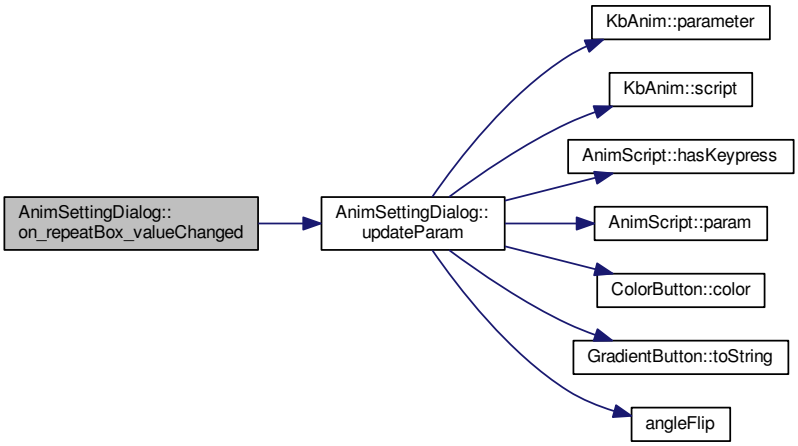
12.5.3.10 void AnimSettingDialog::on_repeatBox_valueChanged (double *arg1*) [private], [slot]

Definition at line 491 of file animsettingdialog.cpp.

References updateParam().

```
491                                                          {
492     updateParam("repeat");
493 }
```

Here is the call graph for this function:



12.5.3.11 void AnimSettingDialog::updateParam (QString *name*) [private], [slot]

Definition at line 401 of file animsettingdialog.cpp.

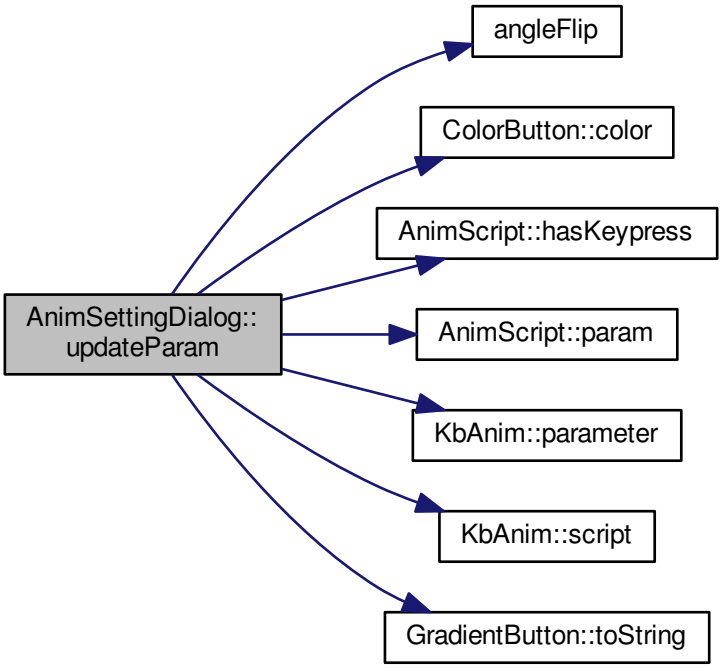
References _anim, AnimScript::Param::AGRADIENT, AnimScript::Param::ANGLE, angleFlip(), AnimScript::Param::ARGB, AnimScript::Param::BOOL, ColorButton::color(), AnimScript::Param::DOUBLE, AnimScript::Param::GRADIENT, AnimScript::hasKeypress(), hasRepeat, kpStopCheck, AnimScript::Param::LONG, AnimScript::param(), KbAnim::parameter(), AnimScript::Param::RGB, KbAnim::script(), settingWidgets, stopCheck, AnimScript::Param::STRING, GradientButton::toString(), and AnimScript::Param::type.

Referenced by angleDialChanged(), angleSpinnerChanged(), AnimSettingDialog(), newDuration(), on_delayBox_valueChanged(), on_kpDelayBox_valueChanged(), on_kpReleaseBox_stateChanged(), on_kpRepeatBox_valueChanged(), on_repeatBox_valueChanged(), and updateStops().

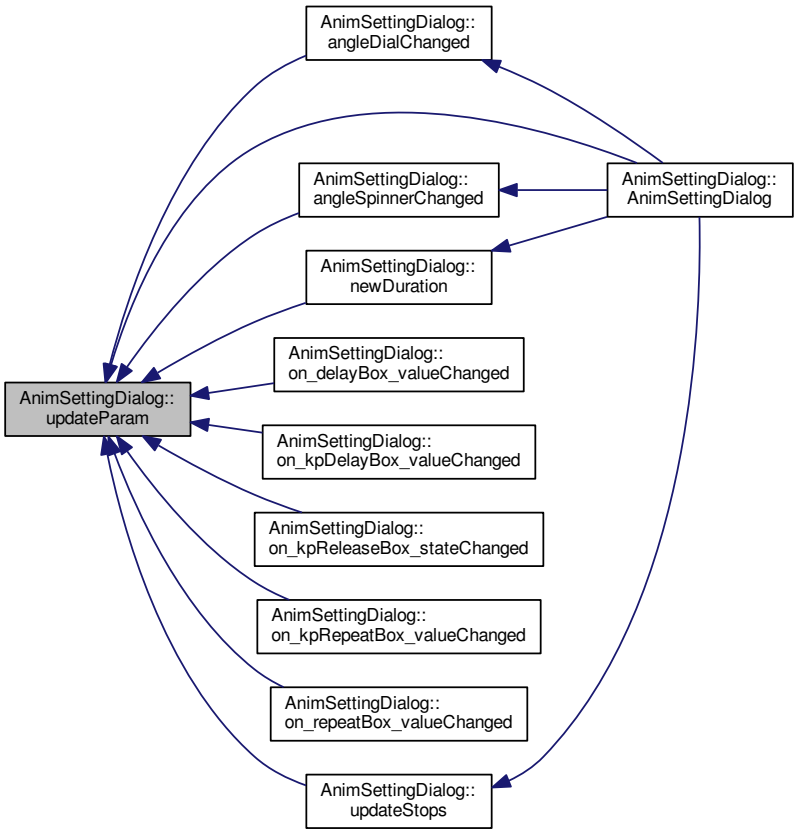
```
401                                                  {
402     if(!settingWidgets.contains(name))
403         return;
404     // stop and kpstop have defeat switches
405     if((name == "stop" && stopCheck->isChecked() == hasRepeat)
406             || (name == "kpstop" && kpStopCheck->isChecked() ==
    hasRepeat)){
407         _anim->parameter(name, -1);
408         return;
409     } else if(name == "kpmode"){
410         // kpmode uses a drop-down, and selection is inverted
411         QComboBox* widget = (QComboBox*)settingWidgets[name];
412         int selected = widget->currentIndex();
413         if(!_anim->script()->hasKeypress())
414             selected++;
415         _anim->parameter(name, selected);
416         return;
417     }
418     // Read value based on type
```

```
419     switch(_anim->script()->param(name).type){
420     case AnimScript::Param::BOOL:{
421         QCheckBox* widget = (QCheckBox*)settingWidgets[name];
422         _anim->parameter(name, (int)widget->isChecked());
423         break;
424     }
425     case AnimScript::Param::LONG:{
426         QSpinBox* widget = (QSpinBox*)settingWidgets[name];
427         _anim->parameter(name, widget->value());
428         break;
429     }
430     case AnimScript::Param::DOUBLE:{
431         QDoubleSpinBox* widget = (QDoubleSpinBox*)settingWidgets[
    name];
432         _anim->parameter(name, widget->value());
433         break;
434     }
435     case AnimScript::Param::RGB:{
436         ColorButton* widget = (ColorButton*)settingWidgets[
    name];
437         QColor color = widget->color();
438         char hex[7];
439         snprintf(hex, sizeof(hex), "%02x%02x%02x", color.red(), color.green(), color.blue());
440         _anim->parameter(name, QString(hex));
441         break;
442     }
443     case AnimScript::Param::ARGB:{
444         ColorButton* widget = (ColorButton*)settingWidgets[
    name];
445         QColor color = widget->color();
446         char hex[9];
447         snprintf(hex, sizeof(hex), "%02x%02x%02x%02x", color.alpha(), color.red(), color.green(), color.
    blue());
448         _anim->parameter(name, QString(hex));
449         break;
450     }
451     case AnimScript::Param::GRADIENT:
452     case AnimScript::Param::AGRADIENT:{
453         GradientButton* widget = (GradientButton*)
    settingWidgets[name];
454         _anim->parameter(name, widget->toString());
455         break;
456     }
457     case AnimScript::Param::ANGLE:{
458         QDial* widget = (QDial*)settingWidgets[name];
459         _anim->parameter(name, angleFlip(widget->value()));
460         break;
461     }
462     case AnimScript::Param::STRING:{
463         QLineEdit* widget = (QLineEdit*)settingWidgets[name];
464         _anim->parameter(name, widget->text());
465         break;
466     }
467     default:
468         break;
469     }
470 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.5.3.12 void AnimSettingDialog::updateStops ( ) [private], [slot]

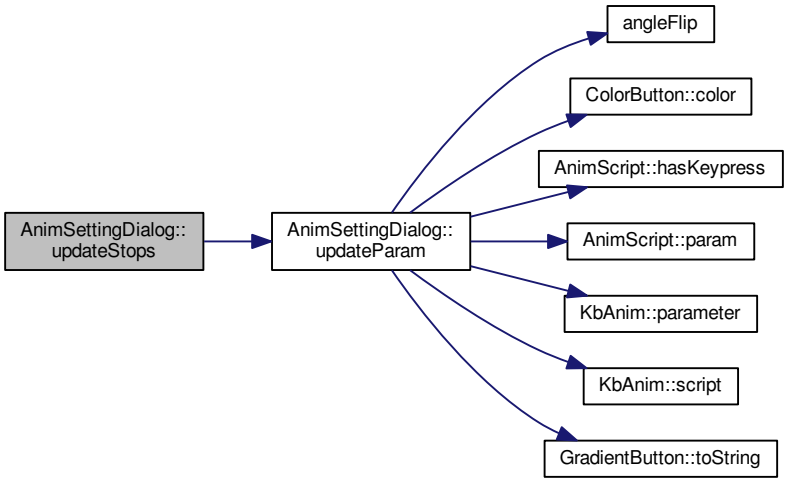
Definition at line 375 of file animsettingdialog.cpp.

References hasRepeat, kpStopCheck, settingWidgets, stopCheck, and updateParam().

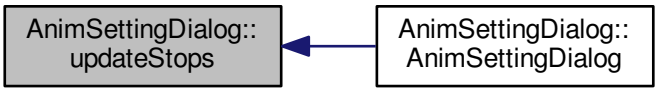
Referenced by AnimSettingDialog().

```
375                                  {
376     if(hasRepeat){
377         // When repeat is enabled, the check box is "repeat forever", so disable stop count if checked
378         settingWidgets.value("stop")->setDisabled(stopCheck->isChecked());
379         settingWidgets.value("kpstop")->setDisabled(kpStopCheck->isChecked());
380     } else {
381         // When repeat is disabled, the check box is "stop after", so disable stop count UNLESS checked
382         settingWidgets.value("stop")->setEnabled(stopCheck->isChecked());
383         settingWidgets.value("kpstop")->setEnabled(kpStopCheck->isChecked());
384     }
385     updateParam("stop");
386     updateParam("kpstop");
387 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.5.4 Field Documentation

12.5.4.1 KbAnim* AnimSettingDialog::_anim [private]

Definition at line 32 of file animsettingdialog.h.

Referenced by updateParam().

12.5.4.2 QSignalMapper AnimSettingDialog::angleDialMapper [private]

Definition at line 38 of file animsettingdialog.h.

Referenced by AnimSettingDialog().

12.5.4.3 QSignalMapper AnimSettingDialog::angleSpinnerMapper [private]

Definition at line 38 of file animsettingdialog.h.

Referenced by AnimSettingDialog().

12.5.4.4 QMap<QString, QSpinBox*> AnimSettingDialog::angleSpinners [private]

Definition at line 37 of file animsettingdialog.h.

Referenced by angleDialChanged(), angleSpinnerChanged(), and AnimSettingDialog().

12.5.4.5 bool AnimSettingDialog::hasRepeat [private]

Definition at line 30 of file animsettingdialog.h.

Referenced by AnimSettingDialog(), updateParam(), and updateStops().

12.5.4.6 QCheckBox * AnimSettingDialog::kpStopCheck [private]

Definition at line 29 of file animsettingdialog.h.

Referenced by AnimSettingDialog(), updateParam(), and updateStops().

12.5.4.7 double AnimSettingDialog::lastDuration [private]

Definition at line 33 of file animsettingdialog.h.

Referenced by AnimSettingDialog(), and newDuration().

12.5.4.8 QMap<QString, QWidget*> AnimSettingDialog::settingWidgets [private]

Definition at line 34 of file animsettingdialog.h.

Referenced by angleDialChanged(), angleSpinnerChanged(), AnimSettingDialog(), newDuration(), updateParam(), and updateStops().

12.5.4.9 QCheckBox* AnimSettingDialog::stopCheck [private]

Definition at line 29 of file animsettingdialog.h.

Referenced by AnimSettingDialog(), updateParam(), and updateStops().

12.5.4.10 Ui::AnimSettingDialog* AnimSettingDialog::ui [private]

Definition at line 28 of file animsettingdialog.h.

Referenced by AnimSettingDialog(), name(), and ~AnimSettingDialog().

12.5.4.11 QSignalMapper AnimSettingDialog::updateMapper [private]

Definition at line 35 of file animsettingdialog.h.

Referenced by AnimSettingDialog().

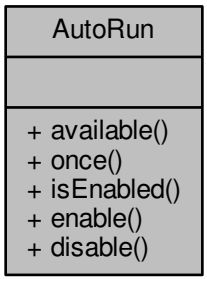
The documentation for this class was generated from the following files:

- src/ckb/animsettingdialog.h
- src/ckb/animsettingdialog.cpp

12.6 AutoRun Class Reference

`#include <src/ckb/autorun.h>`

Collaboration diagram for AutoRun:



Static Public Member Functions

- static bool available ()
- static bool once ()
- static bool isEnabled ()
- static void enable ()
- static void disable ()

12.6.1 Detailed Description

Definition at line 8 of file autorun.h.

12.6.2 Member Function Documentation

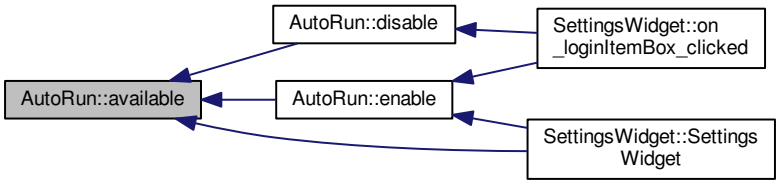
12.6.2.1 bool AutoRun::available ( ) [static]

Definition at line 20 of file autorun.cpp.

Referenced by disable(), enable(), and SettingsWidget::SettingsWidget().

```
20                      {
21     // Allow autostart if the program is located in a system path
22 #ifdef Q_OS_LINUX
23     return QDir::root().absoluteFilePath(QStandardPaths::findExecutable("ckb")) == qApp->
       applicationFilePath();
24 #elif defined(Q_OS_MACX)
25     return qApp->applicationFilePath().startsWith("/Applications/ckb.app", Qt::CaseInsensitive);
26 #endif
27 }
```

Here is the caller graph for this function:



12.6.2.2 void AutoRun::disable ( ) [static]

Definition at line 53 of file autorun.cpp.

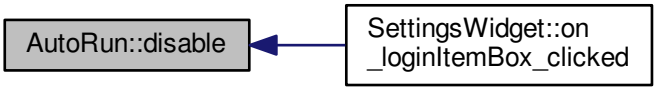
References available().

Referenced by SettingsWidget::on_loginItemBox_clicked().

```
53                      {
54     if(!available())
55         return;
56     // Remove file
57     QFile::remove(path.absoluteFilePath(file));
58 }
```

Here is the call graph for this function:

Here is the caller graph for this function:



12.6.2.3 void AutoRun::enable ( ) [static]

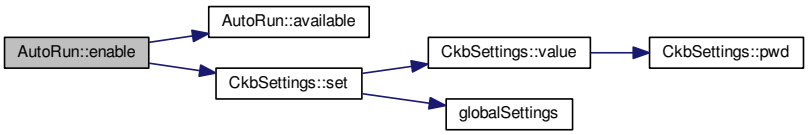
Definition at line 42 of file autorun.cpp.

References available(), and CkbSettings::set().

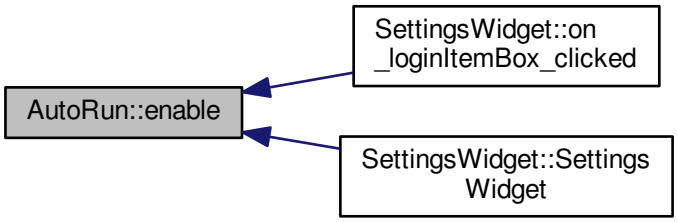
Referenced by SettingsWidget::on_loginItemBox_clicked(), and SettingsWidget::SettingsWidget().

```
42                      {
43     if(!available())
44         return;
45     // Copy file into place
46     if(!path.exists())
47         QDir::home().mkpath(path.absolutePath());
48     QFile::copy(internalFile, path.absoluteFilePath(file));
49     // Mark once() as done
50     CkbSettings::set(settingPath, true);
51 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.6.2.4 bool AutoRun::isEnabled ( ) [static]

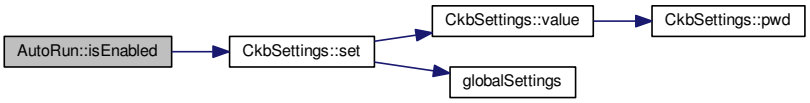
Definition at line 33 of file autorun.cpp.

References CkbSettings::set().

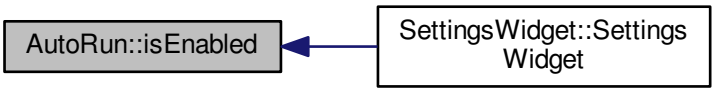
Referenced by SettingsWidget::SettingsWidget().

```
33                      {
34     // Check if the file exists. If not, autostart is disabled.
35     if(!path.exists() || !path.exists(file))
36         return false;
37     // If autostart is enabled, set the flag from once() (in case it hasn't been done yet)
38     CkbSettings::set(settingPath, true);
39     return true;
40 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.6.2.5 bool AutoRun::once ( ) [static]

Definition at line 29 of file autorun.cpp.

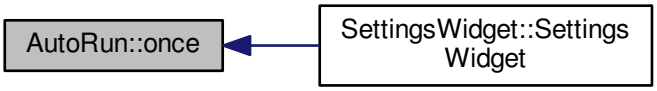
References CkbSettings::get().

Referenced by SettingsWidget::SettingsWidget().

```
29                  {
30      return CkbSettings::get(settingPath).toBool();
31  }
```

Here is the call graph for this function:

Here is the caller graph for this function:



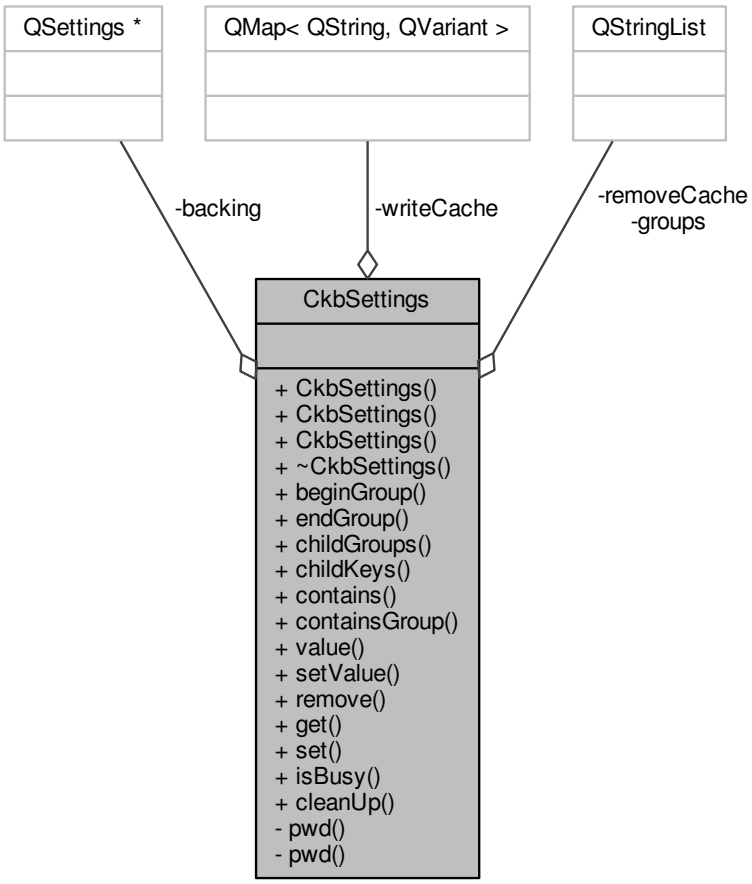
The documentation for this class was generated from the following files:

- src/ckb/autorun.h
- src/ckb/autorun.cpp

12.7 CkbSettings Class Reference

```
#include <src/ckb/ckbsettings.h>
```

Collaboration diagram for CkbSettings:



Public Member Functions

- CkbSettings ()
- CkbSettings (const QString &basePath, bool eraseExisting=false)
- CkbSettings (QSettings &settings)
- ~CkbSettings ()
- void beginGroup (const QString &prefix)
- void endGroup ()
- QStringList childGroups () const
- QStringList childKeys () const
- bool contains (const QString &key) const
- bool containsGroup (const QString &group)
- QVariant value (const QString &key, const QVariant &defaultValue=QVariant()) const
- void setValue (const QString &key, const QVariant &value)
- void remove (const QString &key)

Static Public Member Functions

- static QVariant get (const QString &key, const QVariant &defaultValue=QVariant())

- static void set (const QString &key, const QVariant &value)
- static bool isBusy ()
- static void cleanUp ()

Private Member Functions

- QString pwd () const
- QString pwd (const QString &key) const

Private Attributes

- QSettings * backing
- QStringList groups
- QStringList removeCache
- QMap< QString, QVariant > writeCache

12.7.1 Detailed Description

Definition at line 11 of file ckbsettings.h.

12.7.2 Constructor & Destructor Documentation

12.7.2.1 CkbSettings::CkbSettings ( )

Definition at line 61 of file ckbsettings.cpp.

```
61                          :
62     backing(globalSettings()) {
63 }
```

12.7.2.2 CkbSettings::CkbSettings (const QString & *basePath*, bool *eraseExisting* = `false`)

Definition at line 65 of file ckbsettings.cpp.

References beginGroup().

```
65                                                                  :
66     backing(globalSettings()) {
67     if(basePath.isEmpty()){
68         if(eraseExisting)
69             qDebug() << "CkbSettings created with basePath = \"\" and eraseExisting = true. This is a
       mistake.";
70         return;
71     }
72     if(eraseExisting)
73         remove(basePath);
74     beginGroup(basePath);
75 }
```

Here is the call graph for this function:

12.7.2.3 CkbSettings::CkbSettings (QSettings & *settings*)

Definition at line 77 of file ckbsettings.cpp.

```
77                                              :
78     backing(&settings) {
79 }
```

12.7.2.4 CkbSettings::~CkbSettings ( )

Definition at line 145 of file ckbsettings.cpp.

References backing, removeCache, and writeCache.

```
145                           {
146     if(removeCache.isEmpty() && writeCache.isEmpty())
147         return;
148     // Save the settings from the settings thread.
149     // They have to be saved from that thread specifically to avoid performance issues
150     CkbSettingsWriter* writer = new CkbSettingsWriter(
      backing, removeCache, writeCache);
151     writer->moveToThread(globalThread);
152     QObject::staticMetaObject.invokeMethod(writer, "run", Qt::QueuedConnection);
153 }
```

12.7.3 Member Function Documentation

12.7.3.1 void CkbSettings::beginGroup (const QString & *prefix*)

Definition at line 81 of file ckbsettings.cpp.

References groups.

Referenced by CkbSettings(), KbAnim::save(), and SGroup::SGroup().

```
81                                                      {
82     groups.append(prefix);
83 }
```

Here is the caller graph for this function:

12.7.3.2 QStringList CkbSettings::childGroups ( ) const

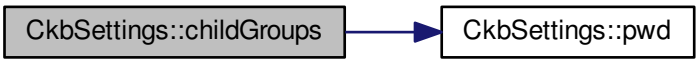
Definition at line 89 of file ckbsettings.cpp.

References backing, lockMutex, and pwd().

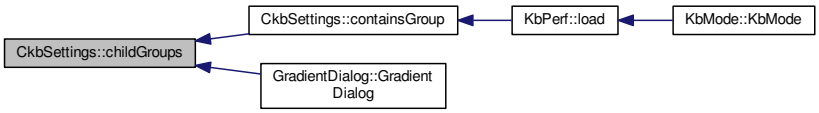
Referenced by containsGroup(), and GradientDialog::GradientDialog().

```
89                                              {
90     QString current = pwd();
91     lockMutex;
92     if(!current.isEmpty())
93         backing->beginGroup(current);
94     QStringList res = backing->childGroups();
95     if(!current.isEmpty())
96         backing->endGroup();
97     return res;
98 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.7.3.3 QStringList CkbSettings::childKeys ( ) const

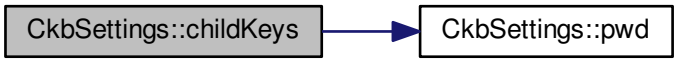
Definition at line 100 of file ckbsettings.cpp.

References backing, lockMutex, and pwd().

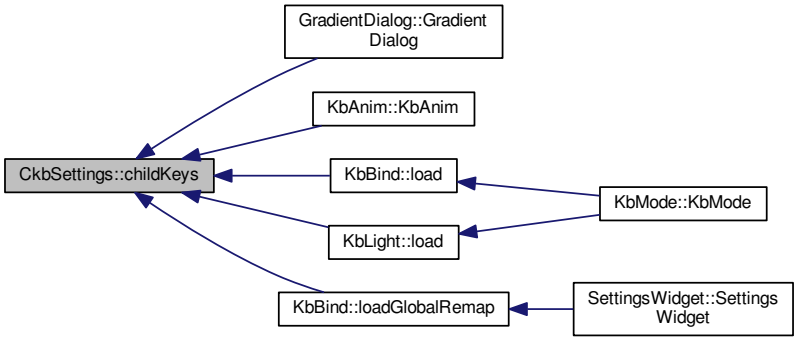
Referenced by GradientDialog::GradientDialog(), KbAnim::KbAnim(), KbBind::load(), KbLight::load(), and KbBind::loadGlobalRemap().

```
100                                         {
101     QString current = pwd();
102     lockMutex;
103     if(!current.isEmpty())
104         backing->beginGroup(current);
105     QStringList res = backing->childKeys();
106     if(!current.isEmpty())
107         backing->endGroup();
108     return res;
109 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.7.3.4 void CkbSettings::cleanUp ( ) [static]

Definition at line 46 of file ckbsettings.cpp.

References _globalSettings, cacheWritesInProgress(), and globalThread.

Referenced by MainWindow::cleanup().

```
46                          {
47     if(!_globalSettings)
48         return;
49     // Wait for all writers to finish
50     while(cacheWritesInProgress.load() > 0)
51         QThread::yieldCurrentThread();
52     // Stop thread and delete objects
53     globalThread->quit();
54     globalThread->wait();
55     delete globalThread;
56     delete _globalSettings;
57     globalThread = 0;
58     _globalSettings = 0;
59 }
```

Here is the call graph for this function:

Here is the caller graph for this function:

12.7.3.5 bool CkbSettings::contains (const QString & *key*) const

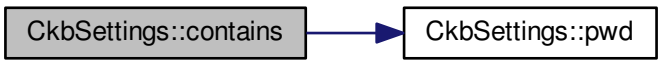
Definition at line 111 of file ckbsettings.cpp.

References backing, lockMutex, and pwd().

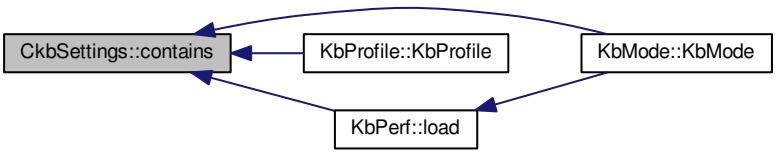
Referenced by KbMode::KbMode(), KbProfile::KbProfile(), and KbPerf::load().

```
111                                                 {
112     lockMutex;
113     return backing->contains(pwd(key));
114 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.7.3.6 bool CkbSettings::containsGroup (const QString & *group*)

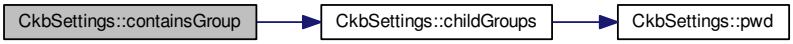
Definition at line 116 of file ckbsettings.cpp.

References childGroups().

Referenced by KbPerf::load().

```
116                                                 {
117     QStringList components = group.split("/");
118     if(components.length() > 1){
119         // Find sub-group
120         SGroup group(*this, components[0]);
121         return containsGroup(QStringList(components.mid(1)).join('/'));
122     }
123     return childGroups().contains(group);
124 }
```

Here is the call graph for this function:



Here is the caller graph for this function:

12.7.3.7 void CkbSettings::endGroup ( )

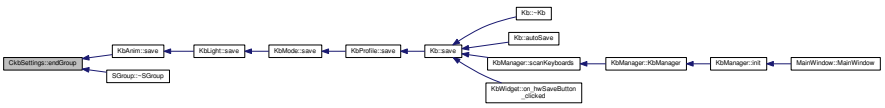
Definition at line 85 of file ckbsettings.cpp.

References groups.

Referenced by KbAnim::save(), and SGroup::~SGroup().

```
85                          {
86     groups.removeLast();
87 }
```

Here is the caller graph for this function:



12.7.3.8 QVariant CkbSettings::get (const QString & *key*, const QVariant & *defaultValue* = `QVariant()`) [static]

Definition at line 155 of file ckbsettings.cpp.

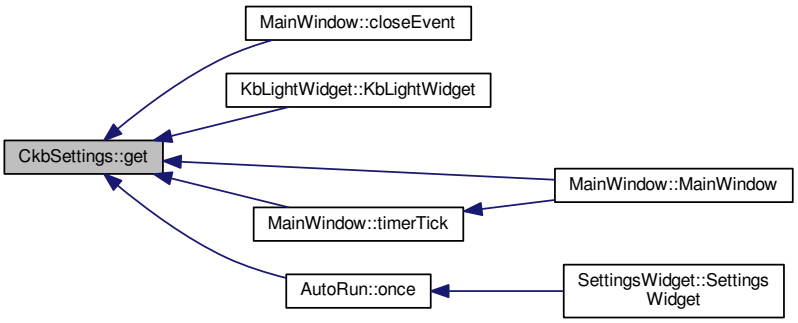
References globalCache, globalSettings(), lockMutexCache, and lockMutexStatic2.

Referenced by MainWindow::closeEvent(), KbLightWidget::KbLightWidget(), MainWindow::MainWindow(), AutoRun::once(), and MainWindow::timerTick().

```
155                                                                     {
156     // Store these settings in a memory cache
157     lockMutexCache;
158     if(globalCache.contains(key))
159         return globalCache.value(key);
160     // If it wasn't found in the memory cache, look for it in QSettings
161     lockMutexStatic2;
162     QSettings* settings = globalSettings();
163     return globalCache[key] = settings->value(key, defaultValue);
164 }
```

Here is the call graph for this function:

Here is the caller graph for this function:



12.7.3.9 bool CkbSettings::isBusy ( ) [static]

Definition at line 42 of file ckbsettings.cpp.

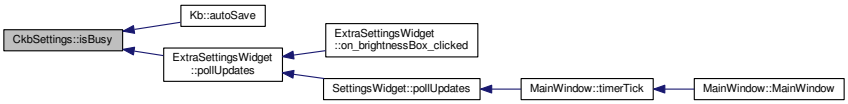
References cacheWritesInProgress().

Referenced by Kb::autoSave(), and ExtraSettingsWidget::pollUpdates().

```
42                         {
43     return cacheWritesInProgress.load() > 0;
44 }
```

Here is the call graph for this function:

Here is the caller graph for this function:



12.7.3.10 QString CkbSettings::pwd ( ) const [inline], [private]

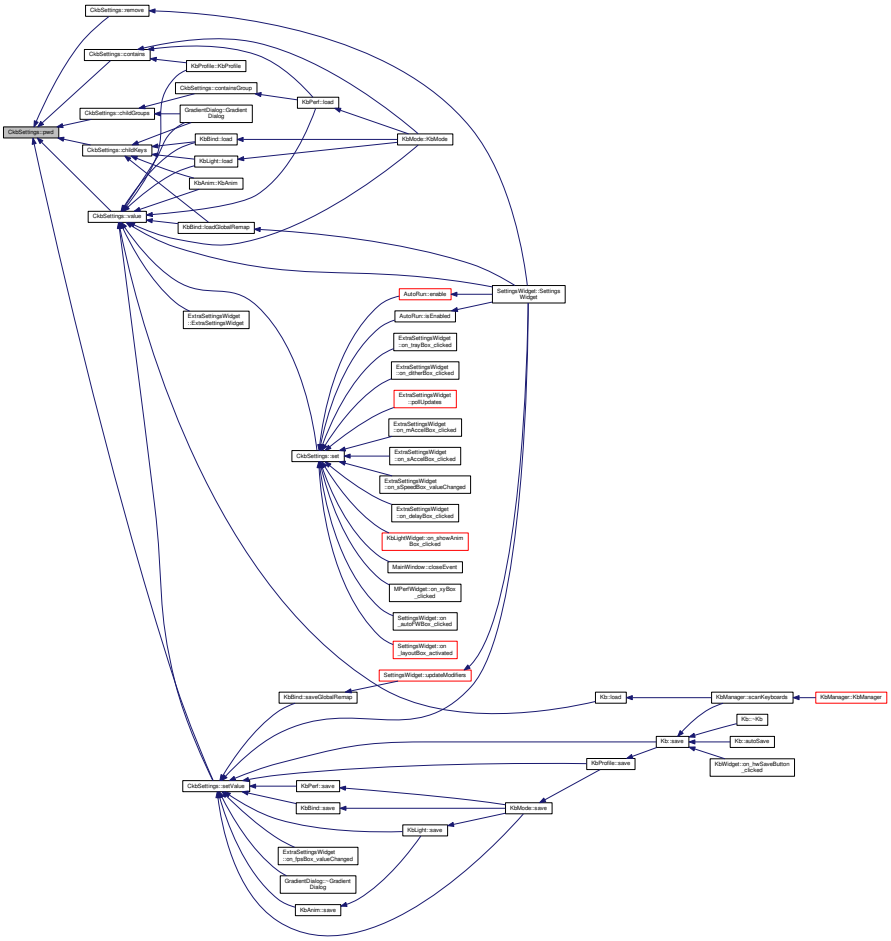
Definition at line 51 of file ckbsettings.h.

References groups.

Referenced by childGroups(), childKeys(), contains(), remove(), setValue(), and value().

```
51 { return groups.join("/"); }
```

Here is the caller graph for this function:



12.7.3.11 QString CkbSettings::pwd (const QString & *key*) const [inline], [private]

Definition at line 52 of file ckbsettings.h.

References groups, and pwd().

Referenced by pwd().

```
52 { return pwd() + (groups.isEmpty() ? "" : "/") + key; }
```

Here is the call graph for this function:

Here is the caller graph for this function:

12.7.3.12 void CkbSettings::remove (const QString & *key*)

Definition at line 141 of file ckbsettings.cpp.

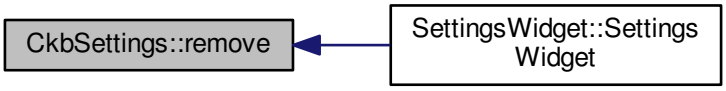
References pwd(), and removeCache.

Referenced by SettingsWidget::SettingsWidget().

```
141                                          {
142     removeCache.append(pwd(key));
143 }
```

Here is the call graph for this function:

Here is the caller graph for this function:



12.7.3.13 void CkbSettings::set (const QString & *key*, const QVariant & *value*) [static]

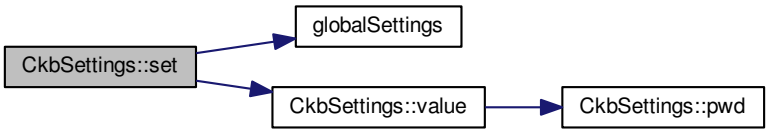
Definition at line 166 of file ckbsettings.cpp.

References globalCache, globalSettings(), lockMutexCache, lockMutexStatic, and value().

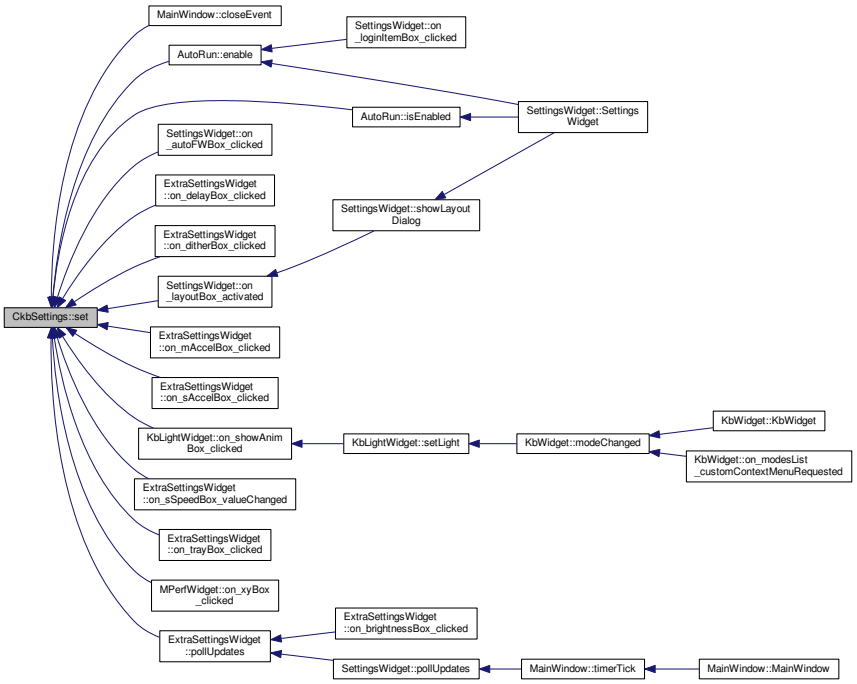
Referenced by MainWindow::closeEvent(), AutoRun::enable(), AutoRun::isEnabled(), SettingsWidget::on_autoFWBox_clicked(), ExtraSettingsWidget::on_delayBox_clicked(), ExtraSettingsWidget::on_ditherBox_clicked(), SettingsWidget::on_layoutBox_activated(), ExtraSettingsWidget::on_mAccelBox_clicked(), ExtraSettingsWidget::on_sAccelBox_clicked(), KbLightWidget::on_showAnimBox_clicked(), ExtraSettingsWidget::on_sSpeedBox_valueChanged(), ExtraSettingsWidget::on_trayBox_clicked(), MPerfWidget::on_xyBox_clicked(), and ExtraSettingsWidget::pollUpdates().

```
166                                                                 {
167     {
168         lockMutexCache;
169         if(globalCache.value(key) == value)
170             return;
171         globalCache[key] = value;
172     }
173     lockMutexStatic;
174     globalSettings()->setValue(key, value);
175 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.7.3.14 void CkbSettings::setValue (const QString & *key,* const QVariant & *value*)

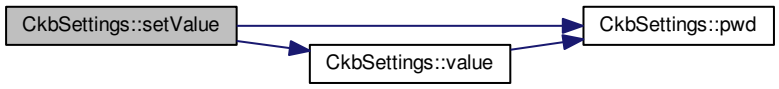
Definition at line 131 of file ckbsettings.cpp.

References globalCache, lockMutexCache, pwd(), value(), and writeCache.

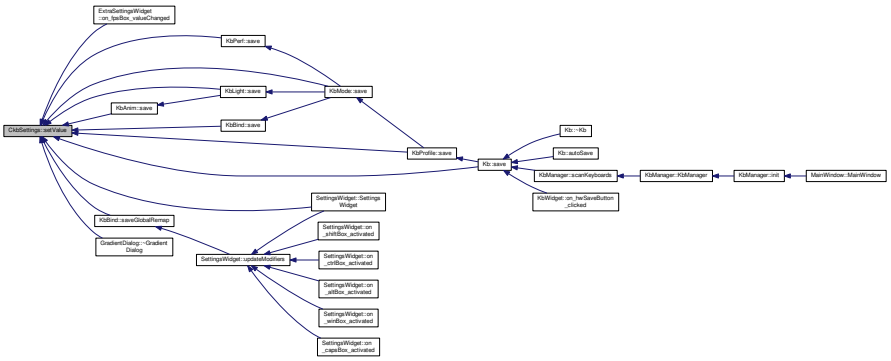
Referenced by ExtraSettingsWidget::on_fpsBox_valueChanged(), KbProfile::save(), KbPerf::save(), KbAnim::save(), KbBind::save(), KbMode::save(), KbLight::save(), Kb::save(), KbBind::saveGlobalRemap(), SettingsWidget::SettingsWidget(), and GradientDialog::~GradientDialog().

```
131                                                                 {
132     // Cache the write values, save them when the object is destroyed
133     QString realKey = pwd(key);
134     writeCache[realKey] = value;
135     // Update global cache if needed
136     lockMutexCache;
137     if(globalCache.contains(realKey))
138         globalCache[realKey] = value;
139 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.7.3.15 QVariant CkbSettings::value (const QString & *key,* const QVariant & *defaultValue* = `QVariant()`) const

Definition at line 126 of file ckbsettings.cpp.

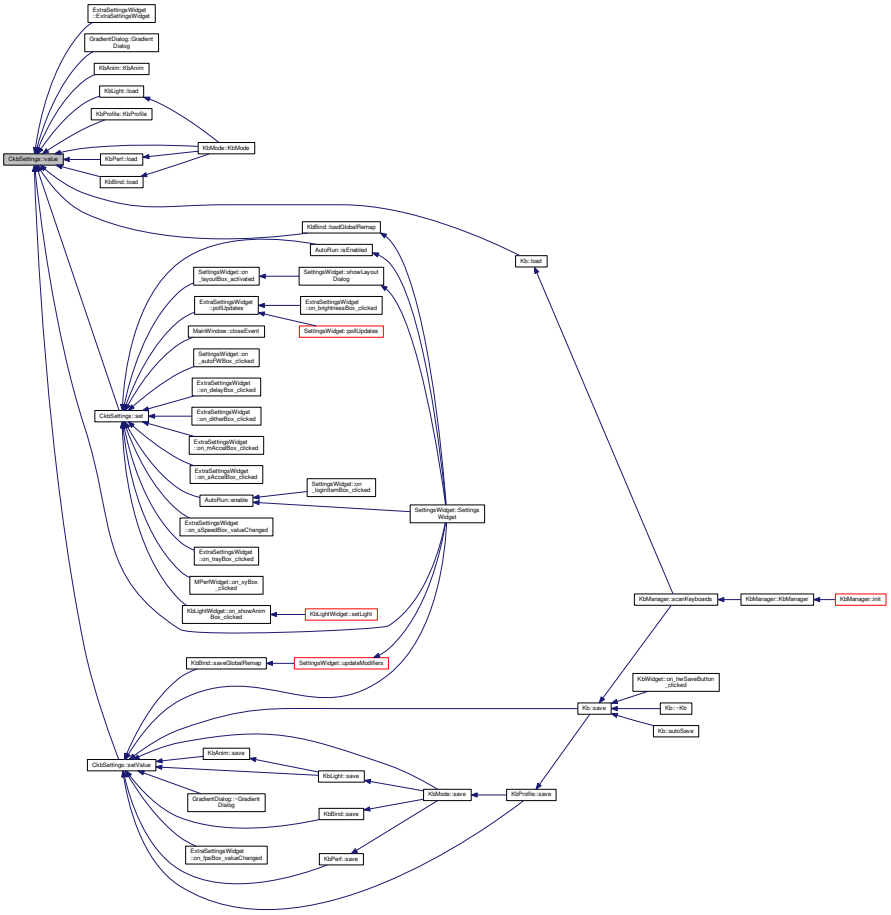
References backing, lockMutex, and pwd().

Referenced by ExtraSettingsWidget::ExtraSettingsWidget(), GradientDialog::GradientDialog(), KbAnim::KbAnim(), KbMode::KbMode(), KbProfile::KbProfile(), KbPerf::load(), KbBind::load(), KbLight::load(), Kb::load(), KbBind::loadGlobalRemap(), set(), SettingsWidget::SettingsWidget(), and setValue().

```
126                                                                 {
127     lockMutex;
128     return backing->value(pwd(key), defaultValue);
129 }
```

Here is the call graph for this function:

Here is the caller graph for this function:



12.7.4 Field Documentation

12.7.4.1 QSettings* CkbSettings::backing [private]

Definition at line 46 of file ckbsettings.h.

Referenced by childGroups(), childKeys(), contains(), value(), and ~CkbSettings().

12.7.4.2 QStringList CkbSettings::groups [private]

Definition at line 47 of file ckbsettings.h.

Referenced by beginGroup(), endGroup(), and pwd().

12.7.4.3 QStringList CkbSettings::removeCache [private]

Definition at line 48 of file ckbsettings.h.

Referenced by remove(), and ~CkbSettings().

12.7.4.4 QMap<QString, QVariant> CkbSettings::writeCache [private]

Definition at line 49 of file ckbsettings.h.

Referenced by setValue(), and ~CkbSettings().

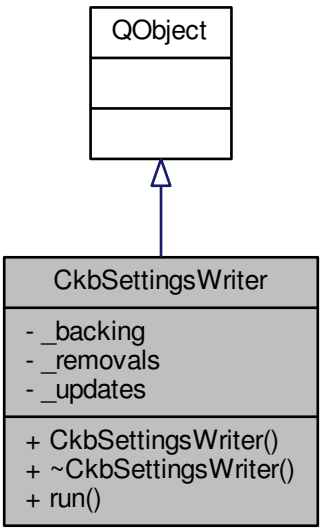
The documentation for this class was generated from the following files:

- src/ckb/ckbsettings.h
- src/ckb/ckbsettings.cpp

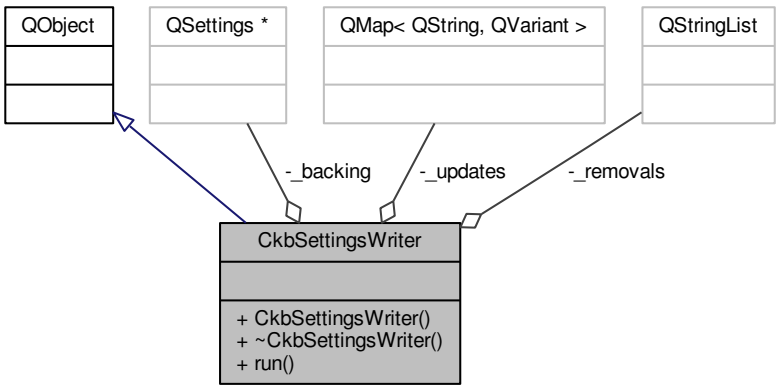
12.8 CkbSettingsWriter Class Reference

`#include <src/ckb/ckbsettingswriter.h>`

Inheritance diagram for CkbSettingsWriter:



Collaboration diagram for CkbSettingsWriter:



Public Member Functions

- CkbSettingsWriter (QSettings ∗backing, const QStringList &removals, const QMap< QString, QVariant > &updates)
- ∼CkbSettingsWriter ()
- Q_SLOT void run ()

Private Attributes

- QSettings ∗ _backing
- QStringList _removals
- QMap< QString, QVariant > _updates

12.8.1 Detailed Description

Definition at line 12 of file ckbsettingswriter.h.

12.8.2 Constructor & Destructor Documentation

12.8.2.1 CkbSettingsWriter::CkbSettingsWriter (QSettings ∗ *backing*, const QStringList & *removals*, const QMap< QString, QVariant > & *updates*)

Definition at line 12 of file ckbsettingswriter.cpp.

References cacheWritesInProgress.

```
12
                    :
13     _backing(backing), _removals(removals), _updates(updates) {
14     cacheWritesInProgress.ref();
15 }
```

12.8.2.2 CkbSettingsWriter::~CkbSettingsWriter ( )

Definition at line 17 of file ckbsettingswriter.cpp.

References cacheWritesInProgress.

```
17                                      {
18     cacheWritesInProgress.deref();
19 }
```

12.8.3 Member Function Documentation

12.8.3.1 void CkbSettingsWriter::run ( )

Definition at line 21 of file ckbsettingswriter.cpp.

References _backing, _removals, _updates, and lockMutexStatic.

```
21                                {
22     lockMutexStatic;
23     // Process key removals
24     foreach(const QString& rm, _removals){
25         _backing->remove(rm);
26     }
27     // Process writes
28     QMapIterator<QString, QVariant> i(_updates);
29     while(i.hasNext()){
30         i.next();
31         _backing->setValue(i.key(), i.value());
32         // Updating the global cache was done above
33     }
34     _backing->sync();
35     deleteLater();
36 }
```

12.8.4 Field Documentation

12.8.4.1 QSettings* CkbSettingsWriter::_backing [private]

Definition at line 21 of file ckbsettingswriter.h.

Referenced by run().

12.8.4.2 QStringList CkbSettingsWriter::_removals [private]

Definition at line 22 of file ckbsettingswriter.h.

Referenced by run().

12.8.4.3 QMap<QString, QVariant> CkbSettingsWriter::_updates [private]

Definition at line 23 of file ckbsettingswriter.h.

Referenced by run().

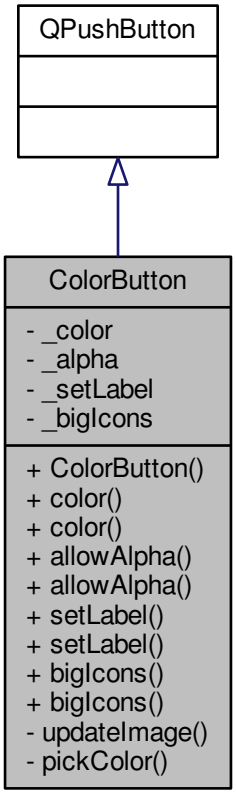
The documentation for this class was generated from the following files:

- src/ckb/ckbsettingswriter.h
- src/ckb/ckbsettingswriter.cpp

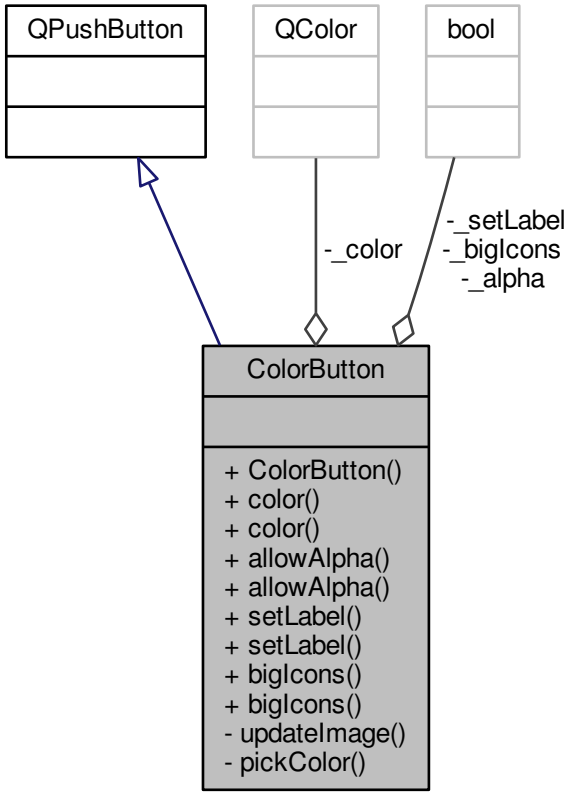
12.9 ColorButton Class Reference

`#include <src/ckb/colorbutton.h>`

Inheritance diagram for ColorButton:



Collaboration diagram for ColorButton:



Signals

- void colorChanged (QColor newColor)

Public Member Functions

- ColorButton (QWidget *parent=0, bool allowAlpha=false)
- const QColor & color () const
- void color (const QColor &newColor)
- bool allowAlpha () const
- void allowAlpha (bool newAllowAlpha)
- bool setLabel () const
- void setLabel (bool newSetLabel)
- bool bigIcons () const
- void bigIcons (bool newBigIcons)

Private Slots

- void pickColor ()

Private Member Functions

- void updateImage ()

Private Attributes

- QColor _color
- bool _alpha
- bool _setLabel
- bool _bigIcons

12.9.1 Detailed Description

Definition at line 6 of file colorbutton.h.

12.9.2 Constructor & Destructor Documentation

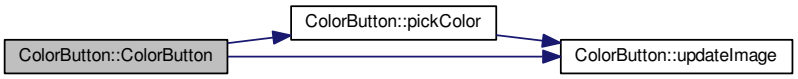
12.9.2.1 ColorButton::ColorButton (QWidget * *parent* = 0, bool *allowAlpha* = false) [explicit]

Definition at line 5 of file colorbutton.cpp.

References pickColor(), and updateImage().

```
5                                                       :
6     QPushButton(parent), _alpha(allowAlpha),
      _setLabel(true), _bigIcons(false)
7 {
8     setAutoDefault(false);
9     setDefault(false);
10     updateImage();
11     // Pick color on click (use queued connection so that any on_*_clicked() events can be processed first)
12     connect(this, SIGNAL(clicked()), this, SLOT(pickColor()), Qt::QueuedConnection);
13 }
```

Here is the call graph for this function:



12.9.3 Member Function Documentation

12.9.3.1 bool ColorButton::allowAlpha ( ) const [inline]

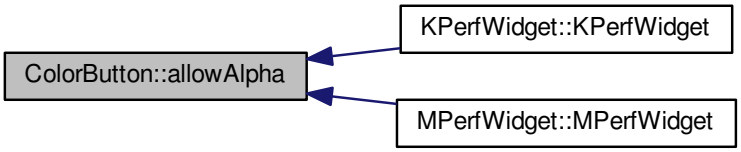
Definition at line 17 of file colorbutton.h.

References _alpha.

Referenced by KPerfWidget::KPerfWidget(), and MPerfWidget::MPerfWidget().

```
17 { return _alpha; }
```

Here is the caller graph for this function:



12.9.3.2 void ColorButton::allowAlpha (bool *newAllowAlpha*)

Definition at line 22 of file colorbutton.cpp.

References _alpha, and updateImage().

```
22                                              {
23     _alpha = newAllowAlpha;
24     updateImage();
25 }
```

Here is the call graph for this function:

12.9.3.3 bool ColorButton::bigIcons ( ) const [inline]

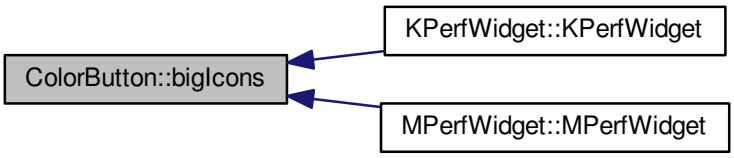
Definition at line 23 of file colorbutton.h.

References _bigIcons.

Referenced by KPerfWidget::KPerfWidget(), and MPerfWidget::MPerfWidget().

```
23 { return _bigIcons; }
```

Here is the caller graph for this function:



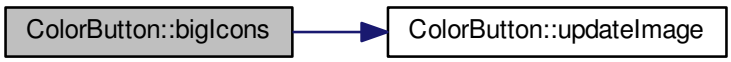
12.9.3.4 void ColorButton::bigIcons (bool *newBigIcons*)

Definition at line 27 of file colorbutton.cpp.

References _bigIcons, and updateImage().

```
27                                          {
28     _bigIcons = newBigIcons;
29     updateImage();
30 }
```

Here is the call graph for this function:



12.9.3.5 const QColor& ColorButton::color ( ) const [inline]

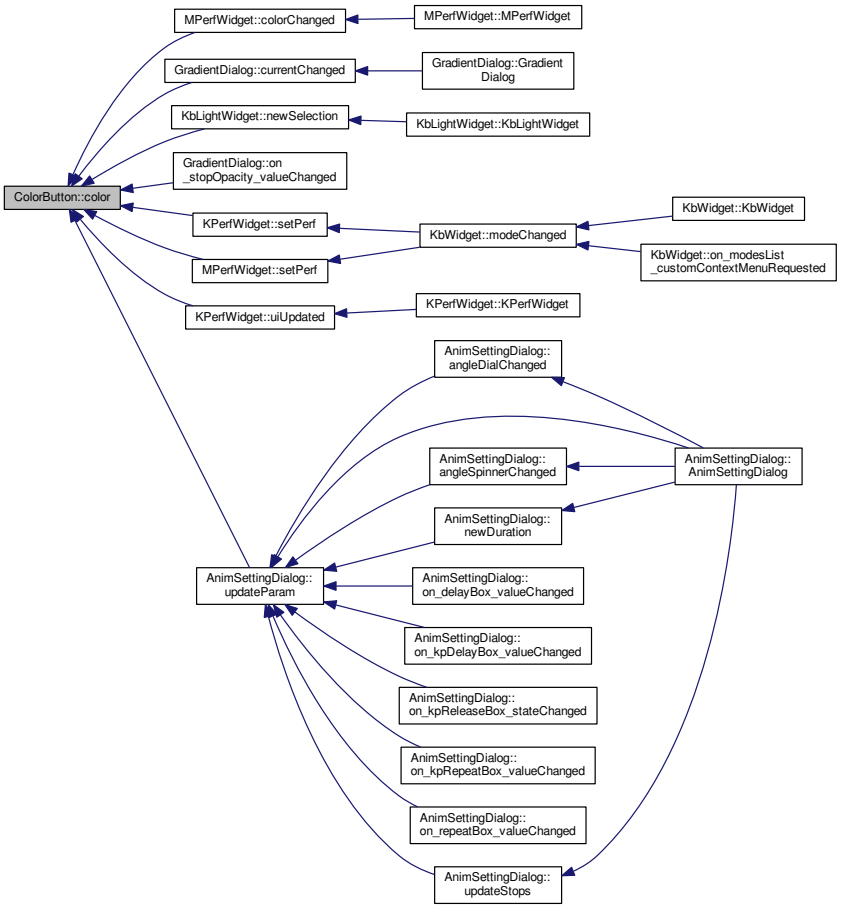
Definition at line 13 of file colorbutton.h.

References _color.

Referenced by MPerfWidget::colorChanged(), GradientDialog::currentChanged(), KbLightWidget::newSelection(), GradientDialog::on_stopOpacity_valueChanged(), KPerfWidget::setPerf(), MPerfWidget::setPerf(), KPerfWidget::uiUpdated(), and AnimSettingDialog::updateParam().

```
13 { return _color; }
```

Here is the caller graph for this function:



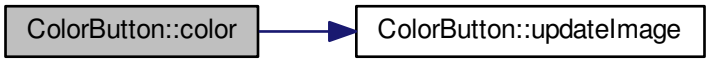
12.9.3.6 void ColorButton::color (const QColor & *newColor*)

Definition at line 15 of file colorbutton.cpp.

References _alpha, _color, and updateImage().

```
15                                          {
16     _color = newColor;
17     if(!_alpha)
18         _color.setAlpha(255);
19     updateImage();
20 }
```

Here is the call graph for this function:



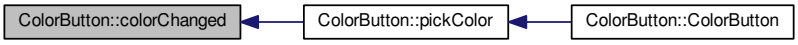
12.9.3.7 void ColorButton::colorChanged (QColor *newColor*) [signal]

Definition at line 132 of file moc_colorbutton.cpp.

Referenced by pickColor().

```
133 {
134     void *_a[] = { Q_NULLPTR, const_cast<void*>(reinterpret_cast<const void*>(&_t1)) };
135     QMetaObject::activate(this, &staticMetaObject, 0, _a);
136 }
```

Here is the caller graph for this function:



12.9.3.8 void ColorButton::pickColor ( ) [private], [slot]

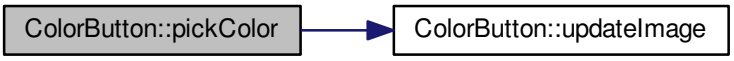
Definition at line 60 of file colorbutton.cpp.

References _alpha, _color, colorChanged(), and updateImage().

Referenced by ColorButton().

```
60                            {
61     QColor newColor = QColorDialog::getColor(_color, this, QString(),
      QColorDialog::ColorDialogOptions(_alpha ? QColorDialog::ShowAlphaChannel : 0));
62     if(newColor.isValid()){
63         _color = newColor;
64         updateImage();
65         emit colorChanged(_color);
66     }
67 }
```

Here is the call graph for this function:



Here is the caller graph for this function:

12.9.3.9 bool ColorButton::setLabel ( ) const [inline]

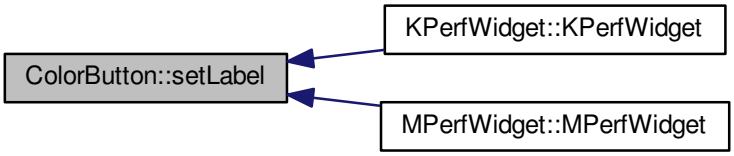
Definition at line 20 of file colorbutton.h.

References _setLabel.

Referenced by KPerfWidget::KPerfWidget(), and MPerfWidget::MPerfWidget().

```
20 { return _setLabel; }
```

Here is the caller graph for this function:



12.9.3.10 void ColorButton::setLabel (bool *newSetLabel*) [inline]

Definition at line 21 of file colorbutton.h.

References _setLabel.

```
21 { _setLabel = newSetLabel; }
```

12.9.3.11 void ColorButton::updateImage ( ) [private]

Definition at line 32 of file colorbutton.cpp.

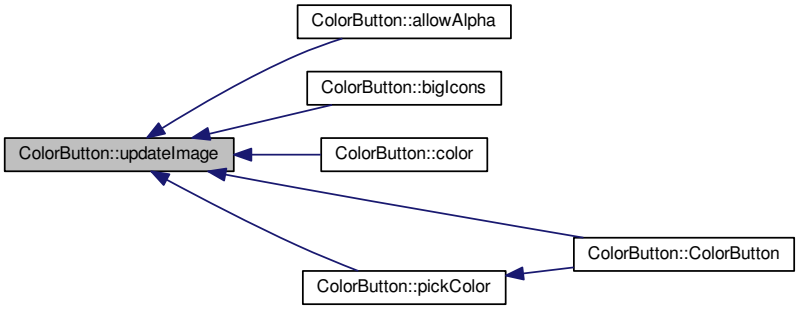
References _alpha, _bigIcons, _color, and _setLabel.

Referenced by allowAlpha(), bigIcons(), color(), ColorButton(), and pickColor().

```
32                              {
33     if(!_color.isValid()){
34         setIcon(QIcon());
35         if(_setLabel)
36             setText("Change color...");
37         return;
38     }
39     const int w = 24, h = _bigIcons ? 24 : 12;
40     QImage image(w, h, QImage::Format_RGB888);
41     QPainter painter(&image);
42     painter.setPen(Qt::NoPen);
43     painter.fillRect(0, 0, w, h, QColor(0, 0, 0));
44     if(_alpha && _color.alpha() != 255){
45         painter.fillRect(1, 1, w / 2 - 1, h / 2 - 1, QColor(255, 255, 255));
46         painter.fillRect(w / 2, 1, w / 2 - 1, h / 2 - 1, QColor(192, 192, 192));
47         painter.fillRect(1, h / 2, w / 2 - 1, h / 2 - 1, QColor(192, 192, 192));
48         painter.fillRect(w / 2, h / 2, w / 2 - 1, h / 2 - 1, QColor(255, 255, 255));
49     }
50     painter.fillRect(1, 1, w - 2, h - 2, _color);
51     setIcon(QIcon(QPixmap::fromImage(image)));
52     if(_setLabel){
53         QString text = QString(" (%1, %2, %3)").arg(_color.red()).arg(
       _color.green()).arg(_color.blue());
54         if(_alpha)
```

```
55             text += QString(", %4%").arg(QString::number(_color.alphaF() * 100., 'f', 0));
56         setText(text);
57     }
58 }
```

Here is the caller graph for this function:



12.9.4 Field Documentation

12.9.4.1 bool ColorButton::_alpha [private]

Definition at line 34 of file colorbutton.h.

Referenced by allowAlpha(), color(), pickColor(), and updateImage().

12.9.4.2 bool ColorButton::_bigIcons [private]

Definition at line 34 of file colorbutton.h.

Referenced by bigIcons(), and updateImage().

12.9.4.3 QColor ColorButton::_color [private]

Definition at line 33 of file colorbutton.h.

Referenced by color(), pickColor(), and updateImage().

12.9.4.4 bool ColorButton::_setLabel [private]

Definition at line 34 of file colorbutton.h.

Referenced by setLabel(), and updateImage().

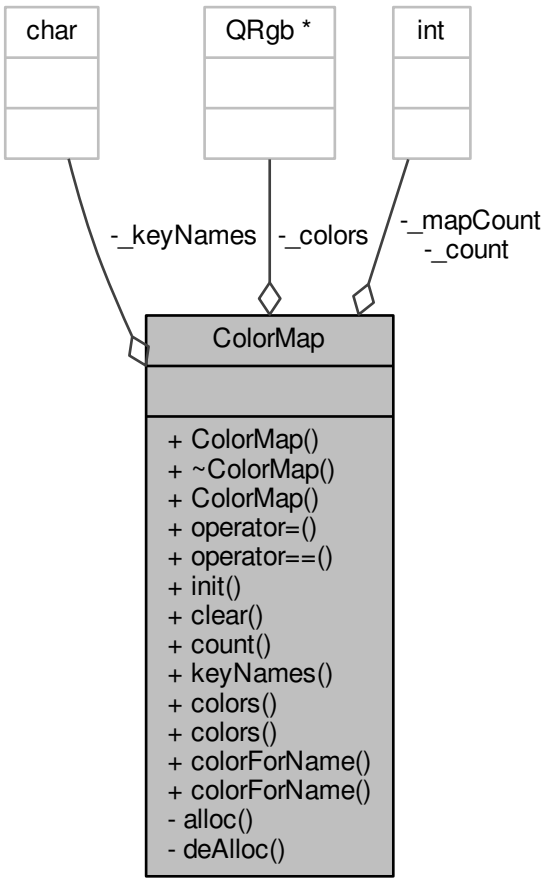
The documentation for this class was generated from the following files:

- src/ckb/colorbutton.h
- src/ckb/colorbutton.cpp
- src/ckb/release/moc_colorbutton.cpp

12.10 ColorMap Class Reference

`#include <src/ckb/colormap.h>`

Collaboration diagram for ColorMap:



Public Member Functions

- ColorMap ()
- ~ColorMap ()
- ColorMap (const ColorMap &rhs)
- const ColorMap & operator= (const ColorMap &rhs)
- bool operator== (const ColorMap &rhs) const
- void init (const KeyMap &map)
- void clear ()
- int count () const
- const char *const * keyNames () const
- QRgb * colors ()
- const QRgb * colors () const
- QRgb * colorForName (const char *name)
- const QRgb * colorForName (const char *name) const

Private Member Functions

- void alloc (int count)
- void deAlloc ()

Private Attributes

- const char ** _keyNames
- QRgb * _colors
- int _count
- int _mapCount

12.10.1 Detailed Description

Definition at line 15 of file colormap.h.

12.10.2 Constructor & Destructor Documentation

12.10.2.1 ColorMap::ColorMap ( )

Definition at line 3 of file colormap.cpp.

```
3                   :
4     _keyNames(0), _colors(0), _count(0), _mapCount(0)
5 {
6 }
```

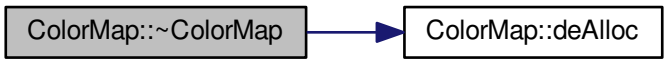
12.10.2.2 ColorMap::~ColorMap ( )

Definition at line 8 of file colormap.cpp.

References deAlloc().

```
8                   {
9     deAlloc();
10 }
```

Here is the call graph for this function:



12.10.2.3 ColorMap::ColorMap (const ColorMap & *rhs*)

Definition at line 12 of file colormap.cpp.

```
12                                  :
13     _keyNames(0), _colors(0), _count(0), _mapCount(0)
14 {
15     *this = rhs;
16 }
```

12.10.3 Member Function Documentation

12.10.3.1 void ColorMap::alloc (int *count*) [private]

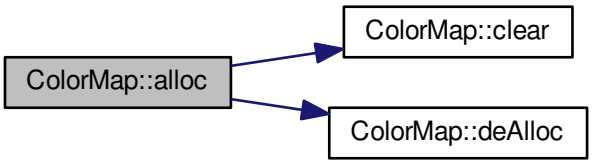
Definition at line 43 of file colormap.cpp.

References _colors, _count, _keyNames, _mapCount, clear(), and deAlloc().

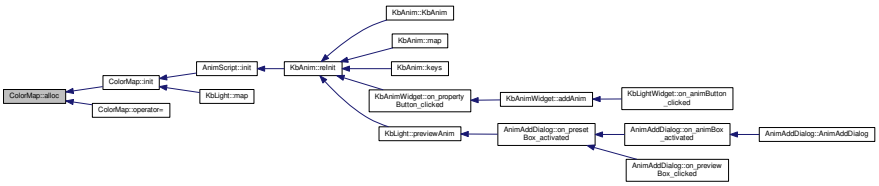
Referenced by init(), and operator=().

```
43                                {
44     if(newCount > _mapCount){
45         // ^ map never shrinks, only expands
46         deAlloc();
47         _mapCount = newCount;
48         _keyNames = new const char*[_mapCount];
49         _colors = new QRgb[_mapCount];
50     }
51     _count = newCount;
52     clear();
53 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.10.3.2 void ColorMap::clear ( )

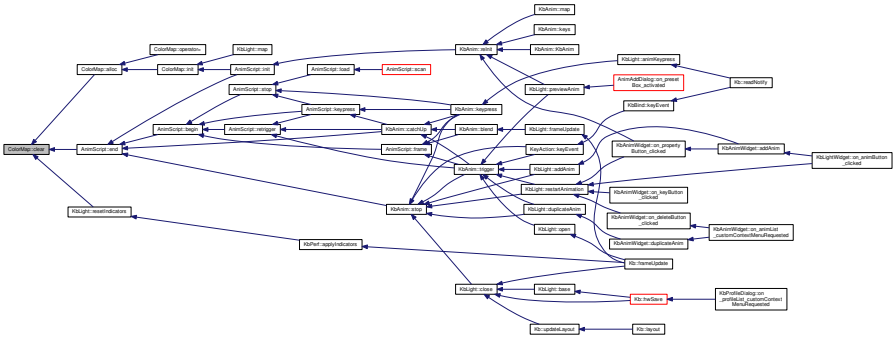
Definition at line 61 of file colormap.cpp.

References _colors, and _count.

Referenced by alloc(), AnimScript::end(), and KbLight::resetIndicators().

```
61                      {
62     memset(_colors, 0, _count * sizeof(QRgb));
63 }
```

Here is the caller graph for this function:



12.10.3.3 QRgb * ColorMap::colorForName (const char * *name*)

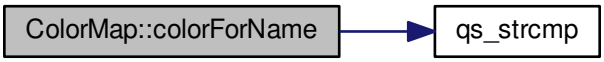
Definition at line 88 of file colormap.cpp.

References _colors, _count, _keyNames, and qs_strcmp().

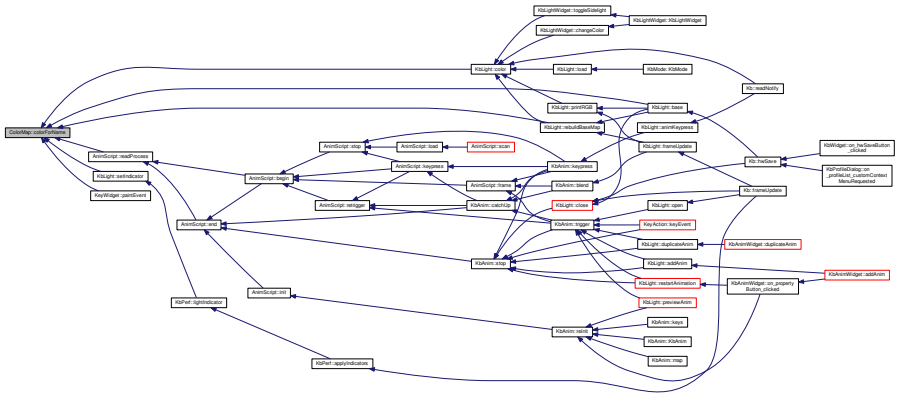
Referenced by KbLight::base(), KbLight::color(), KeyWidget::paintEvent(), AnimScript::readProcess(), KbLight::rebuildBaseMap(), and KbLight::setIndicator().

```
88                                          {
89     if(!_keyNames)
90         return 0;
91     const char** namePtr = (const char**)bsearch(&name, _keyNames,
      _count, sizeof(const char*), qs_strcmp);
92     if(!namePtr)
93         return 0;
94     ptrdiff_t position = namePtr - _keyNames;
95     return _colors + position;
96 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



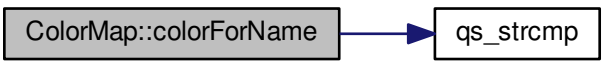
12.10.3.4 const QRgb * ColorMap::colorForName (const char * *name*) const

Definition at line 98 of file colormap.cpp.

References _colors, _count, _keyNames, and qs_strcmp().

```
98                                                          {
99      if(!_keyNames)
100         return 0;
101     const char** namePtr = (const char**)bsearch(&name, _keyNames,
    _count, sizeof(const char*), qs_strcmp);
102     if(!namePtr)
103         return 0;
104     ptrdiff_t position = namePtr - _keyNames;
105     return _colors + position;
106 }
```

Here is the call graph for this function:



12.10.3.5 QRgb* ColorMap::colors ( ) [inline]

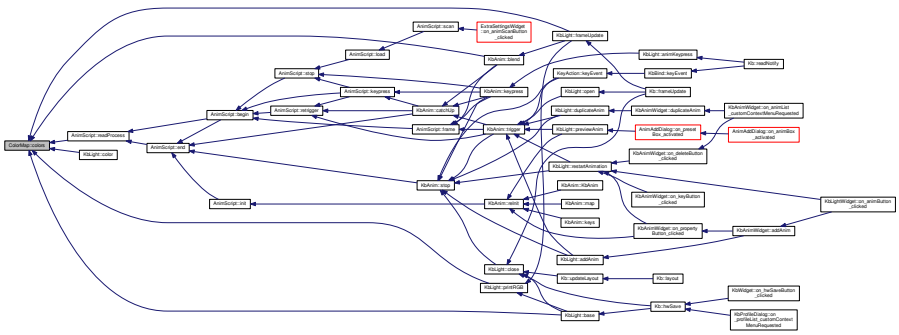
Definition at line 33 of file colormap.h.

References _colors.

Referenced by KbLight::base(), KbAnim::blend(), KbLight::color(), KbLight::frameUpdate(), KbLight::printRGB(), and AnimScript::readProcess().

```
33 { return _colors; }
```

Here is the caller graph for this function:



12.10.3.6 const QRgb* ColorMap::colors ( ) const [inline]

Definition at line 34 of file colormap.h.

References _colors.

```
34 { return _colors; }
```

12.10.3.7 int ColorMap::count ( ) const [inline]

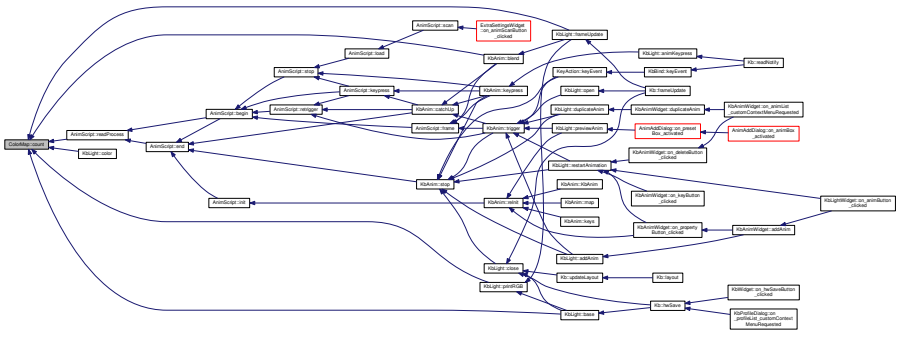
Definition at line 31 of file colormap.h.

References _count.

Referenced by KbLight::base(), KbAnim::blend(), KbLight::color(), KbLight::frameUpdate(), KbLight::printRGB(), and AnimScript::readProcess().

```
31 { return _count; }
```

Here is the caller graph for this function:



12.10.3.8 void ColorMap::deAlloc ( ) [private]

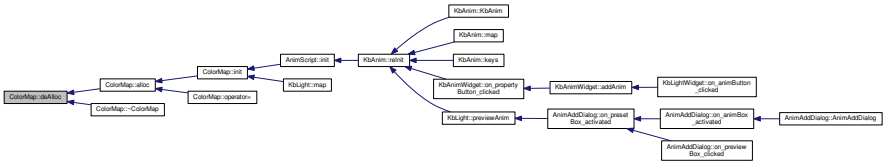
Definition at line 55 of file colormap.cpp.

References _colors, _count, _keyNames, and _mapCount.

Referenced by alloc(), and ~ColorMap().

```
55                      {
56     _count = _mapCount = 0;
57     if(_keyNames) delete[] _keyNames;
58     if(_colors) delete[] _colors;
59 }
```

Here is the caller graph for this function:



12.10.3.9 void ColorMap::init (const KeyMap & *map*)

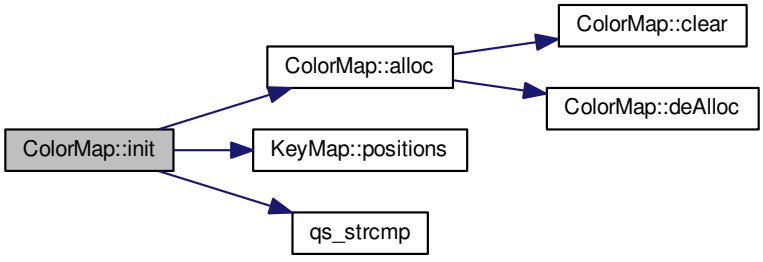
Definition at line 69 of file colormap.cpp.

References _count, _keyNames, alloc(), Key::hasLed, Key::name, KeyMap::positions(), and qs_strcmp().

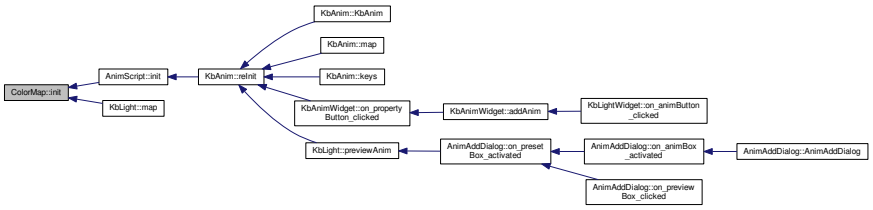
Referenced by AnimScript::init(), and KbLight::map().

```
69                                      {
70     QList<Key> newKeys = map.positions();
71     // There's no point including keys that don't have LEDs, so remove them now
72     QMutableListIterator<Key> i(newKeys);
73     while(i.hasNext()){
74         Key key = i.next();
75         if(!key.hasLed)
76             i.remove();
77     }
78     // Now that we know how many keys we'll have, check memory
79     alloc(newKeys.count());
80     // Copy key names
81     int keyPos = 0;
82     foreach(const Key& key, newKeys)
83         _keyNames[keyPos++] = key.name;      // as above, it's safe to copy these since the
       strings are constants
84     // Sort names for quick access
85     qsort(_keyNames, _count, sizeof(const char*), qs_strcmp);
86 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.10.3.10 const char* const* ColorMap::keyNames ( ) const [inline]

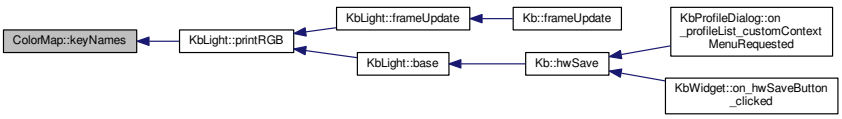
Definition at line 32 of file colormap.h.

References _keyNames.

Referenced by KbLight::printRGB().

```
32 { return _keyNames; }
```

Here is the caller graph for this function:



12.10.3.11 const ColorMap & ColorMap::operator= (const ColorMap & *rhs*)

Definition at line 18 of file colormap.cpp.

References _colors, _count, _keyNames, and alloc().

```
18                                                  {
19     alloc(rhs._count);
20     // Copy key names and colors
21     // (Note: it's safe to copy the const char*'s because key names are constants and will never move or be
       deleted)
22     memcpy(_keyNames, rhs._keyNames, sizeof(const char*) *
      _count);
23     memcpy(_colors, rhs._colors, sizeof(QRgb) * _count);
24     return rhs;
25 }
```

Here is the call graph for this function:

12.10.3.12 bool ColorMap::operator== (const ColorMap & *rhs*) const

Definition at line 27 of file colormap.cpp.

References _colors, _count, and _keyNames.

```
27                                                  {
28     if(_count != rhs._count)
29         return false;
30
31     // Compare just the pointers because key names are sorted and the strings are constants.
32     if(memcmp(_keyNames, rhs._keyNames, sizeof(const char*) *
       _count) != 0)
33         return false;
34
35     for(int x = 0;x < _count;x++){
36         if(_colors[x] != rhs._colors[x])
37             return false;
38     }
39
40     return true;
41 }
```

12.10.4 Field Documentation

12.10.4.1 QRgb* ColorMap::_colors [private]

Definition at line 45 of file colormap.h.

Referenced by alloc(), clear(), colorForName(), colors(), deAlloc(), operator=(), and operator==().

12.10.4.2 int ColorMap::_count [private]

Definition at line 46 of file colormap.h.

Referenced by alloc(), clear(), colorForName(), count(), deAlloc(), init(), operator=(), and operator==().

12.10.4.3 const char** ColorMap::_keyNames [private]

Definition at line 44 of file colormap.h.

Referenced by alloc(), colorForName(), deAlloc(), init(), keyNames(), operator=(), and operator==().

12.10.4.4 int ColorMap::_mapCount [private]

Definition at line 46 of file colormap.h.

Referenced by alloc(), and deAlloc().

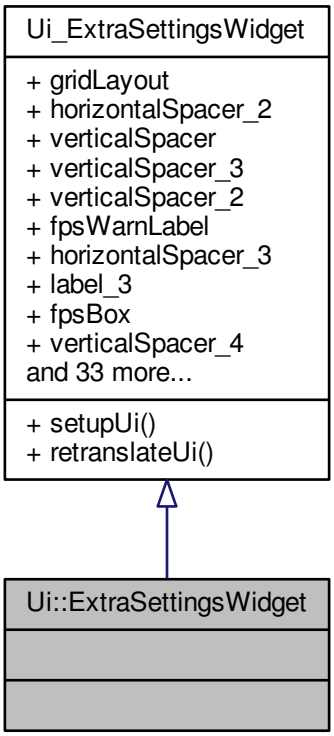
The documentation for this class was generated from the following files:

- src/ckb/colormap.h
- src/ckb/colormap.cpp

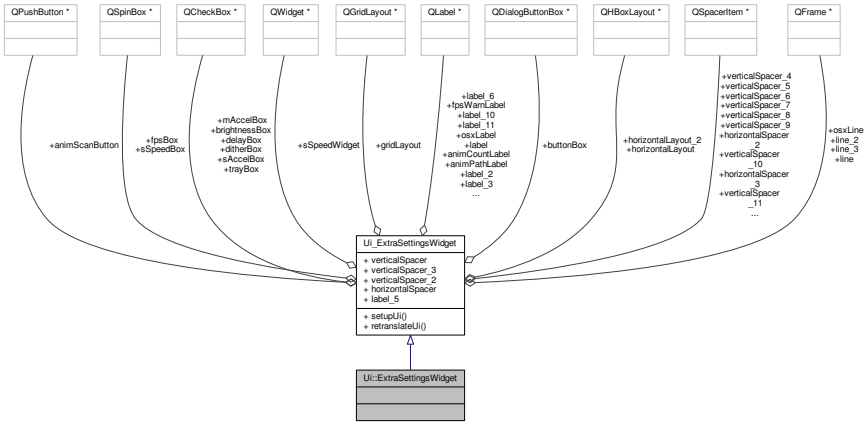
12.11 Ui::ExtraSettingsWidget Class Reference

`#include <src/ckb/ui_extrasettingswidget.h>`

Inheritance diagram for Ui::ExtraSettingsWidget:



Collaboration diagram for Ui::ExtraSettingsWidget:



Public Member Functions

- void setupUi (QDialog *ExtraSettingsWidget)
- void retranslateUi (QDialog *ExtraSettingsWidget)

Data Fields

- QGridLayout * gridLayout
- QSpacerItem * horizontalSpacer_2
- QSpacerItem * verticalSpacer
- QSpacerItem * verticalSpacer_3
- QSpacerItem * verticalSpacer_2
- QLabel * fpsWarnLabel
- QSpacerItem * horizontalSpacer_3
- QLabel * label_3
- QSpinBox * fpsBox
- QSpacerItem * verticalSpacer_4
- QSpacerItem * verticalSpacer_9
- QLabel * label_6
- QSpacerItem * verticalSpacer_6
- QSpacerItem * verticalSpacer_5
- QCheckBox * sAccelBox
- QSpacerItem * verticalSpacer_7
- QSpacerItem * verticalSpacer_8
- QSpacerItem * verticalSpacer_10
- QSpacerItem * verticalSpacer_11
- QSpacerItem * horizontalSpacer
- QFrame * line_2
- QHBoxLayout * horizontalLayout
- QLabel * animPathLabel
- QLabel * animCountLabel
- QPushButton * animScanButton
- QCheckBox * brightnessBox
- QCheckBox * trayBox
- QLabel * label

- QFrame * line
- QLabel * label_5
- QFrame * line_3
- QLabel * label_2
- QCheckBox * ditherBox
- QCheckBox * delayBox
- QLabel * osxLabel
- QFrame * osxLine
- QCheckBox * mAccelBox
- QWidget * sSpeedWidget
- QHBoxLayout * horizontalLayout_2
- QLabel * label_10
- QSpinBox * sSpeedBox
- QLabel * label_11
- QDialogButtonBox * buttonBox

12.11.1 Detailed Description

Definition at line 374 of file ui_extrasettingswidget.h.

12.11.2 Member Function Documentation

12.11.2.1 void Ui_ExtraSettingsWidget::retranslateUi (QDialog * *ExtraSettingsWidget*) [inline], [inherited]

Definition at line 330 of file ui_extrasettingswidget.h.

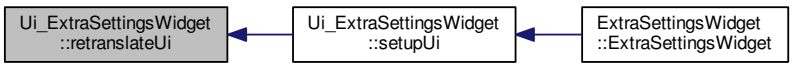
References Ui_ExtraSettingsWidget::animCountLabel, Ui_ExtraSettingsWidget::animPathLabel, Ui_ExtraSettingsWidget::animScanButton, Ui_ExtraSettingsWidget::brightnessBox, Ui_ExtraSettingsWidget::delayBox, Ui_ExtraSettingsWidget::ditherBox, Ui_ExtraSettingsWidget::fpsWarnLabel, Ui_ExtraSettingsWidget::label, Ui_ExtraSettingsWidget::label_10, Ui_ExtraSettingsWidget::label_11, Ui_ExtraSettingsWidget::label_2, Ui_ExtraSettingsWidget::label_3, Ui_ExtraSettingsWidget::label_5, Ui_ExtraSettingsWidget::label_6, Ui_ExtraSettingsWidget::mAccelBox, Ui_ExtraSettingsWidget::osxLabel, Ui_ExtraSettingsWidget::sAccelBox, and Ui_ExtraSettingsWidget::trayBox.

Referenced by Ui_ExtraSettingsWidget::setupUi().

```
331     {
332         ExtraSettingsWidget->setWindowTitle(QApplication::translate("ExtraSettingsWidget", "Application
       Settings", Q_NULLPTR));
333         fpsWarnLabel->setText(QApplication::translate("ExtraSettingsWidget", "Warning: high
       frame rates may cause stability issues", Q_NULLPTR));
334         label_3->setText(QApplication::translate("ExtraSettingsWidget", "Frame rate (FPS):",
      Q_NULLPTR));
335         label_6->setText(QApplication::translate("ExtraSettingsWidget", "Location:", Q_NULLPTR));
336 #ifndef QT_NO_TOOLTIP
337         sAccelBox->setToolTip(QApplication::translate("ExtraSettingsWidget", "Try this if you're
       having problems with the scroll wheel.", Q_NULLPTR));
338 #endif // QT_NO_TOOLTIP
339         sAccelBox->setText(QApplication::translate("ExtraSettingsWidget", "Disable scroll
       acceleration", Q_NULLPTR));
340         animPathLabel->setText(QApplication::translate("ExtraSettingsWidget", "
      /path/to/animations", Q_NULLPTR));
341         animCountLabel->setText(QApplication::translate("ExtraSettingsWidget", "0 animations
       found", Q_NULLPTR));
342         animScanButton->setText(QApplication::translate("ExtraSettingsWidget", "Re-scan",
      Q_NULLPTR));
343 #ifndef QT_NO_TOOLTIP
344         brightnessBox->setToolTip(QApplication::translate("ExtraSettingsWidget", "By default,
       the same brightness level will be applied to all profiles and all devices. Enable this to store it with the
       lighting mode instead.", Q_NULLPTR));
345 #endif // QT_NO_TOOLTIP
346         brightnessBox->setText(QApplication::translate("ExtraSettingsWidget", "Set brightness
       per-mode", Q_NULLPTR));
347 #ifndef QT_NO_TOOLTIP
348         trayBox->setToolTip(QApplication::translate("ExtraSettingsWidget", "The tray icon will not
       be displayed. The application will still run in the background; re-launch the app to see the GUI again.",
```

```
    Q_NULLPTR));
349 #endif // QT_NO_TOOLTIP
350         trayBox->setText(QApplication::translate("ExtraSettingsWidget", "Hide tray icon", Q_NULLPTR)
    );
351         label->setText(QApplication::translate("ExtraSettingsWidget", "Behavior", Q_NULLPTR));
352         label_5->setText(QApplication::translate("ExtraSettingsWidget", "Animation scripts",
    Q_NULLPTR));
353         label_2->setText(QApplication::translate("ExtraSettingsWidget", "Hardware", Q_NULLPTR));
354 #ifndef QT_NO_TOOLTIP
355         ditherBox->setToolTip(QApplication::translate("ExtraSettingsWidget", "May improve
     appearance on some keyboards.", Q_NULLPTR));
356 #endif // QT_NO_TOOLTIP
357         ditherBox->setText(QApplication::translate("ExtraSettingsWidget", "Use spatial dithering
     to simulate extra color resolution", Q_NULLPTR));
358 #ifndef QT_NO_TOOLTIP
359         delayBox->setToolTip(QApplication::translate("ExtraSettingsWidget", "When using macros with
     strings longer than 25 chars, some OS may lose characters (e.g. Mint 17.2). Select to prevent that bug.",
    Q_NULLPTR));
360 #endif // QT_NO_TOOLTIP
361         delayBox->setText(QApplication::translate("ExtraSettingsWidget", "Use delay for very long
     macros", Q_NULLPTR));
362         osxLabel->setText(QApplication::translate("ExtraSettingsWidget", "OS X tweaks", Q_NULLPTR))
    ;
363 #ifndef QT_NO_TOOLTIP
364         mAccelBox->setToolTip(QApplication::translate("ExtraSettingsWidget", "Try this if you're
     having problems with mouse movement.", Q_NULLPTR));
365 #endif // QT_NO_TOOLTIP
366         mAccelBox->setText(QApplication::translate("ExtraSettingsWidget", "Disable mouse
     acceleration", Q_NULLPTR));
367         label_10->setText(QApplication::translate("ExtraSettingsWidget", "Scroll", Q_NULLPTR));
368         label_11->setText(QApplication::translate("ExtraSettingsWidget", "line(s) at a time",
    Q_NULLPTR));
369     } // retranslateUi
```

Here is the caller graph for this function:



12.11.2.2 void Ui_ExtraSettingsWidget::setupUi (QDialog * *ExtraSettingsWidget*) [inline], [inherited]

Definition at line 78 of file ui_extrasettingswidget.h.

References Ui_ExtraSettingsWidget::animCountLabel, Ui_ExtraSettingsWidget::animPathLabel, Ui_ExtraSettingsWidget::animScanButton, Ui_ExtraSettingsWidget::brightnessBox, Ui_ExtraSettingsWidget::buttonBox, Ui_ExtraSettingsWidget::delayBox, Ui_ExtraSettingsWidget::ditherBox, Ui_ExtraSettingsWidget::fpsBox, Ui_ExtraSettingsWidget::fpsWarnLabel, Ui_ExtraSettingsWidget::gridLayout, Ui_ExtraSettingsWidget::horizontalLayout, Ui_ExtraSettingsWidget::horizontalLayout_2, Ui_ExtraSettingsWidget::horizontalSpacer, Ui_ExtraSettingsWidget::horizontalSpacer_2, Ui_ExtraSettingsWidget::horizontalSpacer_3, Ui_ExtraSettingsWidget::label, Ui_ExtraSettingsWidget::label_10, Ui_ExtraSettingsWidget::label_11, Ui_ExtraSettingsWidget::label_2, Ui_ExtraSettingsWidget::label_3, Ui_ExtraSettingsWidget::label_5, Ui_ExtraSettingsWidget::label_6, Ui_ExtraSettingsWidget::line, Ui_ExtraSettingsWidget::line_2, Ui_ExtraSettingsWidget::line_3, Ui_ExtraSettingsWidget::mAccelBox, Ui_ExtraSettingsWidget::osxLabel, Ui_ExtraSettingsWidget::osxLine, Ui_ExtraSettingsWidget::retranslateUi(), Ui_ExtraSettingsWidget::sAccelBox, Ui_ExtraSettingsWidget::sSpeedBox, Ui_ExtraSettingsWidget::sSpeedWidget, Ui_ExtraSettingsWidget::trayBox, Ui_ExtraSettingsWidget::verticalSpacer, Ui_ExtraSettingsWidget::verticalSpacer_10, Ui_ExtraSettingsWidget::verticalSpacer_11, Ui_ExtraSettingsWidget::verticalSpacer_2, Ui_ExtraSettingsWidget::verticalSpacer_3, Ui_ExtraSettingsWidget::verticalSpacer_4, Ui_ExtraSettingsWidget::verticalSpacer_5, Ui_ExtraSettingsWidget::verticalSpacer_6, Ui_ExtraSettingsWidget::verticalSpacer_7, Ui_ExtraSettingsWidget::verticalSpacer_8, and Ui_ExtraSettingsWidget::verticalSpacer_9.

Referenced by ExtraSettingsWidget::ExtraSettingsWidget().

```
79     {
```

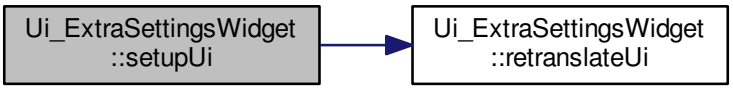
```
80          if (ExtraSettingsWidget->objectName().isEmpty())
81              ExtraSettingsWidget->setObjectName(QStringLiteral("ExtraSettingsWidget"));
82          ExtraSettingsWidget->resize(650, 515);
83          gridLayout = new QGridLayout(ExtraSettingsWidget);
84          gridLayout->setObjectName(QStringLiteral("gridLayout"));
85          horizontalSpacer_2 = new QSpacerItem(30, 20, QSizePolicy::Fixed,
    QSizePolicy::Minimum);
86
87          gridLayout->addItem(horizontalSpacer_2, 12, 2, 1, 1);
88
89          verticalSpacer = new QSpacerItem(0, 28, QSizePolicy::Minimum, QSizePolicy::Fixed);
90
91          gridLayout->addItem(verticalSpacer, 2, 0, 1, 1);
92
93          verticalSpacer_3 = new QSpacerItem(0, 28, QSizePolicy::Minimum, QSizePolicy::Fixed)
    ;
94
95          gridLayout->addItem(verticalSpacer_3, 3, 0, 1, 1);
96
97          verticalSpacer_2 = new QSpacerItem(20, 40, QSizePolicy::Minimum,
    QSizePolicy::Expanding);
98
99          gridLayout->addItem(verticalSpacer_2, 19, 5, 1, 1);
100
101          fpsWarnLabel = new QLabel(ExtraSettingsWidget);
102          fpsWarnLabel->setObjectName(QStringLiteral("fpsWarnLabel"));
103          QSizePolicy sizePolicy(QSizePolicy::Expanding, QSizePolicy::Preferred);
104          sizePolicy.setHorizontalStretch(0);
105          sizePolicy.setVerticalStretch(0);
106          sizePolicy.setHeightForWidth(fpsWarnLabel->sizePolicy().hasHeightForWidth());
107          fpsWarnLabel->setSizePolicy(sizePolicy);
108
109          gridLayout->addWidget(fpsWarnLabel, 12, 5, 1, 1);
110
111          horizontalSpacer_3 = new QSpacerItem(10, 20, QSizePolicy::Fixed,
    QSizePolicy::Minimum);
112
113          gridLayout->addItem(horizontalSpacer_3, 12, 4, 1, 1);
114
115          label_3 = new QLabel(ExtraSettingsWidget);
116          label_3->setObjectName(QStringLiteral("label_3"));
117
118          gridLayout->addWidget(label_3, 12, 1, 1, 1);
119
120          fpsBox = new QSpinBox(ExtraSettingsWidget);
121          fpsBox->setObjectName(QStringLiteral("fpsBox"));
122          fpsBox->setMinimum(5);
123          fpsBox->setMaximum(60);
124          fpsBox->setValue(30);
125
126          gridLayout->addWidget(fpsBox, 12, 3, 1, 1);
127
128          verticalSpacer_4 = new QSpacerItem(0, 28, QSizePolicy::Minimum, QSizePolicy::Fixed)
    ;
129
130          gridLayout->addItem(verticalSpacer_4, 7, 0, 1, 1);
131
132          verticalSpacer_9 = new QSpacerItem(0, 28, QSizePolicy::Minimum, QSizePolicy::Fixed)
    ;
133
134          gridLayout->addItem(verticalSpacer_9, 12, 0, 1, 1);
135
136          label_6 = new QLabel(ExtraSettingsWidget);
137          label_6->setObjectName(QStringLiteral("label_6"));
138
139          gridLayout->addWidget(label_6, 7, 1, 1, 1);
140
141          verticalSpacer_6 = new QSpacerItem(20, 10, QSizePolicy::Minimum, QSizePolicy::Fixed
    );
142
143          gridLayout->addItem(verticalSpacer_6, 8, 1, 1, 1);
144
145          verticalSpacer_5 = new QSpacerItem(20, 10, QSizePolicy::Minimum, QSizePolicy::Fixed
    );
146
147          gridLayout->addItem(verticalSpacer_5, 4, 1, 1, 1);
148
149          sAccelBox = new QCheckBox(ExtraSettingsWidget);
150          sAccelBox->setObjectName(QStringLiteral("sAccelBox"));
151
152          gridLayout->addWidget(sAccelBox, 18, 1, 1, 4);
153
154          verticalSpacer_7 = new QSpacerItem(20, 10, QSizePolicy::Minimum, QSizePolicy::Fixed
    );
155
156          gridLayout->addItem(verticalSpacer_7, 14, 1, 1, 1);
157
```

```
158         verticalSpacer_8 = new QSpacerItem(0, 28, QSizePolicy::Minimum, QSizePolicy::Fixed)
    ;
159
160         gridLayout->addItem(verticalSpacer_8, 13, 0, 1, 1);
161
162         verticalSpacer_10 = new QSpacerItem(0, 28, QSizePolicy::Minimum,
    QSizePolicy::Fixed);
163
164         gridLayout->addItem(verticalSpacer_10, 17, 0, 1, 1);
165
166         verticalSpacer_11 = new QSpacerItem(0, 28, QSizePolicy::Minimum,
    QSizePolicy::Fixed);
167
168         gridLayout->addItem(verticalSpacer_11, 18, 0, 1, 1);
169
170         horizontalSpacer = new QSpacerItem(40, 20, QSizePolicy::Expanding,
    QSizePolicy::Minimum);
171
172         gridLayout->addItem(horizontalSpacer, 12, 6, 1, 1);
173
174         line_2 = new QFrame(ExtraSettingsWidget);
175         line_2->setObjectName(QStringLiteral("line_2"));
176         line_2->setFrameShape(QFrame::HLine);
177         line_2->setFrameShadow(QFrame::Sunken);
178
179         gridLayout->addWidget(line_2, 10, 0, 1, 7);
180
181         horizontalLayout = new QHBoxLayout();
182         horizontalLayout->setObjectName(QStringLiteral("horizontalLayout"));
183         animPathLabel = new QLabel(ExtraSettingsWidget);
184         animPathLabel->setObjectName(QStringLiteral("animPathLabel"));
185
186         horizontalLayout->addWidget(animPathLabel);
187
188         animCountLabel = new QLabel(ExtraSettingsWidget);
189         animCountLabel->setObjectName(QStringLiteral("animCountLabel"));
190         sizePolicy.setHeightForWidth(animCountLabel->sizePolicy().hasHeightForWidth());
191         animCountLabel->setSizePolicy(sizePolicy);
192         animCountLabel->setAlignment(Qt::AlignRight|Qt::AlignTrailing|Qt::AlignVCenter);
193
194         horizontalLayout->addWidget(animCountLabel);
195
196         animScanButton = new QPushButton(ExtraSettingsWidget);
197         animScanButton->setObjectName(QStringLiteral("animScanButton"));
198
199         horizontalLayout->addWidget(animScanButton);
200
201
202         gridLayout->addLayout(horizontalLayout, 7, 3, 1, 4);
203
204         brightnessBox = new QCheckBox(ExtraSettingsWidget);
205         brightnessBox->setObjectName(QStringLiteral("brightnessBox"));
206
207         gridLayout->addWidget(brightnessBox, 3, 1, 1, 6);
208
209         trayBox = new QCheckBox(ExtraSettingsWidget);
210         trayBox->setObjectName(QStringLiteral("trayBox"));
211
212         gridLayout->addWidget(trayBox, 2, 1, 1, 6);
213
214         label = new QLabel(ExtraSettingsWidget);
215         label->setObjectName(QStringLiteral("label"));
216         QFont font;
217         font.setBold(true);
218         font.setWeight(75);
219         label->setFont(font);
220
221         gridLayout->addWidget(label, 0, 0, 1, 7);
222
223         line = new QFrame(ExtraSettingsWidget);
224         line->setObjectName(QStringLiteral("line"));
225         line->setFrameShape(QFrame::HLine);
226         line->setFrameShadow(QFrame::Sunken);
227
228         gridLayout->addWidget(line, 1, 0, 1, 7);
229
230         label_5 = new QLabel(ExtraSettingsWidget);
231         label_5->setObjectName(QStringLiteral("label_5"));
232         label_5->setFont(font);
233
234         gridLayout->addWidget(label_5, 5, 0, 1, 7);
235
236         line_3 = new QFrame(ExtraSettingsWidget);
237         line_3->setObjectName(QStringLiteral("line_3"));
238         line_3->setFrameShape(QFrame::HLine);
239         line_3->setFrameShadow(QFrame::Sunken);
240
```

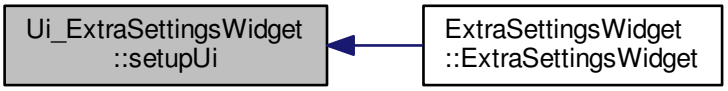
```
241         gridLayout->addWidget(line_3, 6, 0, 1, 7);
242 
243         label_2 = new QLabel(ExtraSettingsWidget);
244         label_2->setObjectName(QStringLiteral("label_2"));
245         label_2->setFont(font);
246 
247         gridLayout->addWidget(label_2, 9, 0, 1, 7);
248 
249         ditherBox = new QCheckBox(ExtraSettingsWidget);
250         ditherBox->setObjectName(QStringLiteral("ditherBox"));
251 
252         gridLayout->addWidget(ditherBox, 13, 1, 1, 6);
253 
254         delayBox = new QCheckBox(ExtraSettingsWidget);
255         delayBox->setObjectName(QStringLiteral("delayBox"));
256 
257         gridLayout->addWidget(delayBox, 14, 1, 1, 6);
258 
259         osxLabel = new QLabel(ExtraSettingsWidget);
260         osxLabel->setObjectName(QStringLiteral("osxLabel"));
261         osxLabel->setFont(font);
262 
263         gridLayout->addWidget(osxLabel, 15, 0, 1, 7);
264 
265         osxLine = new QFrame(ExtraSettingsWidget);
266         osxLine->setObjectName(QStringLiteral("osxLine"));
267         osxLine->setFrameShape(QFrame::HLine);
268         osxLine->setFrameShadow(QFrame::Sunken);
269 
270         gridLayout->addWidget(osxLine, 16, 0, 1, 7);
271 
272         mAccelBox = new QCheckBox(ExtraSettingsWidget);
273         mAccelBox->setObjectName(QStringLiteral("mAccelBox"));
274 
275         gridLayout->addWidget(mAccelBox, 17, 1, 1, 6);
276 
277         sSpeedWidget = new QWidget(ExtraSettingsWidget);
278         sSpeedWidget->setObjectName(QStringLiteral("sSpeedWidget"));
279         sizePolicy.setHeightForWidth(sSpeedWidget->sizePolicy().hasHeightForWidth());
280         sSpeedWidget->setSizePolicy(sizePolicy);
281         horizontalLayout_2 = new QHBoxLayout(sSpeedWidget);
282         horizontalLayout_2->setObjectName(QStringLiteral("horizontalLayout_2"));
283         horizontalLayout_2->setContentsMargins(0, 0, 0, 0);
284         label_10 = new QLabel(sSpeedWidget);
285         label_10->setObjectName(QStringLiteral("label_10"));
286 
287         horizontalLayout_2->addWidget(label_10);
288 
289         sSpeedBox = new QSpinBox(sSpeedWidget);
290         sSpeedBox->setObjectName(QStringLiteral("sSpeedBox"));
291         sSpeedBox->setMinimum(1);
292         sSpeedBox->setMaximum(10);
293         sSpeedBox->setValue(3);
294 
295         horizontalLayout_2->addWidget(sSpeedBox);
296 
297         label_11 = new QLabel(sSpeedWidget);
298         label_11->setObjectName(QStringLiteral("label_11"));
299         sizePolicy.setHeightForWidth(label_11->sizePolicy().hasHeightForWidth());
300         label_11->setSizePolicy(sizePolicy);
301 
302         horizontalLayout_2->addWidget(label_11);
303 
304 
305         gridLayout->addWidget(sSpeedWidget, 18, 5, 1, 2);
306 
307         buttonBox = new QDialogButtonBox(ExtraSettingsWidget);
308         buttonBox->setObjectName(QStringLiteral("buttonBox"));
309         buttonBox->setOrientation(Qt::Horizontal);
310         buttonBox->setStandardButtons(QDialogButtonBox::Ok);
311 
312         gridLayout->addWidget(buttonBox, 20, 0, 1, 7);
313 
314         QWidget::setTabOrder(trayBox, brightnessBox);
315         QWidget::setTabOrder(brightnessBox, animScanButton);
316         QWidget::setTabOrder(animScanButton, fpsBox);
317         QWidget::setTabOrder(fpsBox, ditherBox);
318         QWidget::setTabOrder(ditherBox, delayBox);
319         QWidget::setTabOrder(delayBox, mAccelBox);
320         QWidget::setTabOrder(mAccelBox, sAccelBox);
321         QWidget::setTabOrder(sAccelBox, sSpeedBox);
322 
323         retranslateUi(ExtraSettingsWidget);
324         QObject::connect(buttonBox, SIGNAL(accepted()), ExtraSettingsWidget, SLOT(accept()));
325         QObject::connect(buttonBox, SIGNAL(rejected()), ExtraSettingsWidget, SLOT(reject()));
326 
327         QMetaObject::connectSlotsByName(ExtraSettingsWidget);
```

```
328     } // setupUi
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.11.3 Field Documentation

12.11.3.1 QLabel* Ui_ExtraSettingsWidget::animCountLabel [inherited]

Definition at line 57 of file ui_extrasettingswidget.h.

Referenced by ExtraSettingsWidget::on_animScanButton_clicked(), Ui_ExtraSettingsWidget::retranslateUi(), and Ui_ExtraSettingsWidget::setupUi().

12.11.3.2 QLabel* Ui_ExtraSettingsWidget::animPathLabel [inherited]

Definition at line 56 of file ui_extrasettingswidget.h.

Referenced by ExtraSettingsWidget::ExtraSettingsWidget(), Ui_ExtraSettingsWidget::retranslateUi(), and Ui_ExtraSettingsWidget::setupUi().

12.11.3.3 QPushButton* Ui_ExtraSettingsWidget::animScanButton [inherited]

Definition at line 58 of file ui_extrasettingswidget.h.

Referenced by Ui_ExtraSettingsWidget::retranslateUi(), and Ui_ExtraSettingsWidget::setupUi().

12.11.3.4 QCheckBox* Ui_ExtraSettingsWidget::brightnessBox [inherited]

Definition at line 59 of file ui_extrasettingswidget.h.

Referenced by ExtraSettingsWidget::ExtraSettingsWidget(), Ui_ExtraSettingsWidget::retranslateUi(), and Ui_ExtraSettingsWidget::setupUi().

12.11.3.5 QDialogButtonBox* Ui_ExtraSettingsWidget::buttonBox [inherited]

Definition at line 76 of file ui_extrasettingswidget.h.

Referenced by Ui_ExtraSettingsWidget::setupUi().

12.11.3.6 QCheckBox* Ui_ExtraSettingsWidget::delayBox [inherited]

Definition at line 67 of file ui_extrasettingswidget.h.

Referenced by ExtraSettingsWidget::ExtraSettingsWidget(), Ui_ExtraSettingsWidget::retranslateUi(), and Ui_ExtraSettingsWidget::setupUi().

12.11.3.7 QCheckBox* Ui_ExtraSettingsWidget::ditherBox [inherited]

Definition at line 66 of file ui_extrasettingswidget.h.

Referenced by ExtraSettingsWidget::ExtraSettingsWidget(), Ui_ExtraSettingsWidget::retranslateUi(), and Ui_ExtraSettingsWidget::setupUi().

12.11.3.8 QSpinBox* Ui_ExtraSettingsWidget::fpsBox [inherited]

Definition at line 42 of file ui_extrasettingswidget.h.

Referenced by ExtraSettingsWidget::ExtraSettingsWidget(), and Ui_ExtraSettingsWidget::setupUi().

12.11.3.9 QLabel* Ui_ExtraSettingsWidget::fpsWarnLabel [inherited]

Definition at line 39 of file ui_extrasettingswidget.h.

Referenced by ExtraSettingsWidget::ExtraSettingsWidget(), ExtraSettingsWidget::on_fpsBox_valueChanged(), Ui_ExtraSettingsWidget::retranslateUi(), and Ui_ExtraSettingsWidget::setupUi().

12.11.3.10 QGridLayout* Ui_ExtraSettingsWidget::gridLayout [inherited]

Definition at line 34 of file ui_extrasettingswidget.h.

Referenced by Ui_ExtraSettingsWidget::setupUi().

12.11.3.11 QHBoxLayout* Ui_ExtraSettingsWidget::horizontalLayout [inherited]

Definition at line 55 of file ui_extrasettingswidget.h.

Referenced by Ui_ExtraSettingsWidget::setupUi().

12.11.3.12 QHBoxLayout* Ui_ExtraSettingsWidget::horizontalLayout_2 [inherited]

Definition at line 72 of file ui_extrasettingswidget.h.

Referenced by Ui_ExtraSettingsWidget::setupUi().

12.11.3.13 QSpacerItem* Ui_ExtraSettingsWidget::horizontalSpacer [inherited]

Definition at line 53 of file ui_extrasettingswidget.h.

Referenced by Ui_ExtraSettingsWidget::setupUi().

12.11.3.14 QSpacerItem* Ui_ExtraSettingsWidget::horizontalSpacer_2 [inherited]

Definition at line 35 of file ui_extrasettingswidget.h.

Referenced by Ui_ExtraSettingsWidget::setupUi().

12.11.3.15 QSpacerItem* Ui_ExtraSettingsWidget::horizontalSpacer_3 [inherited]

Definition at line 40 of file ui_extrasettingswidget.h.

Referenced by Ui_ExtraSettingsWidget::setupUi().

12.11.3.16 QLabel* Ui_ExtraSettingsWidget::label [inherited]

Definition at line 61 of file ui_extrasettingswidget.h.

Referenced by Ui_ExtraSettingsWidget::retranslateUi(), and Ui_ExtraSettingsWidget::setupUi().

12.11.3.17 QLabel* Ui_ExtraSettingsWidget::label_10 [inherited]

Definition at line 73 of file ui_extrasettingswidget.h.

Referenced by Ui_ExtraSettingsWidget::retranslateUi(), and Ui_ExtraSettingsWidget::setupUi().

12.11.3.18 QLabel* Ui_ExtraSettingsWidget::label_11 [inherited]

Definition at line 75 of file ui_extrasettingswidget.h.

Referenced by Ui_ExtraSettingsWidget::retranslateUi(), and Ui_ExtraSettingsWidget::setupUi().

12.11.3.19 QLabel* Ui_ExtraSettingsWidget::label_2 [inherited]

Definition at line 65 of file ui_extrasettingswidget.h.

Referenced by Ui_ExtraSettingsWidget::retranslateUi(), and Ui_ExtraSettingsWidget::setupUi().

12.11.3.20 QLabel* Ui_ExtraSettingsWidget::label_3 [inherited]

Definition at line 41 of file ui_extrasettingswidget.h.

Referenced by Ui_ExtraSettingsWidget::retranslateUi(), and Ui_ExtraSettingsWidget::setupUi().

12.11.3.21 QLabel* Ui_ExtraSettingsWidget::label_5 [inherited]

Definition at line 63 of file ui_extrasettingswidget.h.

Referenced by Ui_ExtraSettingsWidget::retranslateUi(), and Ui_ExtraSettingsWidget::setupUi().

12.11.3.22 QLabel* Ui_ExtraSettingsWidget::label_6 [inherited]

Definition at line 45 of file ui_extrasettingswidget.h.

Referenced by Ui_ExtraSettingsWidget::retranslateUi(), and Ui_ExtraSettingsWidget::setupUi().

12.11.3.23 QFrame* Ui_ExtraSettingsWidget::line [inherited]

Definition at line 62 of file ui_extrasettingswidget.h.

Referenced by Ui_ExtraSettingsWidget::setupUi().

12.11.3.24 QFrame* Ui_ExtraSettingsWidget::line_2 [inherited]

Definition at line 54 of file ui_extrasettingswidget.h.

Referenced by Ui_ExtraSettingsWidget::setupUi().

12.11.3.25 QFrame* Ui_ExtraSettingsWidget::line_3 [inherited]

Definition at line 64 of file ui_extrasettingswidget.h.

Referenced by Ui_ExtraSettingsWidget::setupUi().

12.11.3.26 QCheckBox* Ui_ExtraSettingsWidget::mAccelBox [inherited]

Definition at line 70 of file ui_extrasettingswidget.h.

Referenced by ExtraSettingsWidget::ExtraSettingsWidget(), Ui_ExtraSettingsWidget::retranslateUi(), and Ui_ExtraSettingsWidget::setupUi().

12.11.3.27 QLabel* Ui_ExtraSettingsWidget::osxLabel [inherited]

Definition at line 68 of file ui_extrasettingswidget.h.

Referenced by ExtraSettingsWidget::ExtraSettingsWidget(), Ui_ExtraSettingsWidget::retranslateUi(), and Ui_ExtraSettingsWidget::setupUi().

12.11.3.28 QFrame* Ui_ExtraSettingsWidget::osxLine [inherited]

Definition at line 69 of file ui_extrasettingswidget.h.

Referenced by ExtraSettingsWidget::ExtraSettingsWidget(), and Ui_ExtraSettingsWidget::setupUi().

12.11.3.29 QCheckBox* Ui_ExtraSettingsWidget::sAccelBox [inherited]

Definition at line 48 of file ui_extrasettingswidget.h.

Referenced by ExtraSettingsWidget::ExtraSettingsWidget(), ExtraSettingsWidget::on_sSpeedBox_valueChanged(), Ui_ExtraSettingsWidget::retranslateUi(), and Ui_ExtraSettingsWidget::setupUi().

12.11.3.30 QSpinBox* Ui_ExtraSettingsWidget::sSpeedBox [inherited]

Definition at line 74 of file ui_extrasettingswidget.h.

Referenced by ExtraSettingsWidget::ExtraSettingsWidget(), ExtraSettingsWidget::on_sAccelBox_clicked(), and Ui_ExtraSettingsWidget::setupUi().

12.11.3.31 QWidget* Ui_ExtraSettingsWidget::sSpeedWidget [inherited]

Definition at line 71 of file ui_extrasettingswidget.h.

Referenced by ExtraSettingsWidget::ExtraSettingsWidget(), ExtraSettingsWidget::on_sAccelBox_clicked(), and Ui_ExtraSettingsWidget::setupUi().

12.11.3.32 QCheckBox* Ui_ExtraSettingsWidget::trayBox [inherited]

Definition at line 60 of file ui_extrasettingswidget.h.

Referenced by ExtraSettingsWidget::ExtraSettingsWidget(), Ui_ExtraSettingsWidget::retranslateUi(), and Ui_ExtraSettingsWidget::setupUi().

12.11.3.33 QSpacerItem* Ui_ExtraSettingsWidget::verticalSpacer [inherited]

Definition at line 36 of file ui_extrasettingswidget.h.

Referenced by Ui_ExtraSettingsWidget::setupUi().

12.11.3.34 QSpacerItem* Ui_ExtraSettingsWidget::verticalSpacer_10 [inherited]

Definition at line 51 of file ui_extrasettingswidget.h.

Referenced by Ui_ExtraSettingsWidget::setupUi().

12.11.3.35 QSpacerItem* Ui_ExtraSettingsWidget::verticalSpacer_11 [inherited]

Definition at line 52 of file ui_extrasettingswidget.h.

Referenced by Ui_ExtraSettingsWidget::setupUi().

12.11.3.36 QSpacerItem* Ui_ExtraSettingsWidget::verticalSpacer_2 [inherited]

Definition at line 38 of file ui_extrasettingswidget.h.

Referenced by Ui_ExtraSettingsWidget::setupUi().

12.11.3.37 QSpacerItem* Ui_ExtraSettingsWidget::verticalSpacer_3 [inherited]

Definition at line 37 of file ui_extrasettingswidget.h.

Referenced by Ui_ExtraSettingsWidget::setupUi().

12.11.3.38 QSpacerItem* Ui_ExtraSettingsWidget::verticalSpacer_4 [inherited]

Definition at line 43 of file ui_extrasettingswidget.h.

Referenced by Ui_ExtraSettingsWidget::setupUi().

12.11.3.39 QSpacerItem* Ui_ExtraSettingsWidget::verticalSpacer_5 [inherited]

Definition at line 47 of file ui_extrasettingswidget.h.

Referenced by Ui_ExtraSettingsWidget::setupUi().

12.11.3.40 QSpacerItem* Ui_ExtraSettingsWidget::verticalSpacer_6 [inherited]

Definition at line 46 of file ui_extrasettingswidget.h.

Referenced by Ui_ExtraSettingsWidget::setupUi().

12.11.3.41 QSpacerItem* Ui_ExtraSettingsWidget::verticalSpacer_7 [inherited]

Definition at line 49 of file ui_extrasettingswidget.h.

Referenced by Ui_ExtraSettingsWidget::setupUi().

12.11.3.42 QSpacerItem* Ui_ExtraSettingsWidget::verticalSpacer_8 [inherited]

Definition at line 50 of file ui_extrasettingswidget.h.

Referenced by Ui_ExtraSettingsWidget::setupUi().

12.11.3.43 QSpacerItem* Ui_ExtraSettingsWidget::verticalSpacer_9 [inherited]

Definition at line 44 of file ui_extrasettingswidget.h.

Referenced by Ui_ExtraSettingsWidget::setupUi().

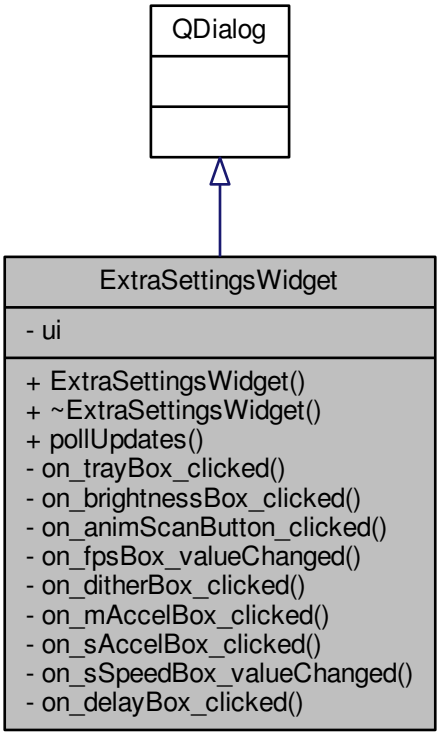
The documentation for this class was generated from the following file:

- src/ckb/ui_extrasettingswidget.h

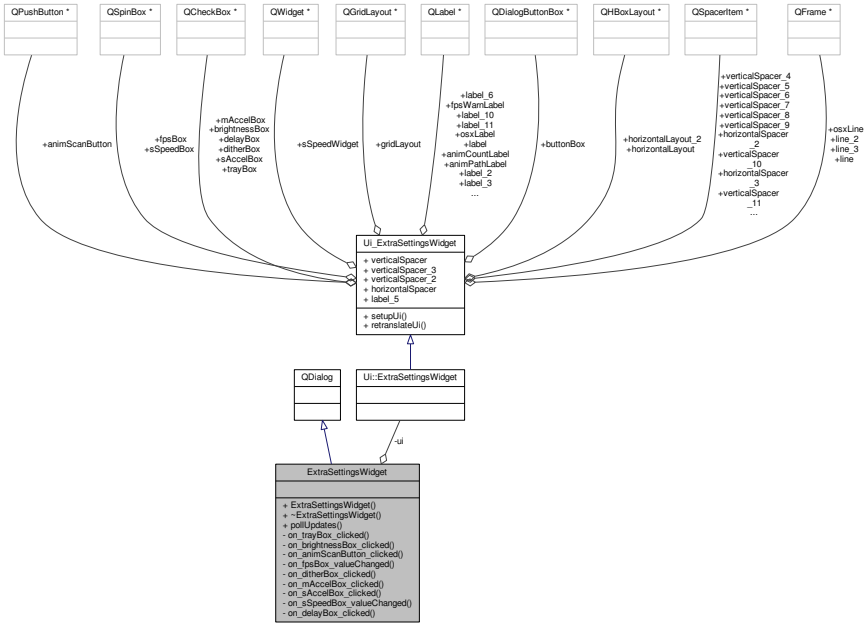
12.12 ExtraSettingsWidget Class Reference

`#include <src/ckb/extrasettingswidget.h>`

Inheritance diagram for ExtraSettingsWidget:



Collaboration diagram for ExtraSettingsWidget:



Public Member Functions

- ExtraSettingsWidget (QWidget *parent=0)
- ~ExtraSettingsWidget ()
- void pollUpdates ()

Private Slots

- void on_trayBox_clicked (bool checked)
- void on_brightnessBox_clicked (bool checked)
- void on_animScanButton_clicked ()
- void on_fpsBox_valueChanged (int arg1)
- void on_ditherBox_clicked (bool checked)
- void on_mAccelBox_clicked (bool checked)
- void on_sAccelBox_clicked (bool checked)
- void on_sSpeedBox_valueChanged (int arg1)
- void on_delayBox_clicked (bool checked)

Private Attributes

- Ui::ExtraSettingsWidget * ui

12.12.1 Detailed Description

Definition at line 10 of file extrasettingswidget.h.

12.12.2 Constructor & Destructor Documentation

12.12.2.1 ExtraSettingsWidget::ExtraSettingsWidget (QWidget * *parent* = 0) [explicit]

Read Macro Delay setting, update UI and Kb-internal flag

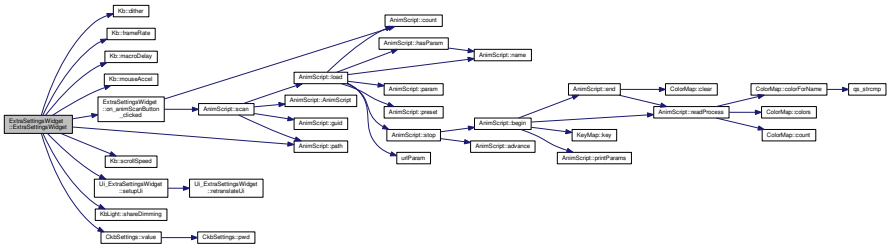
Definition at line 10 of file extrasettingswidget.cpp.

References Ui_ExtraSettingsWidget::animPathLabel, Ui_ExtraSettingsWidget::brightnessBox, Ui_ExtraSettingsWidget::delayBox, Kb::dither(), Ui_ExtraSettingsWidget::ditherBox, Ui_ExtraSettingsWidget::fpsBox, Ui_ExtraSettingsWidget::fpsWarnLabel, Kb::frameRate(), lastSharedDimming, Ui_ExtraSettingsWidget::mAccelBox, Kb::macroDelay(), KbLight::MAX_DIM, Kb::mouseAccel(), on_animScanButton_clicked(), Ui_ExtraSettingsWidget::osxLabel, Ui_ExtraSettingsWidget::osxLine, AnimScript::path(), Ui_ExtraSettingsWidget::sAccelBox, Kb::scrollSpeed(), Ui_ExtraSettingsWidget::setupUi(), KbLight::shareDimming(), Ui_ExtraSettingsWidget::sSpeedBox, Ui_ExtraSettingsWidget::sSpeedWidget, Ui_ExtraSettingsWidget::trayBox, ui, and CkbSettings::value().

```
10                                                      :
11     QDialog(parent),
12     ui(new Ui::ExtraSettingsWidget)
13 {
14     ui->setupUi(this);
15     CkbSettings settings("Program");
16
17     // Read frame rate from settings
18     int rate = settings.value("framerate").toInt();
19     if(rate <= 0 || rate > 60)
20         rate = 30;
21     ui->fpsBox->setValue(rate);
22     Kb::frameRate(rate);
23     ui->fpsWarnLabel->setVisible(rate > 30);
24
25     // Read global brightness setting (default = on, 100% brightness)
26     int dimming = settings.value("GlobalBrightness").toInt();
27     if(dimming < -1 || dimming > KbLight::MAX_DIM)
28         dimming = 0;
29     lastSharedDimming = dimming;
```

```
30     KbLight::shareDimming(dimming);
31     // Set checkbox value (-1 = don't share)
32     ui->brightnessBox->setChecked(dimming == -1);
33
34     // Read dither
35     bool dither = settings.value("Dither").toBool();
36     Kb::dither(dither);
37     ui->ditherBox->setChecked(dither);
38
39 #ifdef Q_OS_MACX
40     // Read OSX settings
41     bool noAccel = settings.value("DisableMouseAccel").toBool();
42     Kb::mouseAccel(!noAccel);
43     ui->mAccelBox->setChecked(noAccel);
44     bool noScrollAccel = settings.value("DisableScrollAccel").toBool();
45     int scrollSpeed = settings.value("ScrollSpeed", 3).toInt();
46     ui->sAccelBox->setChecked(noScrollAccel);
47     ui->sSpeedBox->setValue(scrollSpeed);
48     Kb::scrollSpeed(noScrollAccel ? scrollSpeed : 0);
49     if(!noScrollAccel)
50         ui->sSpeedWidget->hide();
51 #else
52     // Hide all OSX options on Linux
53     ui->osxLabel->hide();
54     ui->osxLine->hide();
55     ui->mAccelBox->hide();
56     ui->sAccelBox->hide();
57     ui->sSpeedWidget->hide();
58 #endif
59
60     // Read tray icon setting
61     ui->trayBox->setChecked(settings.value("SuppressTrayIcon").toBool());
62
63     // Update animation info
64     ui->animPathLabel->setText(AnimScript::path());
65     on_animScanButton_clicked();
66
68     bool macroDelay = settings.value("MacroDelay").toBool();
69     Kb::macroDelay(macroDelay);
70     ui->delayBox->setChecked(macroDelay);
71 }
```

Here is the call graph for this function:



12.12.2.2 ExtraSettingsWidget::~ExtraSettingsWidget ( )

Definition at line 73 of file extrasettingswidget.cpp.

References ui.

```
73                                          {
74     delete ui;
75 }
```

12.12.3 Member Function Documentation

12.12.3.1 void ExtraSettingsWidget::on_animScanButton_clicked ( ) [private], [slot]

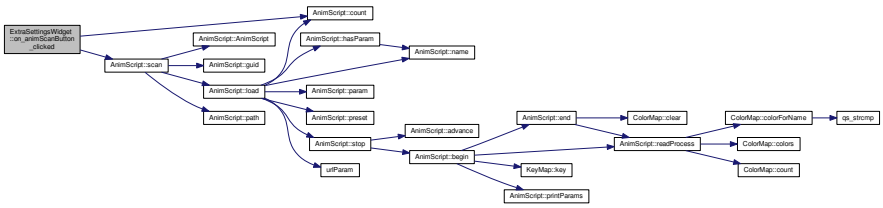
Definition at line 87 of file extrasettingswidget.cpp.

References Ui_ExtraSettingsWidget::animCountLabel, AnimScript::count(), AnimScript::scan(), and ui.

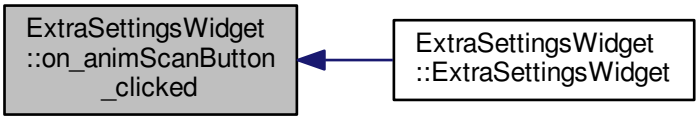
Referenced by ExtraSettingsWidget().

```
87                                                  {
88     AnimScript::scan();
89     int count = AnimScript::count();
90     if(count == 0)
91         ui->animCountLabel->setText("No animations found");
92     else if(count == 1)
93         ui->animCountLabel->setText("1 animation found");
94     else
95         ui->animCountLabel->setText(QString("%1 animations found").arg(count));
96 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



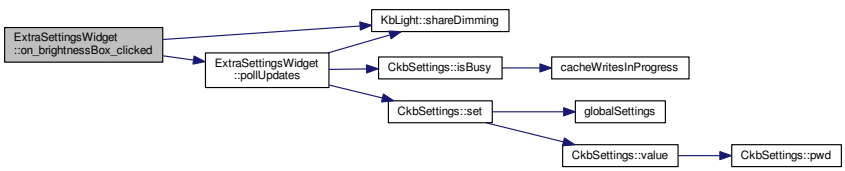
12.12.3.2 void ExtraSettingsWidget::on_brightnessBox_clicked (bool *checked*) [private], [slot]

Definition at line 82 of file extrasettingswidget.cpp.

References pollUpdates(), and KbLight::shareDimming().

```
82                                                          {
83     KbLight::shareDimming(checked ? -1 : 0);
84     pollUpdates();
85 }
```

Here is the call graph for this function:



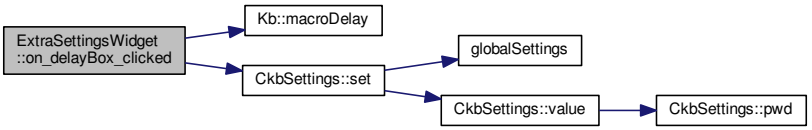
12.12.3.3 void ExtraSettingsWidget::on_delayBox_clicked (bool *checked*) [private], [slot]

Definition at line 143 of file extrasettingswidget.cpp.

References Kb::macroDelay(), and CkbSettings::set().

```
143                                                     {
144     CkbSettings::set("Program/MacroDelay", checked);
145     Kb::macroDelay(checked);
146 }
```

Here is the call graph for this function:



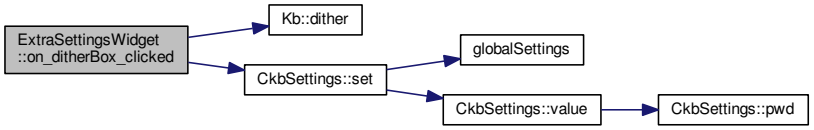
12.12.3.4 void ExtraSettingsWidget::on_ditherBox_clicked (bool *checked*) [private], [slot]

Definition at line 110 of file extrasettingswidget.cpp.

References Kb::dither(), and CkbSettings::set().

```
110                                                     {
111     CkbSettings::set("Program/Dither", checked);
112     Kb::dither(checked);
113 }
```

Here is the call graph for this function:



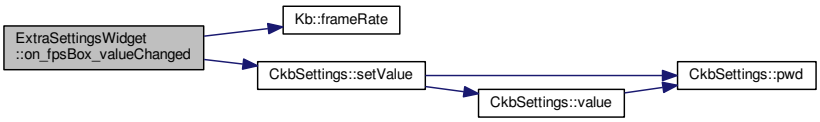
12.12.3.5 void ExtraSettingsWidget::on_fpsBox_valueChanged (int *arg1*) [private], [slot]

Definition at line 98 of file extrasettingswidget.cpp.

References Ui_ExtraSettingsWidget::fpsWarnLabel, Kb::frameRate(), CkbSettings::setValue(), and ui.

```
98                                                      {
99     // Set FPS
100     CkbSettings settings("Program");
101     settings.setValue("framerate", arg1);
102     Kb::frameRate(arg1);
103     // Show warning label if FPS is above 30
104     if(arg1 > 30)
105         ui->fpsWarnLabel->show();
106     else
107         ui->fpsWarnLabel->hide();
108 }
```

Here is the call graph for this function:



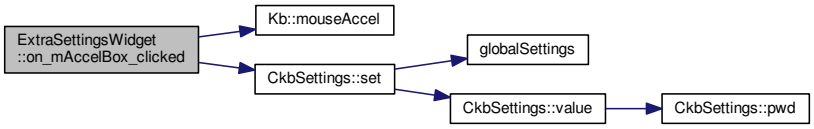
12.12.3.6 void ExtraSettingsWidget::on_mAccelBox_clicked (bool *checked*) [private], [slot]

Definition at line 124 of file extrasettingswidget.cpp.

References Kb::mouseAccel(), and CkbSettings::set().

```
124                                                         {
125     CkbSettings::set("Program/DisableMouseAccel", checked);
126     Kb::mouseAccel(!checked);
127 }
```

Here is the call graph for this function:



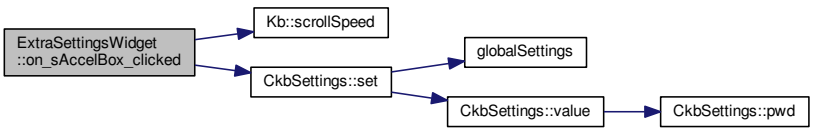
12.12.3.7 void ExtraSettingsWidget::on_sAccelBox_clicked (bool *checked*) [private], [slot]

Definition at line 129 of file extrasettingswidget.cpp.

References Kb::scrollSpeed(), CkbSettings::set(), Ui_ExtraSettingsWidget::sSpeedBox, Ui_ExtraSettingsWidget::sSpeedWidget, and ui.

```
129                                                         {
130     CkbSettings::set("Program/DisableScrollAccel", checked);
131     Kb::scrollSpeed(checked ? ui->sSpeedBox->value() : 0);
132     if(checked)
133         ui->sSpeedWidget->show();
134     else
135         ui->sSpeedWidget->hide();
136 }
```

Here is the call graph for this function:



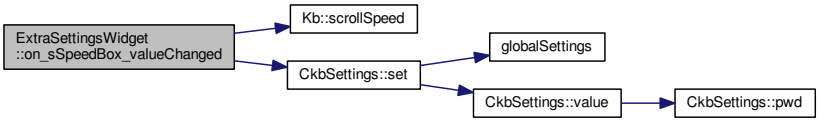
12.12.3.8 void ExtraSettingsWidget::on_sSpeedBox_valueChanged (int *arg1*) [private], [slot]

Definition at line 138 of file extrasettingswidget.cpp.

References Ui_ExtraSettingsWidget::sAccelBox, Kb::scrollSpeed(), CkbSettings::set(), and ui.

```
138                                                          {
139     CkbSettings::set("Program/ScrollSpeed", arg1);
140     Kb::scrollSpeed(ui->sAccelBox->isChecked() ? arg1 : 0);
141 }
```

Here is the call graph for this function:



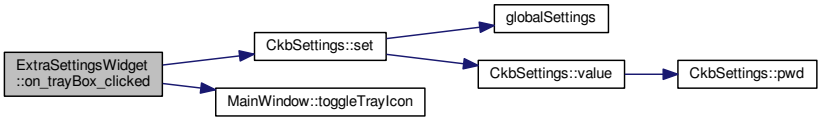
12.12.3.9 void ExtraSettingsWidget::on_trayBox_clicked (bool *checked*) [private], [slot]

Definition at line 77 of file extrasettingswidget.cpp.

References MainWindow::mainWindow, CkbSettings::set(), and MainWindow::toggleTrayIcon().

```
77                                                       {
78     CkbSettings::set("Program/SuppressTrayIcon", checked);
79     MainWindow::mainWindow->toggleTrayIcon(!checked);
80 }
```

Here is the call graph for this function:



12.12.3.10 void ExtraSettingsWidget::pollUpdates ( )

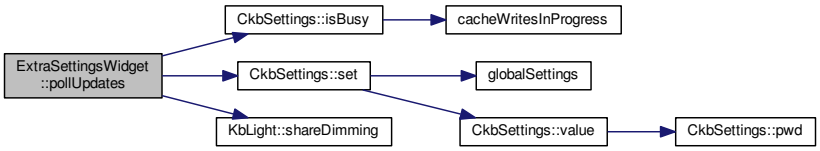
Definition at line 115 of file extrasettingswidget.cpp.

References CkbSettings::isBusy(), lastSharedDimming, CkbSettings::set(), and KbLight::shareDimming().

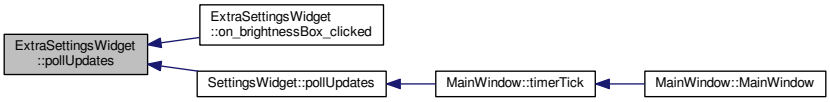
Referenced by on_brightnessBox_clicked(), and SettingsWidget::pollUpdates().

```
115                                      {
116     // Check for changes to shared brightness setting
117     int dimming = KbLight::shareDimming();
118     if(dimming != lastSharedDimming && !CkbSettings::isBusy()){
119         CkbSettings::set("Program/GlobalBrightness", dimming);
120         lastSharedDimming = dimming;
121     }
122 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.12.4 Field Documentation

12.12.4.1 Ui::ExtraSettingsWidget* ExtraSettingsWidget::ui [private]

Definition at line 33 of file extrasettingswidget.h.

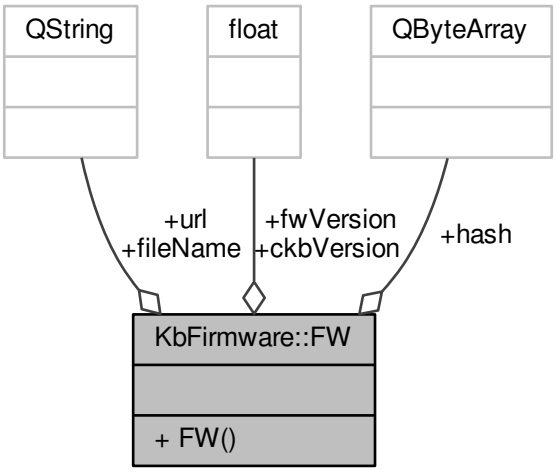
Referenced by ExtraSettingsWidget(), on_animScanButton_clicked(), on_fpsBox_valueChanged(), on_sAccelBox-_clicked(), on_sSpeedBox_valueChanged(), and ~ExtraSettingsWidget().

The documentation for this class was generated from the following files:

- src/ckb/extrasettingswidget.h
- src/ckb/extrasettingswidget.cpp

12.13 KbFirmware::FW Struct Reference

Collaboration diagram for KbFirmware::FW:



Public Member Functions

- FW ()

Data Fields

- QString url
- QString fileName
- QByteArray hash
- float fwVersion
- float ckbVersion

12.13.1 Detailed Description

Definition at line 37 of file kbfirmware.h.

12.13.2 Constructor & Destructor Documentation

12.13.2.1 KbFirmware::FW::FW ( )

Definition at line 12 of file kbfirmware.cpp.

```
12 : fwVersion(0.f), ckbVersion(0.f) {}
```

12.13.3 Field Documentation

12.13.3.1 float KbFirmware::FW::ckbVersion

Definition at line 40 of file kbfirmware.h.

Referenced by KbFirmware::_latestForBoard(), and KbFirmware::processDownload().

12.13.3.2 QString KbFirmware::FW::fileName

Definition at line 38 of file kbfirmware.h.

Referenced by KbFirmware::_fileForBoard(), and KbFirmware::processDownload().

12.13.3.3 float KbFirmware::FW::fwVersion

Definition at line 40 of file kbfirmware.h.

Referenced by KbFirmware::_latestForBoard(), and KbFirmware::processDownload().

12.13.3.4 QByteArray KbFirmware::FW::hash

Definition at line 39 of file kbfirmware.h.

Referenced by KbFirmware::_fileForBoard(), KbFirmware::_latestForBoard(), and KbFirmware::processDownload().

12.13.3.5 QString KbFirmware::FW::url

Definition at line 38 of file kbfirmware.h.

Referenced by KbFirmware::_fileForBoard(), and KbFirmware::processDownload().

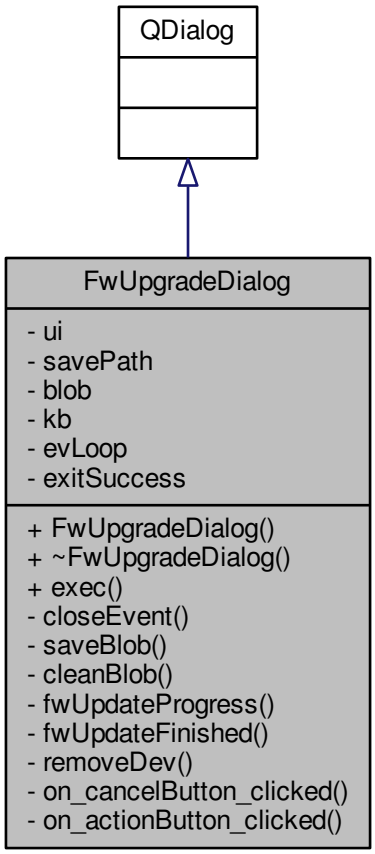
The documentation for this struct was generated from the following files:

- src/ckb/kbfirmware.h
- src/ckb/kbfirmware.cpp

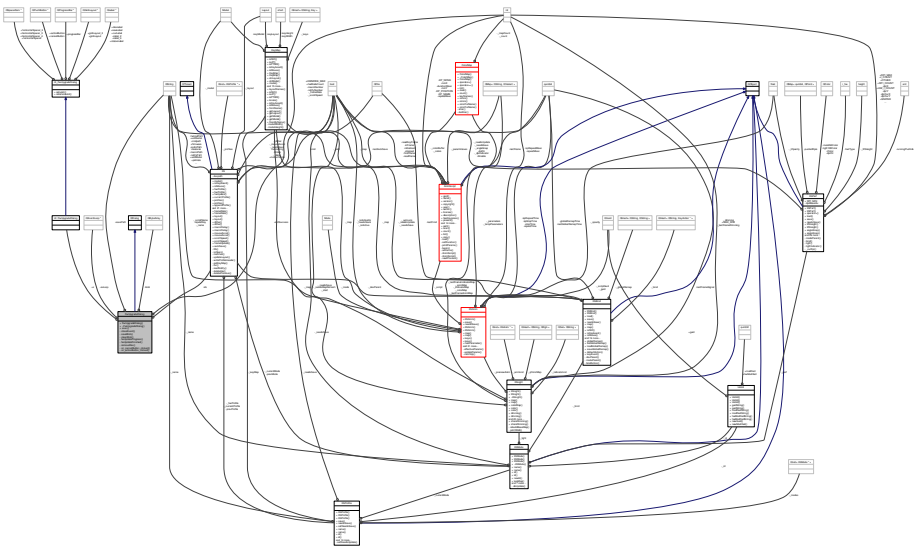
12.14 FwUpgradeDialog Class Reference

`#include <src/ckb/fwupgradedialog.h>`

Inheritance diagram for FwUpgradeDialog:



Collaboration diagram for FwUpgradeDialog:



Public Member Functions

- FwUpgradeDialog (QWidget *parent, float newV, const QByteArray &fwBlob, Kb *device)
- ~FwUpgradeDialog ()
- int exec ()

Private Slots

- void fwUpdateProgress (int current, int total)
- void fwUpdateFinished (bool succeeded)
- void removeDev ()
- void on_cancelButton_clicked ()
- void on_actionButton_clicked ()

Private Member Functions

- void closeEvent (QCloseEvent *event)
- const QString & saveBlob ()
- void cleanBlob ()

Private Attributes

- Ui::FwUpgradeDialog * ui
- QString savePath
- QByteArray blob
- Kb * kb
- QEventLoop * evLoop
- bool exitSuccess

12.14.1 Detailed Description

Definition at line 13 of file fwupgradedialog.h.

12.14.2 Constructor & Destructor Documentation

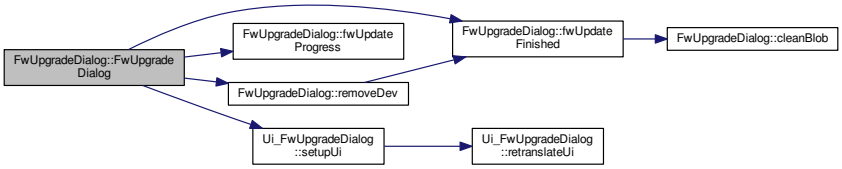
12.14.2.1 FwUpgradeDialog::FwUpgradeDialog (QWidget * *parent,* float *newV,* const QByteArray & *fwBlob,* Kb * *device*)

Definition at line 32 of file fwupgradedialog.cpp.

References Ui_FwUpgradeDialog::curLabel, Ui_FwUpgradeDialog::devLabel, DIALOG_HEIGHT_MIN, DIALOG_WIDTH, Kb::firmware, fwUpdateFinished(), fwUpdateProgress(), kb, Ui_FwUpgradeDialog::newLabel, removeDev(), Ui_FwUpgradeDialog::setupUi(), ui, and Kb::usbModel.

```
32                                                                                                    :
33     QDialog(parent),
34     ui(new Ui::FwUpgradeDialog),
35     blob(fwBlob), kb(device), evLoop(0), exitSuccess(true)
36 {
37     ui->setupUi(this);
38     ui->curLabel->setText(kb->firmware);
39     ui->newLabel->setText(QString::number(newV, 'f', 2));
40     ui->devLabel->setText(kb->usbModel);
41
42     connect(device, SIGNAL(destroyed()), this, SLOT(removeDev()));
43     connect(device, SIGNAL(fwUpdateProgress(int,int)), this, SLOT(
       fwUpdateProgress(int,int)));
44     connect(device, SIGNAL(fwUpdateFinished(bool)), this, SLOT(
       fwUpdateFinished(bool)));
45
46     setFixedSize(DIALOG_WIDTH, DIALOG_HEIGHT_MIN);
47 }
```

Here is the call graph for this function:



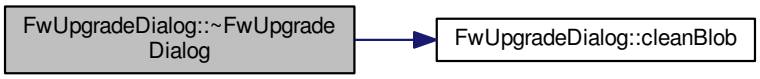
12.14.2.2 FwUpgradeDialog::~FwUpgradeDialog ( )

Definition at line 49 of file fwupgradedialog.cpp.

References cleanBlob(), and ui.

```
49                                      {
50     cleanBlob();
51     delete ui;
52 }
```

Here is the call graph for this function:



12.14.3 Member Function Documentation

12.14.3.1 void FwUpgradeDialog::cleanBlob ( ) [private]

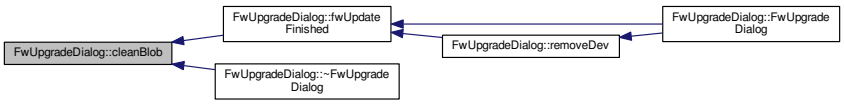
Definition at line 78 of file fwupgradedialog.cpp.

References savePath.

Referenced by fwUpdateFinished(), and ~FwUpgradeDialog().

```
78                                      {
79     if(savePath.isEmpty())
80         return;
81     QFile(savePath).remove();
82     savePath = "";
83 }
```

Here is the caller graph for this function:



12.14.3.2 void FwUpgradeDialog::closeEvent (QCloseEvent * *event*) [private]

Definition at line 54 of file fwupgradedialog.cpp.

```
54                                                {
55     event->ignore();
56 }
```

12.14.3.3 int FwUpgradeDialog::exec ( )

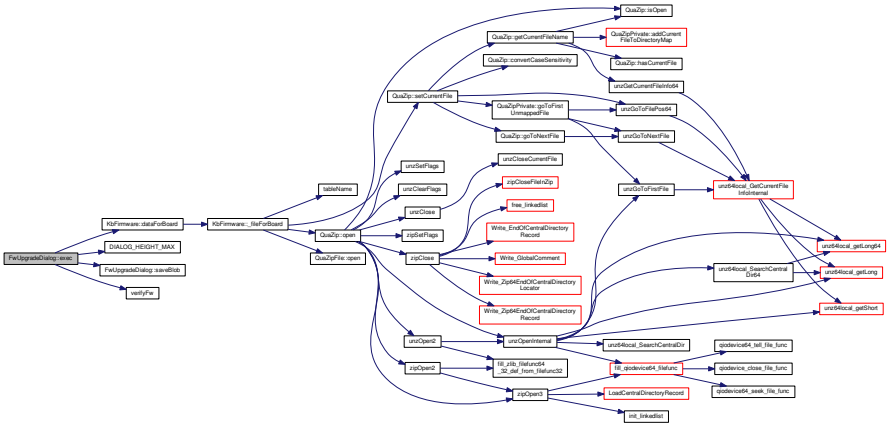
Definition at line 108 of file fwupgradedialog.cpp.

References Ui_FwUpgradeDialog::actionButton, blob, Ui_FwUpgradeDialog::cancelButton, KbFirmware::dataForBoard(), DIALOG_HEIGHT_MAX(), DIALOG_WIDTH, evLoop, exitSuccess, Kb::features, kb, Ui_FwUpgradeDialog::newLabel, Ui_FwUpgradeDialog::progressBar, saveBlob(), Ui_FwUpgradeDialog::statusLabel, ui, and verifyFw().

Referenced by KbWidget::on_fwUpdButton_clicked().

```
108                          {
109     QString features = kb->features;
110     if(!blob.isEmpty()){
111         // If a blob was already specified, check its version and validity
112         float newV = verifyFw(blob, features);
113         if(newV == 0.f){
114             QMessageBox::warning(parentWidget(), "Error", "<center>Not a valid firmware for this
    device.</center>");
115             return QDialog::Rejected;
116         }
117         ui->newLabel->setText(QString::number(newV, 'f', 2));
118     } else {
119         // Download a new blob file
120         ui->progressBar->show();
121         ui->cancelButton->setEnabled(false);
122         ui->actionButton->setEnabled(false);
123         show();
124         // This can take a while
125         blob = KbFirmware::dataForBoard(features);
126         // Check validity
127         float newV = verifyFw(blob, features);
128         if(newV == 0.f){
129             hide();
130             QMessageBox::warning(parentWidget(), "Error", "<center>There was a problem with the downloaded
    file.<br />Please try again later.</center>");
131             return QDialog::Rejected;
132         }
133     }
134     // Save temporary file
135     if(saveBlob().isEmpty()){
136         hide();
137         QMessageBox::warning(parentWidget(), "Error", "<center>Unable to save temporary file.</center>");
138         return QDialog::Rejected;
139     }
140     // Set up UI
141     ui->progressBar->setValue(0);
142     ui->progressBar->setMaximum(1);
143     ui->progressBar->setTextVisible(false);
144     ui->statusLabel->setText("Ready to install new firmware.<br /><br /><b>Disclaimer:</b>
    ckb-next is not endorsed by Corsair.<br />This is <i>unlikely</i> to cause any damage, however the developers
    of this software do not accept any responsibility in such an event.");
145     ui->cancelButton->setEnabled(true);
146     ui->actionButton->setEnabled(true);
147     setFixedSize(DIALOG_WIDTH, DIALOG_HEIGHT_MAX);
148     show();
149     // Run modal event loop
150     evLoop = new QEventLoop(this);
151     evLoop->exec();
152     delete evLoop;
153     hide();
154     return exitSuccess ? QDialog::Accepted : QDialog::Rejected;
155 }
```

Here is the call graph for this function:



Here is the caller graph for this function:

12.14.3.4 void FwUpgradeDialog::fwUpdateFinished (bool *succeeded*) [private],[slot]

Definition at line 171 of file fwupgradedialog.cpp.

References Ui_FwUpgradeDialog::actionButton, cleanBlob(), evLoop, Ui_FwUpgradeDialog::progressBar, Ui_FwUpgradeDialog::statusLabel, and ui.

Referenced by FwUpgradeDialog(), and removeDev().

```
171                                                 {
172     cleanBlob();
173     if(succeeded)
174         ui->statusLabel->setText("Update successful!");
175     else
176         ui->statusLabel->setText("Update failed.");
177     ui->actionButton->setText("OK");
178     ui->actionButton->setEnabled(true);
179     ui->progressBar->setMaximum(1);
180     ui->progressBar->setValue(1);
181     // Exit after 10s
182     if(evLoop)
183         QTimer::singleShot(10000, evLoop, SLOT(quit()));
184 }
```

Here is the call graph for this function:

Here is the caller graph for this function:

12.14.3.5 void FwUpgradeDialog::fwUpdateProgress (int *current,* int *total*) [private],[slot]

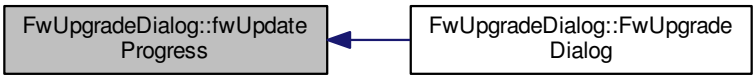
Definition at line 164 of file fwupgradedialog.cpp.

References Ui_FwUpgradeDialog::progressBar, and ui.

Referenced by FwUpgradeDialog().

```
164                                                         {
165     if(current > 0 && total > 0){
166         ui->progressBar->setMaximum(total);
167         ui->progressBar->setValue(current);
168     }
169 }
```

Here is the caller graph for this function:



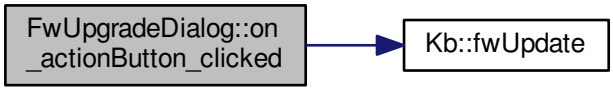
12.14.3.6 void FwUpgradeDialog::on_actionButton_clicked ( ) [private],[slot]

Definition at line 192 of file fwupgradedialog.cpp.

References Ui_FwUpgradeDialog::actionButton, Ui_FwUpgradeDialog::cancelButton, DIALOG_HEIGHT_MIN, DIALOG_WIDTH, evLoop, Kb::fwUpdate(), kb, Ui_FwUpgradeDialog::progressBar, savePath, Ui_FwUpgradeDialog::statusLabel, and ui.

```
192                                             {
193     if(!savePath.isEmpty() && kb){
194         // Start upgrade
195         setFixedSize(DIALOG_WIDTH, DIALOG_HEIGHT_MIN);
196         ui->progressBar->show();
197         ui->progressBar->setValue(0);
198         ui->progressBar->setMaximum(0);
199         ui->progressBar->setTextVisible(true);
200         ui->cancelButton->hide();
201         ui->actionButton->setEnabled(false);
202         ui->actionButton->setText("Please wait");
203         ui->statusLabel->setText("Installing firmware...");
204         kb->fwUpdate(savePath);
205     } else {
206         // Finished, close dialog.
207         if(evLoop)
208             evLoop->quit();
209     }
210 }
```

Here is the call graph for this function:



12.14.3.7 void FwUpgradeDialog::on_cancelButton_clicked ( ) [private], [slot]

Definition at line 186 of file fwupgradedialog.cpp.

References evLoop, and exitSuccess.

```
186                                                 {
187     exitSuccess = false;
188     if(evLoop)
189         evLoop->quit();
190 }
```

12.14.3.8 void FwUpgradeDialog::removeDev ( ) [private], [slot]

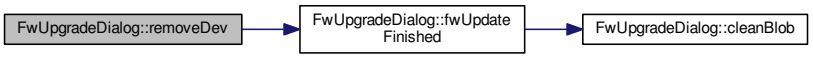
Definition at line 157 of file fwupgradedialog.cpp.

References fwUpdateFinished(), kb, and savePath.

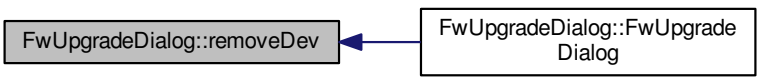
Referenced by FwUpgradeDialog().

```
157                                 {
158     kb = 0;
159     // Assume success if upgrade in progress
160     if(!savePath.isEmpty())
161         fwUpdateFinished(true);
162 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.14.3.9 const QString & FwUpgradeDialog::saveBlob ( ) [private]

Definition at line 58 of file fwupgradedialog.cpp.

References blob, and savePath.

Referenced by exec().

```
58                                      {
59     if(!savePath.isEmpty())
60         return savePath;
61     QDir tmp = QDir::temp();
62     qint64 pid = QCoreApplication::applicationPid();
63     QString path = tmp.absoluteFilePath(QString("ckb-%1-fwblob.bin").arg(pid));
64     QFile output(path);
65     if(!output.open(QIODevice::WriteOnly)){
66         return savePath;
67     }
68     if(!output.write(blob)){
69         output.close();
70         tmp.remove(path);
71         return savePath;
72     }
73     output.close();
74     savePath = path;
75     return savePath;
76 }
```

Here is the caller graph for this function:

12.14.4 Field Documentation

12.14.4.1 QByteArray FwUpgradeDialog::blob [private]

Definition at line 41 of file fwupgradedialog.h.

Referenced by exec(), and saveBlob().

12.14.4.2 QEventLoop* FwUpgradeDialog::evLoop [private]

Definition at line 45 of file fwupgradedialog.h.

Referenced by exec(), fwUpdateFinished(), on_actionButton_clicked(), and on_cancelButton_clicked().

12.14.4.3 bool FwUpgradeDialog::exitSuccess [private]

Definition at line 46 of file fwupgradedialog.h.

Referenced by exec(), and on_cancelButton_clicked().

12.14.4.4 Kb* FwUpgradeDialog::kb [private]

Definition at line 42 of file fwupgradedialog.h.

Referenced by exec(), FwUpgradeDialog(), on_actionButton_clicked(), and removeDev().

12.14.4.5 QString FwUpgradeDialog::savePath [private]

Definition at line 37 of file fwupgradedialog.h.

Referenced by cleanBlob(), on_actionButton_clicked(), removeDev(), and saveBlob().

12.14.4.6 Ui::FwUpgradeDialog* FwUpgradeDialog::ui [private]

Definition at line 32 of file fwupgradedialog.h.

Referenced by exec(), fwUpdateFinished(), fwUpdateProgress(), FwUpgradeDialog(), on_actionButton_clicked(), and ~FwUpgradeDialog().

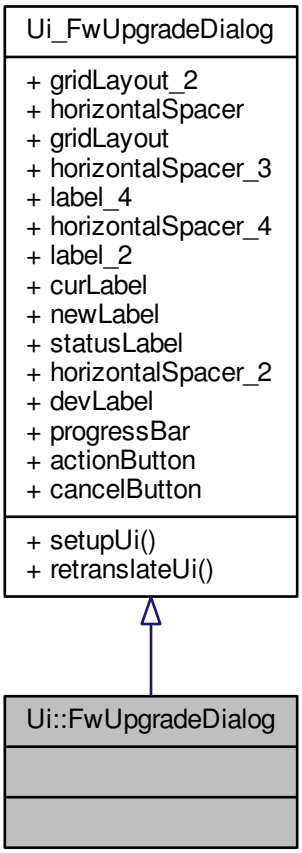
The documentation for this class was generated from the following files:

- src/ckb/fwupgradedialog.h
- src/ckb/fwupgradedialog.cpp

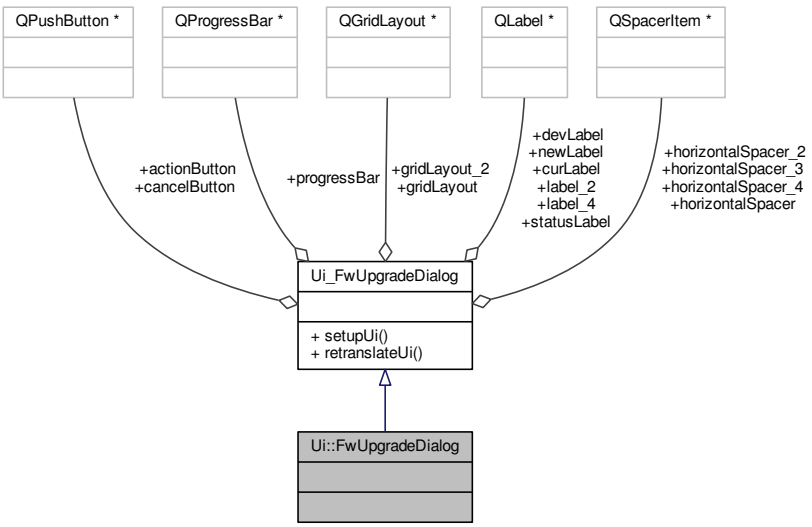
12.15 Ui::FwUpgradeDialog Class Reference

`#include <src/ckb/ui_fwupgradedialog.h>`

Inheritance diagram for Ui::FwUpgradeDialog:



Collaboration diagram for Ui::FwUpgradeDialog:



Public Member Functions

- void setupUi (QDialog *FwUpgradeDialog)
- void retranslateUi (QDialog *FwUpgradeDialog)

Data Fields

- QGridLayout * gridLayout_2
- QSpacerItem * horizontalSpacer
- QGridLayout * gridLayout
- QSpacerItem * horizontalSpacer_3
- QLabel * label_4
- QSpacerItem * horizontalSpacer_4
- QLabel * label_2
- QLabel * curLabel
- QLabel * newLabel
- QLabel * statusLabel
- QSpacerItem * horizontalSpacer_2
- QLabel * devLabel
- QProgressBar * progressBar
- QPushButton * actionButton
- QPushButton * cancelButton

12.15.1 Detailed Description

Definition at line 172 of file ui_fwupgradedialog.h.

12.15.2 Member Function Documentation

12.15.2.1 void Ui_FwUpgradeDialog::retranslateUi (QDialog * *FwUpgradeDialog*) [inline],[inherited]

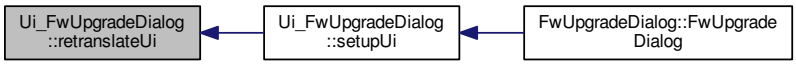
Definition at line 156 of file ui_fwupgradedialog.h.

References Ui_FwUpgradeDialog::actionButton, Ui_FwUpgradeDialog::cancelButton, Ui_FwUpgradeDialog::curLabel, Ui_FwUpgradeDialog::devLabel, Ui_FwUpgradeDialog::label_2, Ui_FwUpgradeDialog::label_4, Ui_FwUpgradeDialog::newLabel, and Ui_FwUpgradeDialog::statusLabel.

Referenced by Ui_FwUpgradeDialog::setupUi().

```
157     {
158         FwUpgradeDialog->setWindowTitle(QApplication::translate("FwUpgradeDialog", "Firmware update",
    Q_NULLPTR));
159         label_4->setText(QApplication::translate("FwUpgradeDialog", "New:", Q_NULLPTR));
160         label_2->setText(QApplication::translate("FwUpgradeDialog", "Current:", Q_NULLPTR));
161         curLabel->setText(QApplication::translate("FwUpgradeDialog", "0.00", Q_NULLPTR));
162         newLabel->setText(QApplication::translate("FwUpgradeDialog", "0.00", Q_NULLPTR));
163         statusLabel->setText(QApplication::translate("FwUpgradeDialog", "Downloading new
     firmware, please wait...", Q_NULLPTR));
164         devLabel->setText(QApplication::translate("FwUpgradeDialog", "Device", Q_NULLPTR));
165         actionButton->setText(QApplication::translate("FwUpgradeDialog", "Start", Q_NULLPTR));
166         cancelButton->setText(QApplication::translate("FwUpgradeDialog", "Cancel", Q_NULLPTR));
167     } // retranslateUi
```

Here is the caller graph for this function:



12.15.2.2 void Ui_FwUpgradeDialog::setupUi (QDialog * *FwUpgradeDialog*) [inline],[inherited]

Definition at line 45 of file ui_fwupgradedialog.h.

References Ui_FwUpgradeDialog::actionButton, Ui_FwUpgradeDialog::cancelButton, Ui_FwUpgradeDialog::curLabel, Ui_FwUpgradeDialog::devLabel, Ui_FwUpgradeDialog::gridLayout, Ui_FwUpgradeDialog::gridLayout_2, Ui_FwUpgradeDialog::horizontalSpacer, Ui_FwUpgradeDialog::horizontalSpacer_2, Ui_FwUpgradeDialog::horizontalSpacer_3, Ui_FwUpgradeDialog::horizontalSpacer_4, Ui_FwUpgradeDialog::label_2, Ui_FwUpgradeDialog::label_4, Ui_FwUpgradeDialog::newLabel, Ui_FwUpgradeDialog::progressBar, Ui_FwUpgradeDialog::retranslateUi(), and Ui_FwUpgradeDialog::statusLabel.

Referenced by FwUpgradeDialog::FwUpgradeDialog().

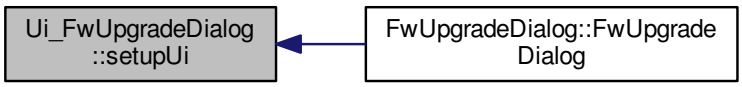
```
46     {
47         if (FwUpgradeDialog->objectName().isEmpty())
48             FwUpgradeDialog->setObjectName(QStringLiteral("FwUpgradeDialog"));
49         FwUpgradeDialog->resize(420, 280);
50         QSizePolicy sizePolicy(QSizePolicy::Fixed, QSizePolicy::Fixed);
51         sizePolicy.setHorizontalStretch(0);
52         sizePolicy.setVerticalStretch(0);
53         sizePolicy.setHeightForWidth(FwUpgradeDialog->sizePolicy().hasHeightForWidth());
54         FwUpgradeDialog->setSizePolicy(sizePolicy);
55         gridLayout_2 = new QGridLayout(FwUpgradeDialog);
56         gridLayout_2->setObjectName(QStringLiteral("gridLayout_2"));
57         horizontalSpacer = new QSpacerItem(127, 20, QSizePolicy::Expanding,
    QSizePolicy::Minimum);
58
59         gridLayout_2->addItem(horizontalSpacer, 1, 0, 1, 1);
60
61         gridLayout = new QGridLayout();
62         gridLayout->setObjectName(QStringLiteral("gridLayout"));
63         horizontalSpacer_3 = new QSpacerItem(40, 20, QSizePolicy::Expanding,
    QSizePolicy::Minimum);
```

```
64
65          gridLayout->addItem(horizontalSpacer_3, 1, 0, 1, 1);
66
67          label_4 = new QLabel(FwUpgradeDialog);
68          label_4->setObjectName(QStringLiteral("label_4"));
69          QFont font;
70          font.setBold(true);
71          font.setWeight(75);
72          label_4->setFont(font);
73
74          gridLayout->addWidget(label_4, 1, 4, 1, 1);
75
76          horizontalSpacer_4 = new QSpacerItem(40, 20, QSizePolicy::Expanding,
      QSizePolicy::Minimum);
77
78          gridLayout->addItem(horizontalSpacer_4, 1, 6, 1, 1);
79
80          label_2 = new QLabel(FwUpgradeDialog);
81          label_2->setObjectName(QStringLiteral("label_2"));
82          label_2->setFont(font);
83
84          gridLayout->addWidget(label_2, 1, 1, 1, 1);
85
86          curLabel = new QLabel(FwUpgradeDialog);
87          curLabel->setObjectName(QStringLiteral("curLabel"));
88
89          gridLayout->addWidget(curLabel, 1, 2, 1, 1);
90
91          newLabel = new QLabel(FwUpgradeDialog);
92          newLabel->setObjectName(QStringLiteral("newLabel"));
93
94          gridLayout->addWidget(newLabel, 1, 5, 1, 1);
95
96          statusLabel = new QLabel(FwUpgradeDialog);
97          statusLabel->setObjectName(QStringLiteral("statusLabel"));
98          QSizePolicy sizePolicy1(QSizePolicy::Preferred, QSizePolicy::Expanding);
99          sizePolicy1.setHorizontalStretch(0);
100          sizePolicy1.setVerticalStretch(0);
101          sizePolicy1.setHeightForWidth(statusLabel->sizePolicy().hasHeightForWidth());
102          statusLabel->setSizePolicy(sizePolicy1);
103          statusLabel->setAlignment(Qt::AlignCenter);
104          statusLabel->setWordWrap(true);
105
106          gridLayout->addWidget(statusLabel, 2, 0, 1, 7);
107
108          horizontalSpacer_2 = new QSpacerItem(40, 20, QSizePolicy::Expanding,
      QSizePolicy::Minimum);
109
110          gridLayout->addItem(horizontalSpacer_2, 1, 3, 1, 1);
111
112          devLabel = new QLabel(FwUpgradeDialog);
113          devLabel->setObjectName(QStringLiteral("devLabel"));
114          QSizePolicy sizePolicy2(QSizePolicy::Preferred, QSizePolicy::Minimum);
115          sizePolicy2.setHorizontalStretch(0);
116          sizePolicy2.setVerticalStretch(0);
117          sizePolicy2.setHeightForWidth(devLabel->sizePolicy().hasHeightForWidth());
118          devLabel->setSizePolicy(sizePolicy2);
119          devLabel->setFont(font);
120          devLabel->setAlignment(Qt::AlignCenter);
121
122          gridLayout->addWidget(devLabel, 0, 0, 1, 7);
123
124          progressBar = new QProgressBar(FwUpgradeDialog);
125          progressBar->setObjectName(QStringLiteral("progressBar"));
126          progressBar->setMaximum(0);
127          progressBar->setValue(0);
128
129          gridLayout->addWidget(progressBar, 3, 0, 1, 7);
130
131
132          gridLayout_2->addLayout(gridLayout, 0, 0, 1, 3);
133
134          actionButton = new QPushButton(FwUpgradeDialog);
135          actionButton->setObjectName(QStringLiteral("actionButton"));
136          actionButton->setEnabled(false);
137
138          gridLayout_2->addWidget(actionButton, 1, 2, 1, 1);
139
140          cancelButton = new QPushButton(FwUpgradeDialog);
141          cancelButton->setObjectName(QStringLiteral("cancelButton"));
142          cancelButton->setEnabled(false);
143
144          gridLayout_2->addWidget(cancelButton, 1, 1, 1, 1);
145
146          QWidget::setTabOrder(actionButton, cancelButton);
147
148          retranslateUi(FwUpgradeDialog);
```

```
149
150         actionButton->setDefault(true);
151
152
153         QMetaObject::connectSlotsByName(FwUpgradeDialog);
154     } // setupUi
```

Here is the call graph for this function:

Here is the caller graph for this function:



12.15.3 Field Documentation

12.15.3.1 QPushButton* Ui_FwUpgradeDialog::actionButton [inherited]

Definition at line 42 of file ui_fwupgradedialog.h.

Referenced by FwUpgradeDialog::exec(), FwUpgradeDialog::fwUpdateFinished(), FwUpgradeDialog::on_actionButton_clicked(), Ui_FwUpgradeDialog::retranslateUi(), and Ui_FwUpgradeDialog::setupUi().

12.15.3.2 QPushButton* Ui_FwUpgradeDialog::cancelButton [inherited]

Definition at line 43 of file ui_fwupgradedialog.h.

Referenced by FwUpgradeDialog::exec(), FwUpgradeDialog::on_actionButton_clicked(), Ui_FwUpgradeDialog::retranslateUi(), and Ui_FwUpgradeDialog::setupUi().

12.15.3.3 QLabel* Ui_FwUpgradeDialog::curLabel [inherited]

Definition at line 36 of file ui_fwupgradedialog.h.

Referenced by FwUpgradeDialog::FwUpgradeDialog(), Ui_FwUpgradeDialog::retranslateUi(), and Ui_FwUpgradeDialog::setupUi().

12.15.3.4 QLabel* Ui_FwUpgradeDialog::devLabel [inherited]

Definition at line 40 of file ui_fwupgradedialog.h.

Referenced by FwUpgradeDialog::FwUpgradeDialog(), Ui_FwUpgradeDialog::retranslateUi(), and Ui_FwUpgrade-Dialog::setupUi().

12.15.3.5 QGridLayout* Ui_FwUpgradeDialog::gridLayout [inherited]

Definition at line 31 of file ui_fwupgradedialog.h.

Referenced by Ui_FwUpgradeDialog::setupUi().

12.15.3.6 QGridLayout* Ui_FwUpgradeDialog::gridLayout_2 [inherited]

Definition at line 29 of file ui_fwupgradedialog.h.

Referenced by Ui_FwUpgradeDialog::setupUi().

12.15.3.7 QSpacerItem* Ui_FwUpgradeDialog::horizontalSpacer [inherited]

Definition at line 30 of file ui_fwupgradedialog.h.

Referenced by Ui_FwUpgradeDialog::setupUi().

12.15.3.8 QSpacerItem* Ui_FwUpgradeDialog::horizontalSpacer_2 [inherited]

Definition at line 39 of file ui_fwupgradedialog.h.

Referenced by Ui_FwUpgradeDialog::setupUi().

12.15.3.9 QSpacerItem* Ui_FwUpgradeDialog::horizontalSpacer_3 [inherited]

Definition at line 32 of file ui_fwupgradedialog.h.

Referenced by Ui_FwUpgradeDialog::setupUi().

12.15.3.10 QSpacerItem* Ui_FwUpgradeDialog::horizontalSpacer_4 [inherited]

Definition at line 34 of file ui_fwupgradedialog.h.

Referenced by Ui_FwUpgradeDialog::setupUi().

12.15.3.11 QLabel* Ui_FwUpgradeDialog::label_2 [inherited]

Definition at line 35 of file ui_fwupgradedialog.h.

Referenced by Ui_FwUpgradeDialog::retranslateUi(), and Ui_FwUpgradeDialog::setupUi().

12.15.3.12 QLabel* Ui_FwUpgradeDialog::label_4 [inherited]

Definition at line 33 of file ui_fwupgradedialog.h.

Referenced by Ui_FwUpgradeDialog::retranslateUi(), and Ui_FwUpgradeDialog::setupUi().

12.15.3.13 QLabel* Ui_FwUpgradeDialog::newLabel [inherited]

Definition at line 37 of file ui_fwupgradedialog.h.

Referenced by FwUpgradeDialog::exec(), FwUpgradeDialog::FwUpgradeDialog(), Ui_FwUpgradeDialog::retranslateUi(), and Ui_FwUpgradeDialog::setupUi().

12.15.3.14 QProgressBar* Ui_FwUpgradeDialog::progressBar [inherited]

Definition at line 41 of file ui_fwupgradedialog.h.

Referenced by FwUpgradeDialog::exec(), FwUpgradeDialog::fwUpdateFinished(), FwUpgradeDialog::fwUpdateProgress(), FwUpgradeDialog::on_actionButton_clicked(), and Ui_FwUpgradeDialog::setupUi().

12.15.3.15 QLabel* Ui_FwUpgradeDialog::statusLabel [inherited]

Definition at line 38 of file ui_fwupgradedialog.h.

Referenced by FwUpgradeDialog::exec(), FwUpgradeDialog::fwUpdateFinished(), FwUpgradeDialog::on_actionButton_clicked(), Ui_FwUpgradeDialog::retranslateUi(), and Ui_FwUpgradeDialog::setupUi().

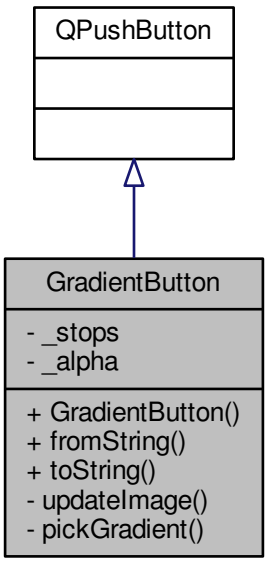
The documentation for this class was generated from the following file:

- src/ckb/ui_fwupgradedialog.h

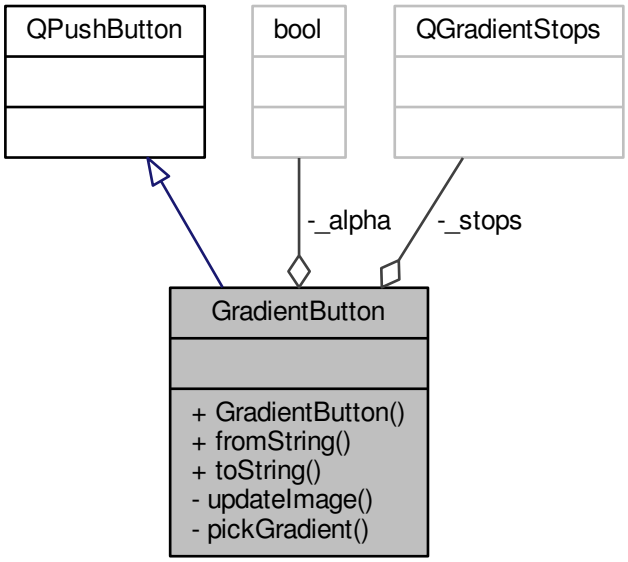
12.16 GradientButton Class Reference

`#include <src/ckb/gradientbutton.h>`

Inheritance diagram for GradientButton:



Collaboration diagram for GradientButton:



Signals

- void gradientChanged ()

Public Member Functions

- GradientButton (QWidget *parent=0, bool allowAlpha=false)
- void fromString (const QString &string)
- QString toString () const

Private Slots

- void pickGradient ()

Private Member Functions

- void updateImage ()

Private Attributes

- QGradientStops _stops
- bool _alpha

12.16.1 Detailed Description

Definition at line 6 of file gradientbutton.h.

12.16.2 Constructor & Destructor Documentation

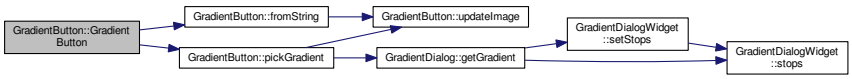
12.16.2.1 GradientButton::GradientButton (QWidget * *parent* = 0, bool *allowAlpha* = false) [explicit]

Definition at line 7 of file gradientbutton.cpp.

References fromString(), and pickGradient().

```
7                                                                   :
8     QPushButton(parent), _alpha(allowAlpha)
9 {
10     connect(this, SIGNAL(clicked()), this, SLOT(pickGradient()));
11     setAutoDefault(false);
12     setDefault(false);
13     fromString("");
14     setText("");
15 }
```

Here is the call graph for this function:



12.16.3 Member Function Documentation

12.16.3.1 void GradientButton::fromString (const QString & *string*)

Definition at line 38 of file gradientbutton.cpp.

References _alpha, _stops, and updateImage().

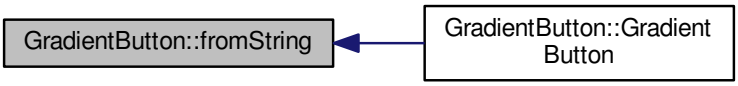
Referenced by GradientButton().

```
38                                                  {
39     // Clear existing stops
40     _stops.clear();
41     // Parse string with sscanf
42     QByteArray cString = string.toLatin1();
43     const char* data = cString.data();
44     char pos = -1;
45     uchar a, r, g, b;
46     while(1){
47         int scanned = 0;
48         char newpos;
49         if(sscanf(data, "%hhd:%2hhx%2hhx%2hhx%2hhx%n", &newpos, &a, &r, &g, &b, &scanned) != 5)
50             break;
51         data += scanned;
52         // Don't allow stops out-of-order or past 100
53         if(newpos <= pos || newpos > 100)
54             break;
55         pos = newpos;
56         if(!_alpha)
57             a = 255;
58         _stops.append(QGradientStop(pos / 100., QColor(r, g, b, a)));
59     }
60     if(_stops.count() == 0){
61         // If nothing was read, try a single ARGB constant.
62         if(sscanf(data, "%2hhx%2hhx%2hhx%2hhx", &a, &r, &g, &b) == 4){
63             _stops.append(QGradientStop(0., QColor(r, g, b, a)));
64             _stops.append(QGradientStop(1., QColor(r, g, b, 0.)));
65         }
66     }
67     // If that still didn't work, fill with white
68     if(_stops.count() == 0){
69         _stops.append(QGradientStop(0., QColor(255, 255, 255)));
70         _stops.append(QGradientStop(1., QColor(255, 255, 255)));
71     }
72     // Make sure there are stops at 0 and at 100
73     if(_stops[0].first != 0.)
74         _stops.prepend(QGradientStop(0., _stops[0].second));
```

```
75     if(_stops.last().first != 1.)
76         _stops.prepend(QGradientStop(1., _stops.last().second));
77     updateImage();
78 }
```

Here is the call graph for this function:

Here is the caller graph for this function:



12.16.3.2 void GradientButton::gradientChanged ( ) [signal]

Definition at line 132 of file moc_gradientbutton.cpp.

Referenced by pickGradient().

```
133 {
134     QMetaObject::activate(this, &staticMetaObject, 0, Q_NULLPTR);
135 }
```

Here is the caller graph for this function:

12.16.3.3 void GradientButton::pickGradient ( ) [private], [slot]

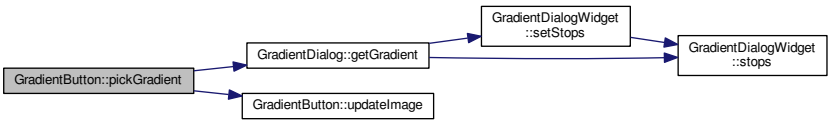
Definition at line 90 of file gradientbutton.cpp.

References _stops, GradientDialog::getGradient(), gradientChanged(), and updateImage().

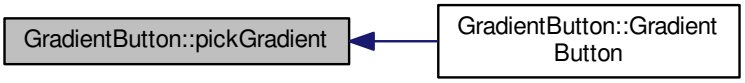
Referenced by GradientButton().

```
90                                  {
91     GradientDialog dialog(this);
92     _stops = dialog.getGradient(_stops);
93     updateImage();
94     emit gradientChanged();
95 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.16.3.4 QString GradientButton::toString ( ) const

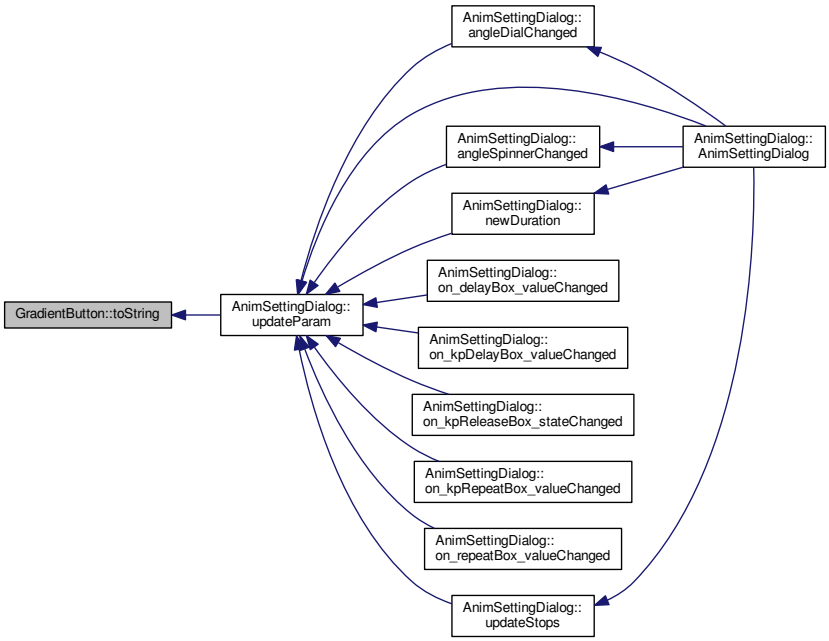
Definition at line 80 of file gradientbutton.cpp.

References _stops.

Referenced by AnimSettingDialog::updateParam().

```
80                                      {
81     QStringList result;
82     foreach(const QGradientStop& stop, _stops){
83         QString string;
84         const QColor& color = stop.second;
85         result << string.sprintf("%d:%02x%02x%02x%02x", (int)round(stop.first * 100.f), color.alpha(),
       color.red(), color.green(), color.blue());
86     }
87     return result.join(" ");
88 }
```

Here is the caller graph for this function:



12.16.3.5 void GradientButton::updateImage ( ) [private]

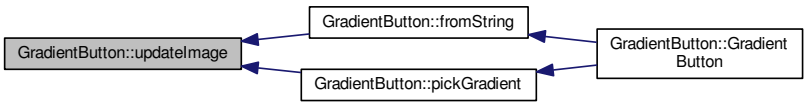
Definition at line 17 of file gradientbutton.cpp.

References _alpha, and _stops.

Referenced by fromString(), and pickGradient().

```
17                                {
18     const int w = 130, h = 16;
19     QImage image(w, h, QImage::Format_RGB888);
20     QPainter painter(&image);
21     painter.setPen(Qt::NoPen);
22     painter.fillRect(0, 0, w, h, QColor(0, 0, 0));
23     if(_alpha){
24         for(int x = 1; x < w - 1; x += h){
25             painter.fillRect(x, 1, h / 2, h / 2 - 1, QColor(255, 255, 255));
26             painter.fillRect(x + h / 2, 1, h / 2, h / 2 - 1, QColor(192, 192, 192));
27             painter.fillRect(x, h / 2, h / 2, h / 2 - 1, QColor(192, 192, 192));
28             painter.fillRect(x + h / 2, h / 2, h / 2, h / 2 - 1, QColor(255, 255, 255));
29         }
30     }
31     QLinearGradient gradient(1., 0., w - 1, 0.);
32     gradient.setStops(_stops);
33     painter.fillRect(1, 1, w - 2, h - 2, QBrush(gradient));
34     setIconSize(QSize(w, h));
35     setIcon(QIcon(QPixmap::fromImage(image)));
36 }
```

Here is the caller graph for this function:



12.16.4 Field Documentation

12.16.4.1 bool GradientButton::_alpha [private]

Definition at line 23 of file gradientbutton.h.

Referenced by fromString(), and updateImage().

12.16.4.2 QGradientStops GradientButton::_stops [private]

Definition at line 22 of file gradientbutton.h.

Referenced by fromString(), pickGradient(), toString(), and updateImage().

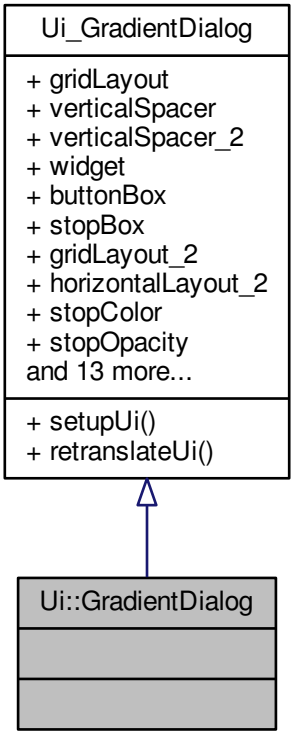
The documentation for this class was generated from the following files:

- src/ckb/gradientbutton.h
- src/ckb/gradientbutton.cpp
- src/ckb/release/moc_gradientbutton.cpp

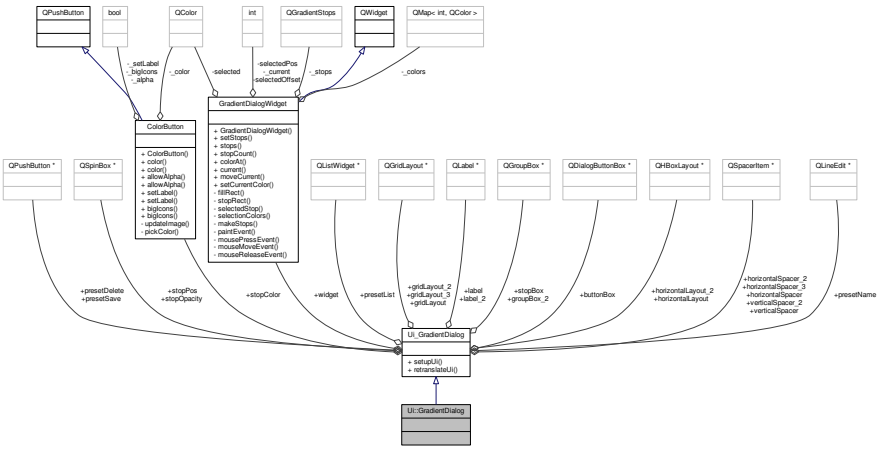
12.17 Ui::GradientDialog Class Reference

`#include <src/ckb/ui_gradientdialog.h>`

Inheritance diagram for Ui::GradientDialog:



Collaboration diagram for Ui::GradientDialog:



Public Member Functions

- void setupUi (QDialog *GradientDialog)
- void retranslateUi (QDialog *GradientDialog)

Data Fields

- QGridLayout * gridLayout
- QSpacerItem * verticalSpacer
- QSpacerItem * verticalSpacer_2
- GradientDialogWidget * widget
- QDialogButtonBox * buttonBox
- QGroupBox * stopBox
- QGridLayout * gridLayout_2
- QHBoxLayout * horizontalLayout_2
- ColorButton * stopColor
- QSpinBox * stopOpacity
- QSpacerItem * horizontalSpacer_2
- QLabel * label
- QHBoxLayout * horizontalLayout
- QSpinBox * stopPos
- QSpacerItem * horizontalSpacer
- QLabel * label_2
- QSpacerItem * horizontalSpacer_3
- QGroupBox * groupBox_2
- QGridLayout * gridLayout_3
- QListWidget * presetList
- QLineEdit * presetName
- QPushButton * presetSave
- QPushButton * presetDelete

12.17.1 Detailed Description

Definition at line 231 of file ui_gradientdialog.h.

12.17.2 Member Function Documentation

12.17.2.1 void Ui_GradientDialog::retranslateUi (QDialog * *GradientDialog*) [inline], [inherited]

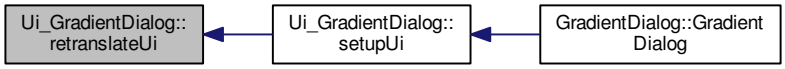
Definition at line 213 of file ui_gradientdialog.h.

References Ui_GradientDialog::groupBox_2, Ui_GradientDialog::label, Ui_GradientDialog::label_2, Ui_GradientDialog::presetDelete, Ui_GradientDialog::presetName, Ui_GradientDialog::presetSave, Ui_GradientDialog::stopBox, Ui_GradientDialog::stopColor, Ui_GradientDialog::stopOpacity, and Ui_GradientDialog::stopPos.

Referenced by Ui_GradientDialog::setupUi().

```
214     {
215         GradientDialog->setWindowTitle(QApplication::translate("GradientDialog", "Pick Gradient", Q_NULLPTR
    ));
216         stopBox->setTitle(QApplication::translate("GradientDialog", "Point X of Y", Q_NULLPTR));
217         stopColor->setText(QApplication::translate("GradientDialog", "Pick Color...", Q_NULLPTR));
218         stopOpacity->setSuffix(QApplication::translate("GradientDialog", "%", Q_NULLPTR));
219         label->setText(QApplication::translate("GradientDialog", "Position:", Q_NULLPTR));
220         stopPos->setSuffix(QApplication::translate("GradientDialog", "%", Q_NULLPTR));
221         label_2->setText(QApplication::translate("GradientDialog", "Color:", Q_NULLPTR));
222         groupBox_2->setTitle(QApplication::translate("GradientDialog", "Presets", Q_NULLPTR));
223         presetName->setPlaceholderText(QApplication::translate("GradientDialog", "Name",
    Q_NULLPTR));
224         presetSave->setText(QApplication::translate("GradientDialog", "Save", Q_NULLPTR));
225         presetDelete->setText(QApplication::translate("GradientDialog", "Delete", Q_NULLPTR));
226     } // retranslateUi
```

Here is the caller graph for this function:



12.17.2.2 void Ui_GradientDialog::setupUi (QDialog * *GradientDialog*) [inline],[inherited]

Definition at line 60 of file ui_gradientdialog.h.

References Ui_GradientDialog::buttonBox, Ui_GradientDialog::gridLayout, Ui_GradientDialog::gridLayout_2, Ui_GradientDialog::gridLayout_3, Ui_GradientDialog::groupBox_2, Ui_GradientDialog::horizontalLayout, Ui_GradientDialog::horizontalLayout_2, Ui_GradientDialog::horizontalSpacer, Ui_GradientDialog::horizontalSpacer_2, Ui_GradientDialog::horizontalSpacer_3, Ui_GradientDialog::label, Ui_GradientDialog::label_2, Ui_GradientDialog::presetDelete, Ui_GradientDialog::presetList, Ui_GradientDialog::presetName, Ui_GradientDialog::presetSave, Ui_GradientDialog::retranslateUi(), Ui_GradientDialog::stopBox, Ui_GradientDialog::stopColor, Ui_GradientDialog::stopOpacity, Ui_GradientDialog::stopPos, Ui_GradientDialog::verticalSpacer, Ui_GradientDialog::verticalSpacer_2, and Ui_GradientDialog::widget.

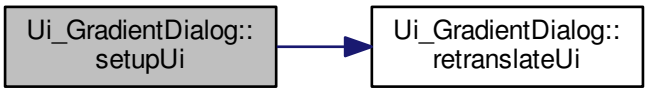
Referenced by GradientDialog::GradientDialog().

```
61     {
62         if (GradientDialog->objectName().isEmpty())
63             GradientDialog->setObjectName(QStringLiteral("GradientDialog"));
64         GradientDialog->resize(400, 440);
65         GradientDialog->setMinimumSize(QSize(400, 440));
66         GradientDialog->setMaximumSize(QSize(400, 440));
67         GradientDialog->setFocusPolicy(Qt::ClickFocus);
68         gridLayout = new QGridLayout(GradientDialog);
69         gridLayout->setObjectName(QStringLiteral("gridLayout"));
70         verticalSpacer = new QSpacerItem(0, 0, QSizePolicy::Minimum, QSizePolicy::Expanding);
71 
72         gridLayout->addItem(verticalSpacer, 1, 0, 1, 1);
73 
74         verticalSpacer_2 = new QSpacerItem(0, 15, QSizePolicy::Minimum, QSizePolicy::Fixed)
    ;
75 
76         gridLayout->addItem(verticalSpacer_2, 3, 0, 1, 1);
77 
78         widget = new GradientDialogWidget(GradientDialog);
79         widget->setObjectName(QStringLiteral("widget"));
80         QSizePolicy sizePolicy(QSizePolicy::Preferred, QSizePolicy::Fixed);
81         sizePolicy.setHorizontalStretch(0);
82         sizePolicy.setVerticalStretch(0);
83         sizePolicy.setHeightForWidth(widget->sizePolicy().hasHeightForWidth());
84         widget->setSizePolicy(sizePolicy);
85         widget->setMinimumSize(QSize(0, 74));
86         widget->setMaximumSize(QSize(16777215, 74));
87 
88         gridLayout->addWidget(widget, 2, 0, 1, 1);
89 
90         buttonBox = new QDialogButtonBox(GradientDialog);
91         buttonBox->setObjectName(QStringLiteral("buttonBox"));
92         buttonBox->setOrientation(Qt::Horizontal);
93         buttonBox->setStandardButtons(QDialogButtonBox::Cancel|QDialogButtonBox::Ok);
94 
95         gridLayout->addWidget(buttonBox, 5, 0, 1, 1);
96 
97         stopBox = new QGroupBox(GradientDialog);
98         stopBox->setObjectName(QStringLiteral("stopBox"));
99         sizePolicy.setHeightForWidth(stopBox->sizePolicy().hasHeightForWidth());
100         stopBox->setSizePolicy(sizePolicy);
101         gridLayout_2 = new QGridLayout(stopBox);
102         gridLayout_2->setObjectName(QStringLiteral("gridLayout_2"));
103         horizontalLayout_2 = new QHBoxLayout();
104         horizontalLayout_2->setObjectName(QStringLiteral("horizontalLayout_2"));
105         stopColor = new ColorButton(stopBox);
106         stopColor->setObjectName(QStringLiteral("stopColor"));
107 
```

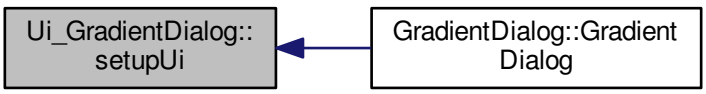
```
108         horizontalLayout_2->addWidget(stopColor);
109 
110         stopOpacity = new QSpinBox(stopBox);
111         stopOpacity->setObjectName(QStringLiteral("stopOpacity"));
112         stopOpacity->setMaximum(100);
113 
114         horizontalLayout_2->addWidget(stopOpacity);
115 
116         horizontalSpacer_2 = new QSpacerItem(40, 20, QSizePolicy::Expanding,
    QSizePolicy::Minimum);
117 
118         horizontalLayout_2->addItem(horizontalSpacer_2);
119 
120 
121         gridLayout_2->addLayout(horizontalLayout_2, 1, 2, 1, 1);
122 
123         label = new QLabel(stopBox);
124         label->setObjectName(QStringLiteral("label"));
125 
126         gridLayout_2->addWidget(label, 0, 0, 1, 1);
127 
128         horizontalLayout = new QHBoxLayout();
129         horizontalLayout->setObjectName(QStringLiteral("horizontalLayout"));
130         stopPos = new QSpinBox(stopBox);
131         stopPos->setObjectName(QStringLiteral("stopPos"));
132         stopPos->setMaximum(100);
133 
134         horizontalLayout->addWidget(stopPos);
135 
136         horizontalSpacer = new QSpacerItem(40, 20, QSizePolicy::Expanding,
    QSizePolicy::Minimum);
137 
138         horizontalLayout->addItem(horizontalSpacer);
139 
140 
141         gridLayout_2->addLayout(horizontalLayout, 0, 2, 1, 1);
142 
143         label_2 = new QLabel(stopBox);
144         label_2->setObjectName(QStringLiteral("label_2"));
145 
146         gridLayout_2->addWidget(label_2, 1, 0, 1, 1);
147 
148         horizontalSpacer_3 = new QSpacerItem(10, 20, QSizePolicy::Fixed,
    QSizePolicy::Minimum);
149 
150         gridLayout_2->addItem(horizontalSpacer_3, 0, 1, 1, 1);
151 
152 
153         gridLayout->addWidget(stopBox, 4, 0, 1, 1);
154 
155         groupBox_2 = new QGroupBox(GradientDialog);
156         groupBox_2->setObjectName(QStringLiteral("groupBox_2"));
157         sizePolicy.setHeightForWidth(groupBox_2->sizePolicy().hasHeightForWidth());
158         groupBox_2->setSizePolicy(sizePolicy);
159         gridLayout_3 = new QGridLayout(groupBox_2);
160         gridLayout_3->setObjectName(QStringLiteral("gridLayout_3"));
161         presetList = new QListWidget(groupBox_2);
162         presetList->setObjectName(QStringLiteral("presetList"));
163         QSizePolicy sizePolicy1(QSizePolicy::Expanding, QSizePolicy::Preferred);
164         sizePolicy1.setHorizontalStretch(0);
165         sizePolicy1.setVerticalStretch(0);
166         sizePolicy1.setHeightForWidth(presetList->sizePolicy().hasHeightForWidth());
167         presetList->setSizePolicy(sizePolicy1);
168         presetList->setFrameShadow(QFrame::Plain);
169         presetList->setHorizontalScrollBarPolicy(Qt::ScrollBarAlwaysOn);
170         presetList->setIconSize(QSize(64, 64));
171         presetList->setHorizontalScrollMode(QAbstractItemView::ScrollPerPixel);
172         presetList->setFlow(QListView::TopToBottom);
173         presetList->setViewMode(QListView::IconMode);
174         presetList->setUniformItemSizes(true);
175 
176         gridLayout_3->addWidget(presetList, 0, 0, 3, 1);
177 
178         presetName = new QLineEdit(groupBox_2);
179         presetName->setObjectName(QStringLiteral("presetName"));
180         presetName->setMaxLength(30);
181 
182         gridLayout_3->addWidget(presetName, 0, 1, 1, 1);
183 
184         presetSave = new QPushButton(groupBox_2);
185         presetSave->setObjectName(QStringLiteral("presetSave"));
186         presetSave->setAutoDefault(false);
187 
188         gridLayout_3->addWidget(presetSave, 1, 1, 1, 1);
189 
190         presetDelete = new QPushButton(groupBox_2);
191         presetDelete->setObjectName(QStringLiteral("presetDelete"));
```

```
192         presetDelete->setAutoDefault(false);
193
194         gridLayout_3->addWidget(presetDelete, 2, 1, 1, 1);
195
196
197         gridLayout->addWidget(groupBox_2, 0, 0, 1, 1);
198
199         QWidget::setTabOrder(presetList, presetName);
200         QWidget::setTabOrder(presetName, presetSave);
201         QWidget::setTabOrder(presetSave, presetDelete);
202         QWidget::setTabOrder(presetDelete, stopPos);
203         QWidget::setTabOrder(stopPos, stopColor);
204         QWidget::setTabOrder(stopColor, stopOpacity);
205
206         retranslateUi(GradientDialog);
207         QObject::connect(buttonBox, SIGNAL(accepted()), GradientDialog, SLOT(accept()));
208         QObject::connect(buttonBox, SIGNAL(rejected()), GradientDialog, SLOT(reject()));
209
210         QMetaObject::connectSlotsByName(GradientDialog);
211     } // setupUi
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.17.3 Field Documentation

12.17.3.1 QDialogButtonBox* Ui_GradientDialog::buttonBox [inherited]

Definition at line 40 of file ui_gradientdialog.h.

Referenced by Ui_GradientDialog::setupUi().

12.17.3.2 QGridLayout* Ui_GradientDialog::gridLayout [inherited]

Definition at line 36 of file ui_gradientdialog.h.

Referenced by Ui_GradientDialog::setupUi().

12.17.3.3 QGridLayout* Ui_GradientDialog::gridLayout_2 [inherited]

Definition at line 42 of file ui_gradientdialog.h.

Referenced by Ui_GradientDialog::setupUi().

12.17.3.4 QGridLayout* Ui_GradientDialog::gridLayout_3 [inherited]

Definition at line 54 of file ui_gradientdialog.h.

Referenced by Ui_GradientDialog::setupUi().

12.17.3.5 QGroupBox* Ui_GradientDialog::groupBox_2 [inherited]

Definition at line 53 of file ui_gradientdialog.h.

Referenced by Ui_GradientDialog::retranslateUi(), and Ui_GradientDialog::setupUi().

12.17.3.6 QHBoxLayout* Ui_GradientDialog::horizontalLayout [inherited]

Definition at line 48 of file ui_gradientdialog.h.

Referenced by Ui_GradientDialog::setupUi().

12.17.3.7 QHBoxLayout* Ui_GradientDialog::horizontalLayout_2 [inherited]

Definition at line 43 of file ui_gradientdialog.h.

Referenced by Ui_GradientDialog::setupUi().

12.17.3.8 QSpacerItem* Ui_GradientDialog::horizontalSpacer [inherited]

Definition at line 50 of file ui_gradientdialog.h.

Referenced by Ui_GradientDialog::setupUi().

12.17.3.9 QSpacerItem* Ui_GradientDialog::horizontalSpacer_2 [inherited]

Definition at line 46 of file ui_gradientdialog.h.

Referenced by Ui_GradientDialog::setupUi().

12.17.3.10 QSpacerItem* Ui_GradientDialog::horizontalSpacer_3 [inherited]

Definition at line 52 of file ui_gradientdialog.h.

Referenced by Ui_GradientDialog::setupUi().

12.17.3.11 QLabel* Ui_GradientDialog::label [inherited]

Definition at line 47 of file ui_gradientdialog.h.

Referenced by Ui_GradientDialog::retranslateUi(), and Ui_GradientDialog::setupUi().

12.17.3.12 QLabel* Ui_GradientDialog::label_2 [inherited]

Definition at line 51 of file ui_gradientdialog.h.

Referenced by Ui_GradientDialog::retranslateUi(), and Ui_GradientDialog::setupUi().

12.17.3.13 QPushButton* Ui_GradientDialog::presetDelete [inherited]

Definition at line 58 of file ui_gradientdialog.h.

Referenced by GradientDialog::getGradient(), GradientDialog::on_presetDelete_clicked(), GradientDialog::on_presetSave_clicked(), Ui_GradientDialog::retranslateUi(), GradientDialog::setPreset(), and Ui_GradientDialog::setupUi().

12.17.3.14 QListWidget* Ui_GradientDialog::presetList [inherited]

Definition at line 55 of file ui_gradientdialog.h.

Referenced by GradientDialog::setPreset(), Ui_GradientDialog::setupUi(), and GradientDialog::updatePresets().

12.17.3.15 QLineEdit* Ui_GradientDialog::presetName [inherited]

Definition at line 56 of file ui_gradientdialog.h.

Referenced by GradientDialog::on_presetSave_clicked(), Ui_GradientDialog::retranslateUi(), GradientDialog::setPreset(), and Ui_GradientDialog::setupUi().

12.17.3.16 QPushButton* Ui_GradientDialog::presetSave [inherited]

Definition at line 57 of file ui_gradientdialog.h.

Referenced by GradientDialog::getGradient(), GradientDialog::on_presetDelete_clicked(), GradientDialog::on_presetSave_clicked(), Ui_GradientDialog::retranslateUi(), GradientDialog::setPreset(), and Ui_GradientDialog::setupUi().

12.17.3.17 QGroupBox* Ui_GradientDialog::stopBox [inherited]

Definition at line 41 of file ui_gradientdialog.h.

Referenced by GradientDialog::currentChanged(), Ui_GradientDialog::retranslateUi(), and Ui_GradientDialog::setupUi().

12.17.3.18 ColorButton* Ui_GradientDialog::stopColor [inherited]

Definition at line 44 of file ui_gradientdialog.h.

Referenced by GradientDialog::currentChanged(), GradientDialog::GradientDialog(), GradientDialog::on_stopOpacity_valueChanged(), Ui_GradientDialog::retranslateUi(), and Ui_GradientDialog::setupUi().

12.17.3.19 QSpinBox* Ui_GradientDialog::stopOpacity [inherited]

Definition at line 45 of file ui_gradientdialog.h.

Referenced by GradientDialog::colorChanged(), GradientDialog::currentChanged(), Ui_GradientDialog::retranslateUi(), and Ui_GradientDialog::setupUi().

12.17.3.20 QSpinBox* Ui_GradientDialog::stopPos [inherited]

Definition at line 49 of file ui_gradientdialog.h.

Referenced by GradientDialog::currentChanged(), GradientDialog::on_stopPos_valueChanged(), Ui_GradientDialog::retranslateUi(), and Ui_GradientDialog::setupUi().

12.17.3.21 QSpacerItem* Ui_GradientDialog::verticalSpacer [inherited]

Definition at line 37 of file ui_gradientdialog.h.

Referenced by Ui_GradientDialog::setupUi().

12.17.3.22 QSpacerItem* Ui_GradientDialog::verticalSpacer_2 [inherited]

Definition at line 38 of file ui_gradientdialog.h.

Referenced by Ui_GradientDialog::setupUi().

12.17.3.23 GradientDialogWidget* Ui_GradientDialog::widget [inherited]

Definition at line 39 of file ui_gradientdialog.h.

Referenced by GradientDialog::colorChanged(), GradientDialog::currentChanged(), GradientDialog::getGradient(), GradientDialog::GradientDialog(), GradientDialog::on_presetSave_clicked(), GradientDialog::on_stopPos_valueChanged(), GradientDialog::setPreset(), and Ui_GradientDialog::setupUi().

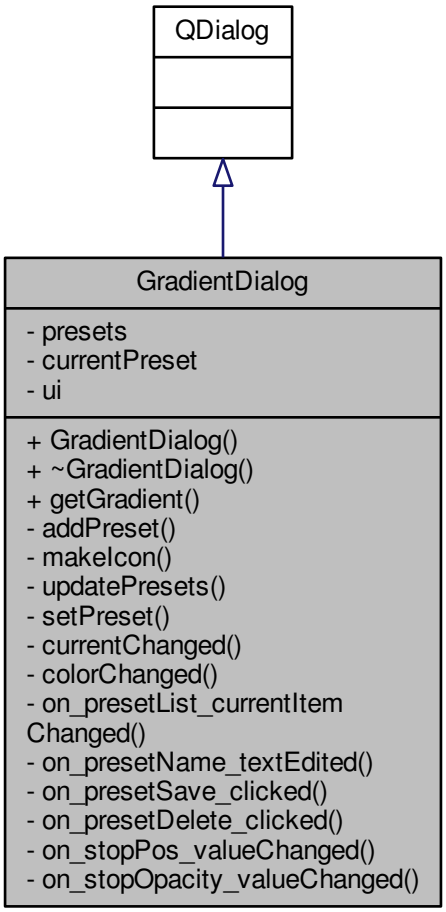
The documentation for this class was generated from the following file:

- src/ckb/ui_gradientdialog.h

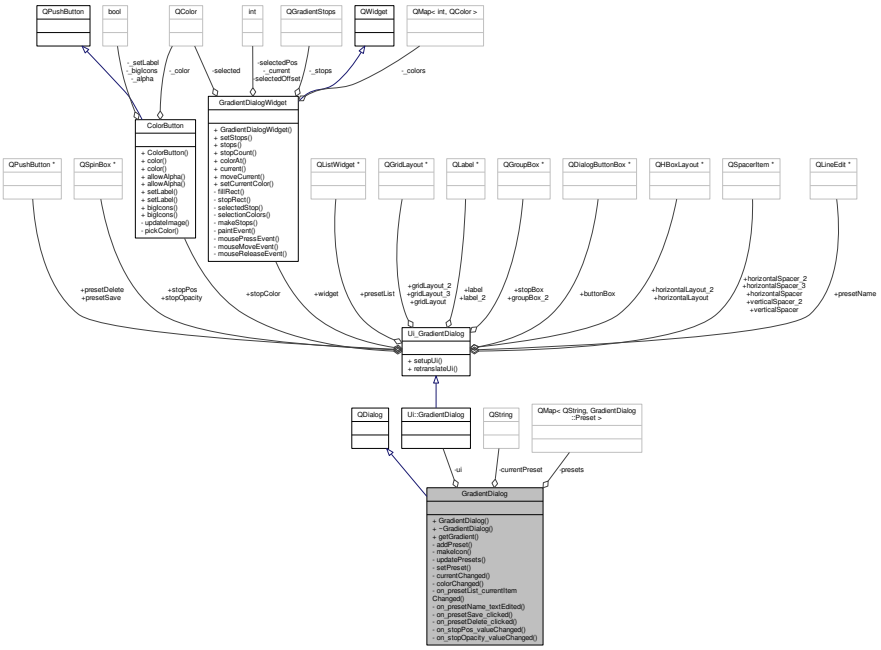
12.18 GradientDialog Class Reference

`#include <src/ckb/gradientdialog.h>`

Inheritance diagram for GradientDialog:



Collaboration diagram for GradientDialog:



Data Structures

- struct Preset

Public Member Functions

- GradientDialog (QWidget *parent=0)
- ~GradientDialog ()
- QGradientStops getGradient (const QGradientStops &prevGradient)

Private Slots

- void currentChanged (QColor color, bool spontaneous, int position)
- void colorChanged (QColor color)
- void on_presetList_currentItemChanged (QListWidgetItem *current, QListWidgetItem *previous)
- void on_presetName_textEdited (const QString &arg1)
- void on_presetSave_clicked ()
- void on_presetDelete_clicked ()
- void on_stopPos_valueChanged (int arg1)
- void on_stopOpacity_valueChanged (int arg1)

Private Member Functions

- void addPreset (const Preset &preset)
- QIcon makeIcon (const Preset &preset)
- void updatePresets ()
- void setPreset (const QString &newPreset)

Private Attributes

- QMap< QString, Preset > presets
- QString currentPreset
- Ui::GradientDialog * ui

12.18.1 Detailed Description

Definition at line 12 of file gradientdialog.h.

12.18.2 Constructor & Destructor Documentation

12.18.2.1 GradientDialog::GradientDialog (QWidget * *parent* = 0) [explicit]

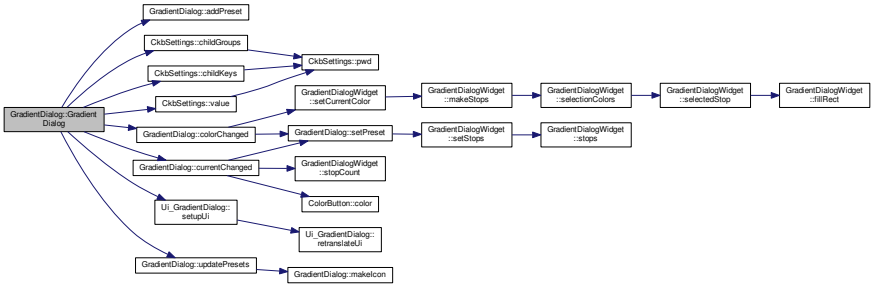
Definition at line 11 of file gradientdialog.cpp.

References addPreset(), CkbSettings::childGroups(), CkbSettings::childKeys(), colorChanged(), currentChanged(), GradientDialog::Preset::gradient, prefsPath, presets, Ui_GradientDialog::setupUi(), Ui_GradientDialog::stopColor, ui, updatePresets(), CkbSettings::value(), and Ui_GradientDialog::widget.

```
11                                                  :
12     QDialog(parent),
13     ui(new Ui::GradientDialog)
14 {
15     ui->setupUi(this);
16     connect(ui->widget, SIGNAL(currentChanged(QColor,bool,int)), this, SLOT(
       currentChanged(QColor,bool,int)));
17     connect(ui->stopColor, SIGNAL(colorChanged(QColor)), this, SLOT(
       colorChanged(QColor)));
18
19     // Add built-in presets
20     Preset blackAndWhite("B & W", true);
21     blackAndWhite.gradient.append(QGradientStop(0., QColor(0, 0, 0)));
22     blackAndWhite.gradient.append(QGradientStop(1., QColor(255, 255, 255)));
23     Preset fade("Fade", true);
24     fade.gradient.append(QGradientStop(0., QColor(255, 255, 255)));
25     fade.gradient.append(QGradientStop(1., QColor(255, 255, 255, 0)));
26     Preset rainbow("Rainbow", true);
27     rainbow.gradient.append(QGradientStop(0.00, QColor(255, 0, 0)));
28     rainbow.gradient.append(QGradientStop(0.17, QColor(255, 255, 0)));
29     rainbow.gradient.append(QGradientStop(0.33, QColor(0, 255, 0)));
30     rainbow.gradient.append(QGradientStop(0.50, QColor(0, 255, 255)));
31     rainbow.gradient.append(QGradientStop(0.67, QColor(0, 0, 255)));
32     rainbow.gradient.append(QGradientStop(0.83, QColor(255, 0, 255)));
33     rainbow.gradient.append(QGradientStop(1.00, QColor(255, 0, 0)));
34     addPreset(blackAndWhite);
35     addPreset(fade);
36     addPreset(rainbow);
37
38     // Load stored presets
39     CkbSettings settings(prefsPath);
40     foreach(const QString& name, settings.childGroups()){
41         QString pName = name.toLower();
42         if(presets.contains(pName))
43             continue;
44         SGroup group(settings, name);
45         // Create keys in a map to ensure correct order and no duplicates
46         QMap<int, QColor> colors;
47         foreach(const QString& position, settings.childKeys()){
48             int pos = position.toInt();
49             if(pos < 0 || pos > 100)
50                 continue;
51             colors[pos] = settings.value(position).value<QColor>();
52         }
53         if(colors.count() < 2)
54             continue;
55         Preset preset(name);
56         QMapIterator<int, QColor> i(colors);
57         while(i.hasNext()){
58             i.next();
59             preset.gradient.append(QGradientStop(i.key() / 100., i.value()));
60         }
61         presets[pName] = preset;
62     }
63
```

```
64     updatePresets();
65 }
```

Here is the call graph for this function:



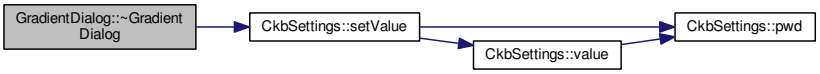
12.18.2.2 GradientDialog::~GradientDialog ( )

Definition at line 77 of file gradientdialog.cpp.

References GradientDialog::Preset::builtIn, GradientDialog::Preset::gradient, GradientDialog::Preset::name, prefsPath, presets, CkbSettings::setValue(), and ui.

```
77                                {
78     // Save presets
79     CkbSettings settings(prefsPath, true);
80     QMapIterator<QString, Preset> i(presets);
81     while(i.hasNext()){
82         Preset preset = i.next().value();
83         if(!preset.builtIn){
84             SGroup group(settings, preset.name);
85             foreach(const QGradientStop& stop, preset.gradient)
86                 settings.setValue(QString::number((int)(stop.first * 100.)), stop.second);
87         }
88     }
89     delete ui;
90 }
```

Here is the call graph for this function:



12.18.3 Member Function Documentation

12.18.3.1 void GradientDialog::addPreset (const Preset & *preset*) [inline], [private]

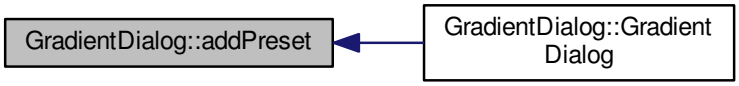
Definition at line 34 of file gradientdialog.h.

References GradientDialog::Preset::name, and presets.

Referenced by GradientDialog().

```
34 { presets[preset.name.toLower()] = preset; }
```

Here is the caller graph for this function:



12.18.3.2 void GradientDialog::colorChanged (QColor *color*) [private],[slot]

Definition at line 234 of file gradientdialog.cpp.

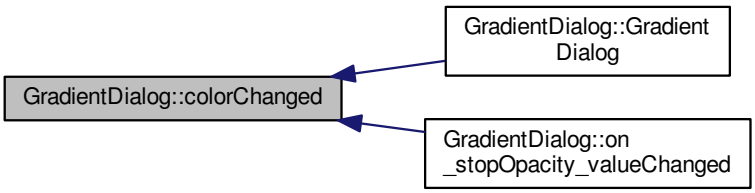
References GradientDialogWidget::setCurrentColor(), setPreset(), Ui_GradientDialog::stopOpacity, ui, and Ui_GradientDialog::widget.

Referenced by GradientDialog(), and on_stopOpacity_valueChanged().

```
234                                          {
235     color.setAlphaF(ui->stopOpacity->value() / 100.f);
236     ui->widget->setCurrentColor(color);
237     setPreset("");
238 }
```

Here is the call graph for this function:

Here is the caller graph for this function:



12.18.3.3 void GradientDialog::currentChanged (QColor *color,* bool *spontaneous,* int *position*) [private],[slot]

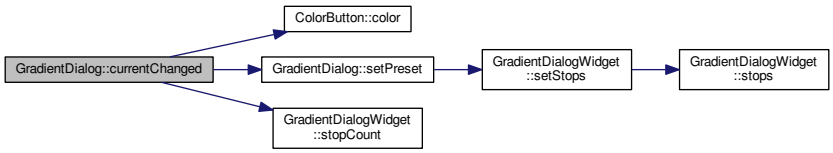
Definition at line 128 of file gradientdialog.cpp.

References ColorButton::color(), setPreset(), Ui_GradientDialog::stopBox, Ui_GradientDialog::stopColor, GradientDialogWidget::stopCount(), Ui_GradientDialog::stopOpacity, Ui_GradientDialog::stopPos, ui, and Ui_GradientDialog::widget.

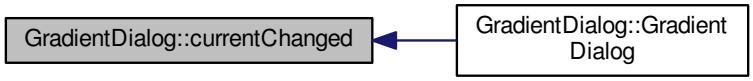
Referenced by GradientDialog().

```
128                                                                                 {
129     int pCount = ui->widget->stopCount();
130     if(pCount == 1)
131         ui->stopBox->setTitle("1 point");
132     else
133         ui->stopBox->setTitle(QString("%1 points").arg(pCount));
134     if(!color.isValid()){
135         ui->stopBox->setEnabled(false);
136         ui->stopPos->setValue(0);
137         ui->stopColor->color(QColor(0, 0, 0));
138         ui->stopOpacity->setValue(0);
139     } else {
140         ui->stopBox->setEnabled(true);
141         ui->stopPos->setValue(position);
142         ui->stopColor->color(color);
143         ui->stopOpacity->setValue(round(color.alphaF() * 100.f));
144     }
145     // Un-set current preset
146     if(spontaneous)
147         setPreset("");
148     setFocus();
149 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.18.3.4 QGradientStops GradientDialog::getGradient (const QGradientStops & *prevGradient*)

Definition at line 67 of file gradientdialog.cpp.

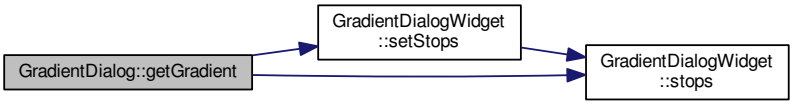
References Ui_GradientDialog::presetDelete, Ui_GradientDialog::presetSave, GradientDialogWidget::setStops(), GradientDialogWidget::stops(), ui, and Ui_GradientDialog::widget.

Referenced by GradientButton::pickGradient().

```
67                                                                                  {
68     ui->presetDelete->setEnabled(false);
69     ui->presetSave->setEnabled(true);
70     ui->widget->setStops(prevGradient);
```

```
71     exec();
72     if(result() != QDialog::Accepted)
73         return prevGradient;
74     return ui->widget->stops();
75 }
```

Here is the call graph for this function:



Here is the caller graph for this function:

12.18.3.5 QIcon GradientDialog::makeIcon (const Preset & *preset*) [private]

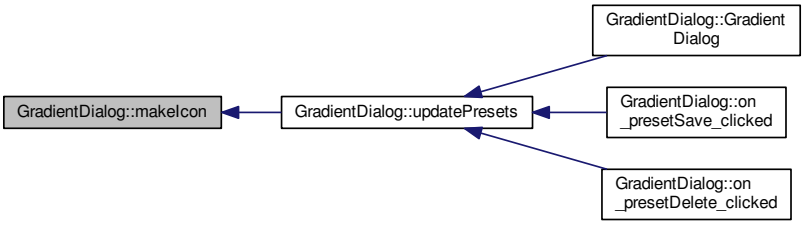
Definition at line 92 of file gradientdialog.cpp.

References GradientDialog::Preset::gradient.

Referenced by updatePresets().

```
92                                                  {
93     // Paint gradient into a square
94     int w = 60, h = 60;
95     QImage image(w, h, QImage::Format_RGB888);
96     QPainter painter(&image);
97     // Draw background
98     for(int x = 0; x < w; x += 30){
99         for(int y = 0; y < h; y += 30){
100             painter.fillRect(x, y, 15, 15, QColor(255, 255, 255));
101             painter.fillRect(x + 15, y, 15, 15, QColor(192, 192, 192));
102             painter.fillRect(x, y + 15, 15, 15, QColor(192, 192, 192));
103             painter.fillRect(x + 15, y + 15, 15, 15, QColor(255, 255, 255));
104         }
105     }
106     // Draw gradient
107     painter.setPen(QColor(0, 0, 0));
108     QLinearGradient gradient(1., 1., w - 1., h - 1.);
109     gradient.setStops(preset.gradient);
110     painter.setBrush(gradient);
111     painter.drawRect(0, 0, w - 1, h - 1);
112     return QPixmap::fromImage(image);
113 }
```

Here is the caller graph for this function:



12.18.3.6 void GradientDialog::on_presetDelete_clicked ( ) [private], [slot]

Definition at line 211 of file gradientdialog.cpp.

References GradientDialog::Preset::builtIn, currentPreset, GradientDialog::Preset::name, Ui_GradientDialog::presetDelete, presets, Ui_GradientDialog::presetSave, ui, and updatePresets().

```
211                                          {
212     Preset current = presets.value(currentPreset);
213     if(current.name == "" || current.builtIn)
214         return;
215     if(QMessageBox::warning(this, "Warning", QString("Delete preset \"%1\"?").arg(current.name),
      QMessageBox::StandardButtons(QMessageBox::Yes | QMessageBox::No)) != QMessageBox::Yes){
216         setFocus();
217         return;
218     }
219     presets.remove(currentPreset);
220     currentPreset = "";
221     ui->presetDelete->setEnabled(false);
222     ui->presetSave->setEnabled(true);
223     updatePresets();
224     setFocus();
225 }
```

Here is the call graph for this function:

12.18.3.7 void GradientDialog::on_presetList_currentItemChanged (QListWidgetItem * *current*, QListWidgetItem * *previous*) [private], [slot]

Definition at line 174 of file gradientdialog.cpp.

References setPreset().

```
174                                                                                                 {
175     setPreset(current ? current->text().toLower() : "");
176 }
```

Here is the call graph for this function:

12.18.3.8 void GradientDialog::on_presetName_textEdited (const QString & *arg1*) [private],[slot]

Definition at line 178 of file gradientdialog.cpp.

References currentPreset, presets, and setPreset().

```
178                                                                 {
179     if(currentPreset != "" && arg1.trimmed() != presets[
      currentPreset].name)
180         setPreset("");
181 }
```

Here is the call graph for this function:

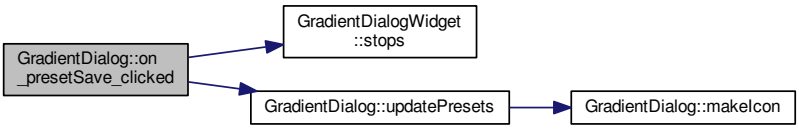
12.18.3.9 void GradientDialog::on_presetSave_clicked ( ) [private],[slot]

Definition at line 183 of file gradientdialog.cpp.

References GradientDialog::Preset::builtIn, currentPreset, GradientDialog::Preset::name, Ui_GradientDialog-::presetDelete, Ui_GradientDialog::presetName, presets, Ui_GradientDialog::presetSave, GradientDialogWidget-::stops(), ui, updatePresets(), and Ui_GradientDialog::widget.

```
183                                         {
184     if(currentPreset != "")
185         return;
186     QString name = ui->presetName->text().trimmed();
187     if(name == "")
188         ui->presetName->setText(name = "Custom");
189     QString pName = name.toLower();
190     // Make sure not to overwrite a built-in preset
191     Preset previous = presets[pName];
192     if(previous.builtIn){
193         QMessageBox::warning(this, "Error", "Can't overwrite a built-in preset. Please choose a different
       name.");
194         setFocus();
195         return;
196     } else if(previous.name != ""){
197         // Warn when overwriting an existing preset
198         if(QMessageBox::warning(this, "Warning", QString("Preset \"%1\" already exists. Replace?").arg(name
      ), QMessageBox::StandardButtons(QMessageBox::Save | QMessageBox::Cancel)) != QMessageBox::Save){
199             setFocus();
200             return;
201         }
202     }
203     presets[pName] = Preset(name, ui->widget->stops());
204     currentPreset = pName;
205     ui->presetDelete->setEnabled(true);
206     ui->presetSave->setEnabled(false);
207     updatePresets();
208     setFocus();
209 }
```

Here is the call graph for this function:



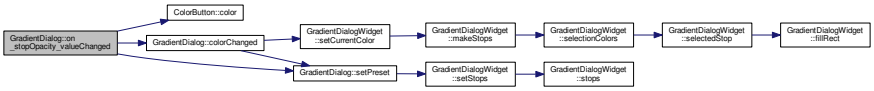
12.18.3.10 void GradientDialog::on_stopOpacity_valueChanged (int *arg1*) [private], [slot]

Definition at line 240 of file gradientdialog.cpp.

References ColorButton::color(), colorChanged(), setPreset(), Ui_GradientDialog::stopColor, and ui.

```
240                                                     {
241     colorChanged(ui->stopColor->color());
242     setPreset("");
243 }
```

Here is the call graph for this function:



12.18.3.11 void GradientDialog::on_stopPos_valueChanged (int *arg1*) [private], [slot]

Definition at line 227 of file gradientdialog.cpp.

References GradientDialogWidget::moveCurrent(), setPreset(), Ui_GradientDialog::stopPos, ui, and Ui_GradientDialog::widget.

```
227                                                     {
228     int res = ui->widget->moveCurrent(arg1);
229     if(res != arg1)
230         ui->stopPos->setValue(res);
231     setPreset("");
232 }
```

Here is the call graph for this function:

12.18.3.12 void GradientDialog::setPreset (const QString & *newPreset*) [private]

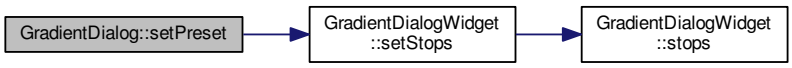
Definition at line 151 of file gradientdialog.cpp.

References GradientDialog::Preset::builtIn, currentPreset, GradientDialog::Preset::gradient, GradientDialog::Preset::name, Ui_GradientDialog::presetDelete, Ui_GradientDialog::presetList, Ui_GradientDialog::presetName, presets, Ui_GradientDialog::presetSave, GradientDialogWidget::setStops(), ui, and Ui_GradientDialog::widget.

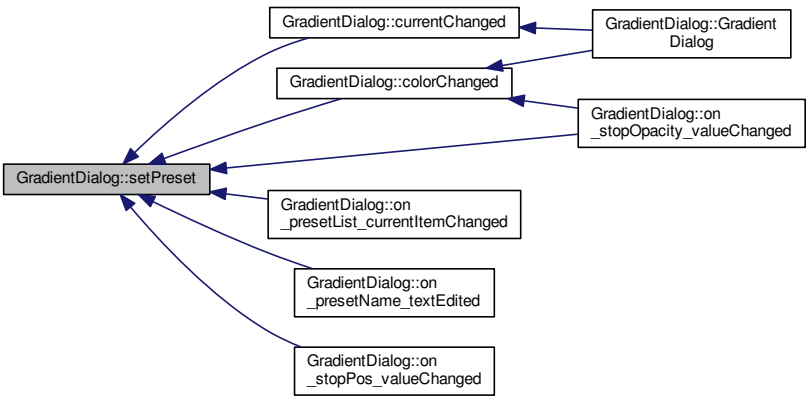
Referenced by colorChanged(), currentChanged(), on_presetList_currentItemChanged(), on_presetName_textEdited(), on_stopOpacity_valueChanged(), and on_stopPos_valueChanged().

```
151                                                 {
152     if(currentPreset == newPreset)
153         return;
154     if(newPreset == ""){
155         ui->presetList->clearSelection();
156         ui->presetList->setCurrentItem(0);
157         Preset current = presets.value(currentPreset);
158         QString curName = ui->presetName->text().trimmed();
159         if(curName == "" || (curName == current.name && current.builtIn))
160             ui->presetName->setText("Custom");
161         ui->presetDelete->setEnabled(false);
162         ui->presetSave->setEnabled(true);
163     } else {
164         Preset current = presets.value(newPreset);
165         ui->presetName->setText(current.name);
166         ui->presetDelete->setEnabled(!current.builtIn);
167         ui->presetSave->setEnabled(false);
168         ui->widget->setStops(current.gradient);
169     }
170     currentPreset = newPreset;
171     setFocus();
172 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.18.3.13 void GradientDialog::updatePresets ( ) [private]

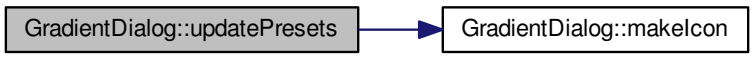
Definition at line 115 of file gradientdialog.cpp.

References currentPreset, makeIcon(), GradientDialog::Preset::name, Ui_GradientDialog::presetList, presets, and ui.

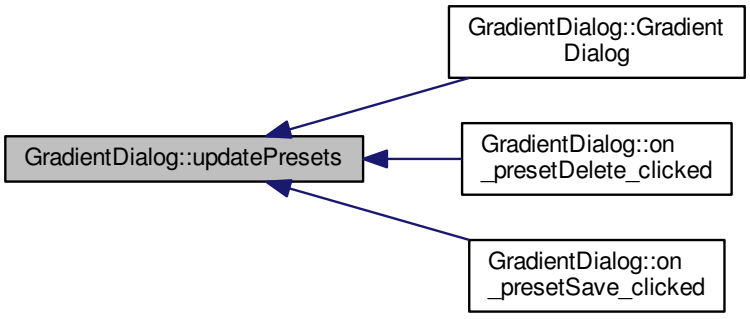
Referenced by GradientDialog(), on_presetDelete_clicked(), and on_presetSave_clicked().

```
115                                {
116     ui->presetList->clear();
117     QMapIterator<QString, Preset> i(presets);
118     while(i.hasNext()){
119         i.next();
120         const Preset& preset = i.value();
121         QListWidgetItem* item = new QListWidgetItem(makeIcon(preset), preset.name,
    ui->presetList);
122         if(currentPreset == i.key())
123             item->setSelected(true);
124         ui->presetList->addItem(item);
125     }
126 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.18.4 Field Documentation

12.18.4.1 QString GradientDialog::currentPreset [private]

Definition at line 32 of file gradientdialog.h.

Referenced by on_presetDelete_clicked(), on_presetName_textEdited(), on_presetSave_clicked(), setPreset(), and updatePresets().

12.18.4.2 QMap<QString, Preset> GradientDialog::presets [private]

Definition at line 31 of file gradientdialog.h.

Referenced by addPreset(), GradientDialog(), on_presetDelete_clicked(), on_presetName_textEdited(), on_presetSave_clicked(), setPreset(), updatePresets(), and ~GradientDialog().

12.18.4.3 Ui::GradientDialog* GradientDialog::ui [private]

Definition at line 39 of file gradientdialog.h.

Referenced by colorChanged(), currentChanged(), getGradient(), GradientDialog(), on_presetDelete_clicked(), on_presetSave_clicked(), on_stopOpacity_valueChanged(), on_stopPos_valueChanged(), setPreset(), updatePresets(), and ~GradientDialog().

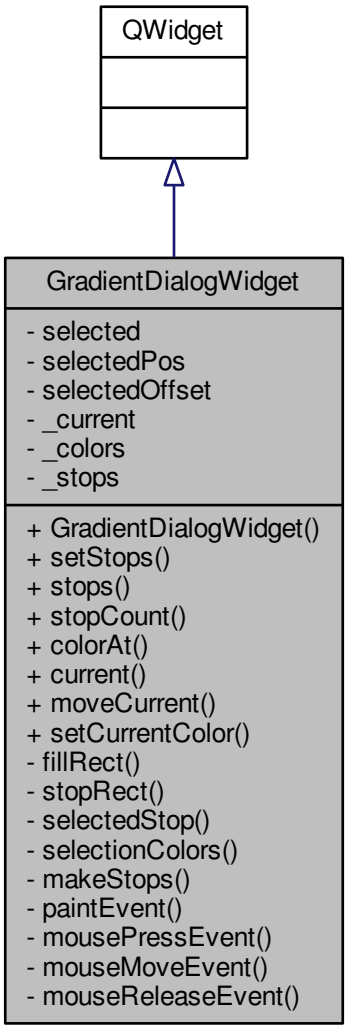
The documentation for this class was generated from the following files:

- src/ckb/gradientdialog.h
- src/ckb/gradientdialog.cpp

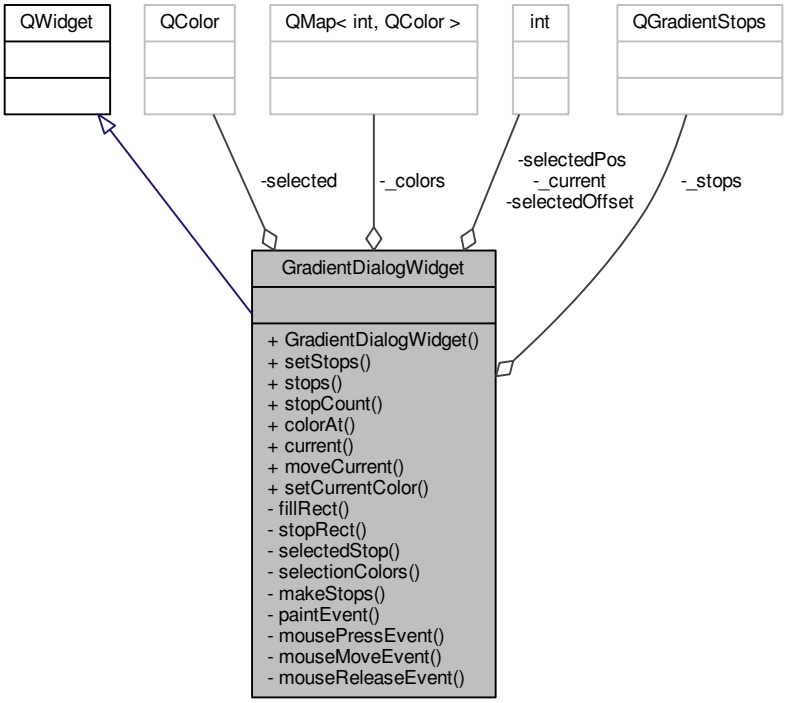
12.19 GradientDialogWidget Class Reference

`#include <src/ckb/gradientdialogwidget.h>`

Inheritance diagram for GradientDialogWidget:



Collaboration diagram for GradientDialogWidget:



Signals

- void currentChanged (QColor color, bool spontaneous, int position)

Public Member Functions

- GradientDialogWidget (QWidget ∗parent=0)
- void setStops (const QGradientStops &stops)
- QGradientStops stops ()
- int stopCount ()
- QColor colorAt (int position)
- int current ()
- int moveCurrent (int to)
- void setCurrentColor (const QColor &color)

Private Member Functions

- QRect fillRect ()
- QRect stopRect (int pos)
- int selectedStop ()
- QMap< int, QColor > selectionColors ()
- void makeStops ()
- void paintEvent (QPaintEvent ∗)
- void mousePressEvent (QMouseEvent ∗event)
- void mouseMoveEvent (QMouseEvent ∗event)
- void mouseReleaseEvent (QMouseEvent ∗event)

Private Attributes

- QColor selected
- int selectedPos
- int selectedOffset
- int _current
- QMap< int, QColor > _colors
- QGradientStops _stops

12.19.1 Detailed Description

Definition at line 8 of file gradientdialogwidget.h.

12.19.2 Constructor & Destructor Documentation

12.19.2.1 GradientDialogWidget::GradientDialogWidget (QWidget * *parent* = 0) [explicit]

Definition at line 6 of file gradientdialogwidget.cpp.

```
6                                                                   :
7     QWidget(parent), selectedPos(-1), _current(-1)
8 {
9     setMouseTracking(true);
10 }
```

12.19.3 Member Function Documentation

12.19.3.1 QColor GradientDialogWidget::colorAt (int *position*)

Definition at line 65 of file gradientdialogwidget.cpp.

References _stops.

Referenced by mousePressEvent().

```
65                                                {
66     // Find which stops the position is between
67     qreal pos = position / 100.;
68     QVectorIterator<QGradientStop> i(_stops);
69     QGradientStop previous = i.next();
70     while(i.hasNext()){
71         QGradientStop next = i.next();
72         if(next.first >= pos){
73             // Get color by linear interpolation. Premultiply the alpha value so that it returns the
       expected color
74             // (i.e. stops with zero opacity won't contribute to color)
75             QColor c1 = next.second, c2 = previous.second;
76             qreal distance = next.first - previous.first;
77             qreal dx = (pos - previous.first) / distance;
78             qreal a1 = c1.alphaF(), a2 = c2.alphaF();
79             qreal a3 = a1 * dx + a2 * (1.f - dx);
80             if(a3 == 0.){
81                 c2.setAlpha(0);
82                 return c2;
83             }
84             qreal r = (c1.redF() * a1 * dx + c2.redF() * a2 * (1.f - dx)) / a3;
85             qreal g = (c1.greenF() * a1 * dx + c2.greenF() * a2 * (1.f - dx)) / a3;
86             qreal b = (c1.blueF() * a1 * dx + c2.blueF() * a2 * (1.f - dx)) / a3;
87             return QColor::fromRgbF(r, g, b, a3);
88         }
89         previous = next;
90     }
91     return QColor();
92 }
```

Here is the caller graph for this function:

12.19.3.2 int GradientDialogWidget::current ( ) [inline]

Definition at line 20 of file gradientdialogwidget.h.

References _current.

```
20 { return _current; }
```

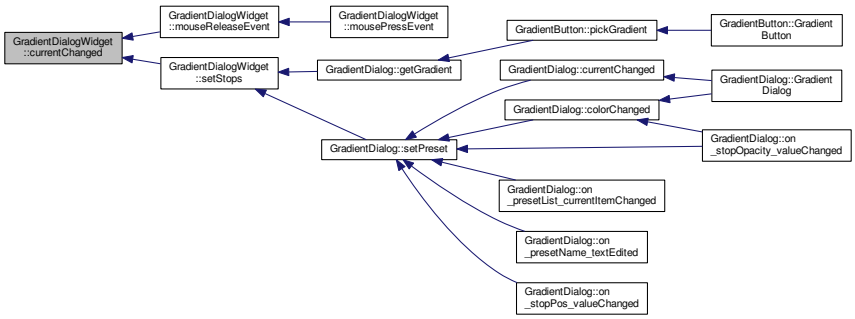
12.19.3.3 void GradientDialogWidget::currentChanged (QColor *color*, bool *spontaneous*, int *position*) [signal]

Definition at line 126 of file moc_gradientdialogwidget.cpp.

Referenced by mouseReleaseEvent(), and setStops().

```
127 {
128     void *_a[] = { Q_NULLPTR, const_cast<void*>(reinterpret_cast<const void*>(&_t1)), const_cast<void*>(
      reinterpret_cast<const void*>(&_t2)), const_cast<void*>(reinterpret_cast<const void*>(&_t3)) };
129     QMetaObject::activate(this, &staticMetaObject, 0, _a);
130 }
```

Here is the caller graph for this function:



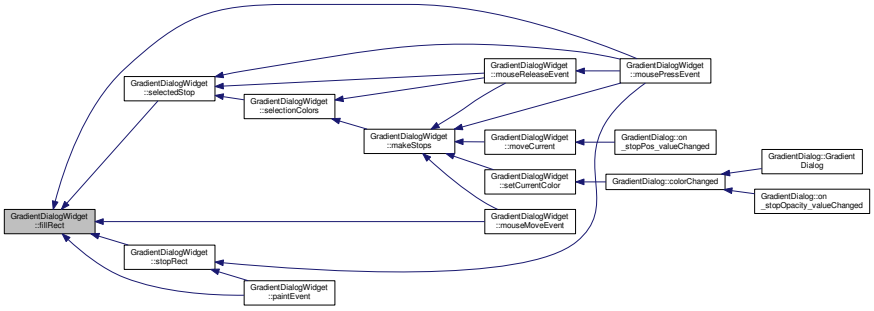
12.19.3.4 QRect GradientDialogWidget::fillRect ( ) [private]

Definition at line 138 of file gradientdialogwidget.cpp.

Referenced by mouseMoveEvent(), mousePressEvent(), paintEvent(), selectedStop(), and stopRect().

```
138                                     {
139     int x = 8, y = 1;
140     int w = (width() - x * 2) / 16 * 16, h = (height() - 24) / 16 * 16;
141     return QRect(x, y, w, h);
142 }
```

Here is the caller graph for this function:



12.19.3.5 void GradientDialogWidget::makeStops ( ) [private]

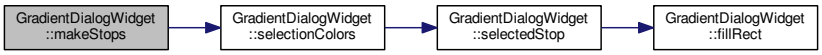
Definition at line 30 of file gradientdialogwidget.cpp.

References _stops, and selectionColors().

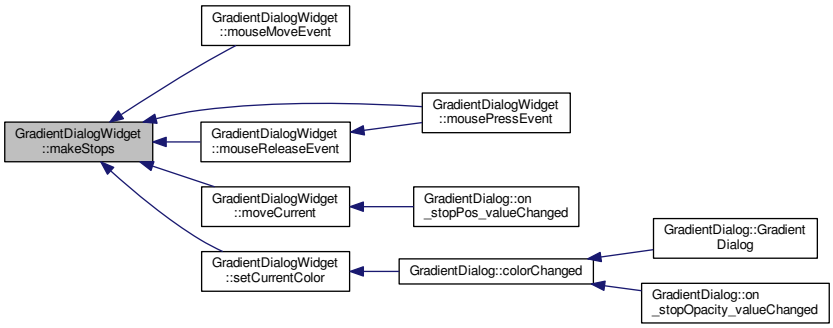
Referenced by mouseMoveEvent(), mousePressEvent(), mouseReleaseEvent(), moveCurrent(), and setCurrentColor().

```
30                                     {
31     _stops.clear();
32     QMap<int, QColor> colors = selectionColors();
33     // If there's only a single color, put it at the beginning and end
34     if(colors.count() == 1){
35 #if QT_VERSION >= QT_VERSION_CHECK(5, 2, 0)
36         QColor color = colors.first();
37 #else
38         QColor color = colors.value(colors.keys().first());
39 #endif
40         _stops.append(QGradientStop(0., color));
41         _stops.append(QGradientStop(1., color));
42         update();
43         return;
44     }
45     // Add colors at 0 and 100 if needed
46 #if QT_VERSION >= QT_VERSION_CHECK(5, 2, 0)
47     if(!colors.contains(0))
48         colors[0] = colors.first();
49     if(!colors.contains(100))
50         colors[100] = colors.last();
51 #else
52     if(!colors.contains(0))
53         colors[0] = colors.value(colors.keys().first());
54     if(!colors.contains(100))
55         colors[100] = colors.value(colors.keys().last());
56 #endif
57     QMapIterator<int, QColor> i(colors);
58     while(i.hasNext()){
59         i.next();
60         _stops.append(QGradientStop(i.key() / 100., i.value()));
61     }
62     update();
63 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



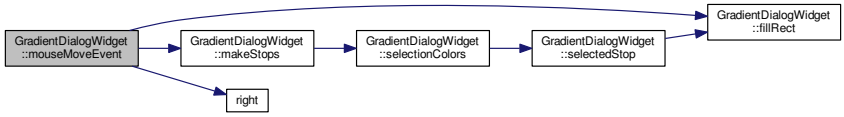
12.19.3.6 void GradientDialogWidget::mouseMoveEvent (QMouseEvent * *event*) [private]

Definition at line 255 of file gradientdialogwidget.cpp.

References _colors, fillRect(), makeStops(), right(), selected, selectedOffset, and selectedPos.

```
255                                                     {
256     if(!selected.isValid()){
257         if(event->y() >= height() - 24)
258             setCursor(QCursor(Qt::PointingHandCursor));
259         else
260             setCursor(QCursor(Qt::ArrowCursor));
261         return;
262     }
263     // Move selected point (if any)
264     setCursor(QCursor(Qt::ClosedHandCursor));
265     selectedPos = event->x() + selectedOffset;
266     // Allow the point to be deleted as long as there is at least 1 other point
267     bool last = (_colors.count() == 0);
268     // Hug the left/right edges
269     QRect rect = fillRect();
270     int left = rect.left(), right = rect.right();
271     if(selectedPos < left && (last || selectedPos >= left - 30))
272         selectedPos = left;
273     else if(selectedPos > right && (last || selectedPos <=
    right + 30))
274         selectedPos = right;
275     // Remove point if it's too far above/below the widget
276     int top = -30, bottom = height() + 30;
277     if(!last && (event->y() < top || event->y() > bottom))
278         selectedPos = -1;
279     makeStops();
280 }
```

Here is the call graph for this function:



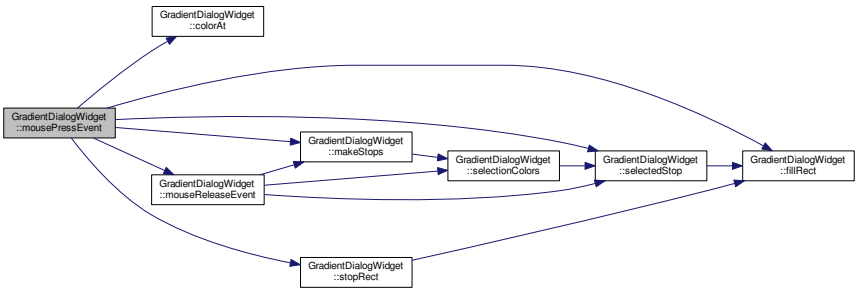
12.19.3.7 void GradientDialogWidget::mousePressEvent (QMouseEvent * *event*) [private]

Definition at line 222 of file gradientdialogwidget.cpp.

References _colors, colorAt(), fillRect(), makeStops(), mouseReleaseEvent(), selected, selectedOffset, selectedPos, selectedStop(), and stopRect().

```
222                                                         {
223     int x = event->x(), y = event->y();
224     if(y < height() - 24)
225         return;
226     QMapIterator<int, QColor> i(_colors);
227     while(i.hasNext()){
228         i.next();
229         // If a point was selected, remove it from the color map and set it as the selected point
230         QRect stop = stopRect(i.key());
231         if(x >= stop.left() - 1 && x <= stop.right() + 1){
232             selected = i.value();
233             selectedPos = stop.x() + stop.width() / 2;
234             selectedOffset = selectedPos - x;
235             _colors.remove(i.key());
236             // On right click, delete the point
237             if(event->button() == Qt::RightButton && _colors.count() > 1){
238                 selectedPos = -1;
239                 mouseReleaseEvent(event);
240             } else
241                 update();
242             return;
243         }
244     }
245     // If nothing was selected, create a new point
246     QRect fill = fillRect();
247     if(x > fill.left() && x < fill.right() && event->button() != Qt::RightButton){
248         selectedPos = x;
249         selectedOffset = 0;
250         selected = colorAt(selectedStop());
251         makeStops();
252     }
253 }
```

Here is the call graph for this function:



12.19.3.8 void GradientDialogWidget::mouseReleaseEvent (QMouseEvent * *event*) [private]

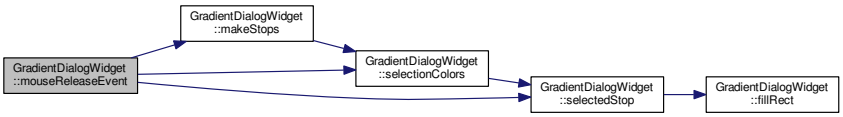
Definition at line 282 of file gradientdialogwidget.cpp.

References _colors, _current, currentChanged(), makeStops(), selected, selectedPos, selectedStop(), and selectionColors().

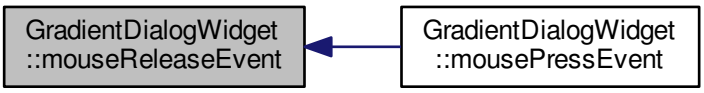
Referenced by mousePressEvent().

```
282                                                         {
283     if(selected.isValid()){
284         // Rejoin selection with gradient
285         _current = selectedStop();
286         _colors = selectionColors();
287         emit currentChanged(_colors.value(_current), true,
    _current);
288         selectedPos = -1;
289         selected = QColor();
290         makeStops();
291     }
292     if(event->y() >= height() - 24)
293         setCursor(QCursor(Qt::PointingHandCursor));
294     else
295         setCursor(QCursor(Qt::ArrowCursor));
296 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.19.3.9 int GradientDialogWidget::moveCurrent (int *to*)

Definition at line 94 of file gradientdialogwidget.cpp.

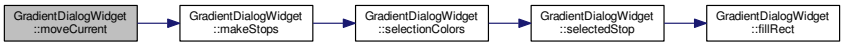
References _colors, _current, makeStops(), and selected.

Referenced by GradientDialog::on_stopPos_valueChanged().

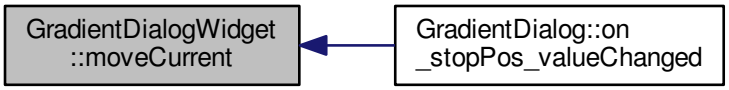
```
94                                                  {
95     if(_current < 0 || to == _current || selected.isValid())
96         return _current;
97     // If there's a point in the way, skip past it by continuing in the current direction
98     if(_current < to){
99         while(_colors.contains(to)){
100             to++;
101             if(to > 100)
```

```
102                 return _current;
103             }
104         } else {
105             while(_colors.contains(to)){
106                 to--;
107                 if(to < 0)
108                     return _current;
109             }
110         }
111         _colors[to] = _colors.take(_current);
112         _current = to;
113         makeStops();
114         return to;
115 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.19.3.10 void GradientDialogWidget::paintEvent (QPaintEvent *) [private]

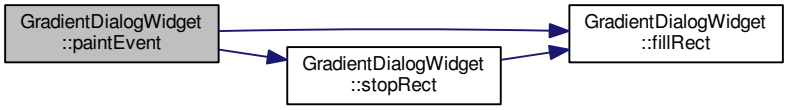
Definition at line 172 of file gradientdialogwidget.cpp.

References _colors, _current, _stops, fillRect(), selected, selectedPos, and stopRect().

```
172                                                 {
173     QPainter painter(this);
174     QRect fill = fillRect();
175     int x = fill.x(), y = fill.y(), w = fill.width(), h = fill.height();
176 
177     // Draw border around gradient
178     painter.setPen(palette().color(QPalette::Dark));
179     painter.drawLine(x - 1, y - 1, x + w, y - 1);
180     painter.drawLine(x - 1, y - 1, x - 1, y + h);
181     painter.setPen(palette().color(QPalette::Light));
182     painter.drawLine(x - 1, y + h, x + w, y + h);
183     painter.drawLine(x + w, y - 1, x + w, y + h);
184     // Draw background grid
185     for(int i = 0; i < w; i += 16){
186         for(int iy = 0; iy < h; iy += 16){
187             if(i % 32 != iy % 32)
188                 painter.fillRect(x + i, y + iy, 16, 16, QColor(192, 192, 192));
189             else
190                 painter.fillRect(x + i, y + iy, 16, 16, QColor(255, 255, 255));
191         }
192     }
193     // Draw gradient
194     painter.setRenderHint(QPainter::Antialiasing);
195     QLinearGradient gradient(x, 0, x + w, 0);
196     gradient.setStops(_stops);
197     painter.fillRect(fill, QBrush(gradient));
198 
199     // Draw stops
```

```
200     QMapIterator<int, QColor> i(_colors);
201     painter.setPen(palette().color(QPalette::Shadow));
202     while(i.hasNext()){
203         i.next();
204         painter.setBrush(QBrush(i.value()));
205         int index = i.key();
206         if(index == _current){
207             // Highlight currently-edited stop
208             painter.setPen(QPen(palette().color(QPalette::Highlight), 2));
209             painter.drawRoundedRect(stopRect(index), 4, 4);
210             painter.setPen(palette().color(QPalette::Shadow));
211         } else
212             painter.drawRoundedRect(stopRect(index), 4, 4);
213     }
214     // Draw selection (if any)
215     if(selected.isValid() && selectedPos >= x && selectedPos <= x + w){
216         painter.setPen(QPen(palette().color(QPalette::Highlight), 2.5));
217         painter.setBrush(selected);
218         painter.drawRoundedRect(selectedPos - 6, height() - 18 - 2, 12, 18, 4, 4);
219     }
220 }
```

Here is the call graph for this function:



12.19.3.11 int GradientDialogWidget::selectedStop ( ) [private]

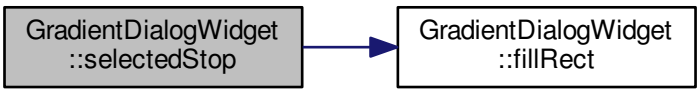
Definition at line 152 of file gradientdialogwidget.cpp.

References fillRect(), and selectedPos.

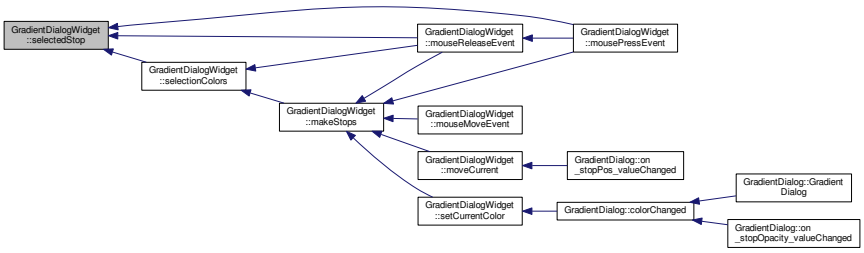
Referenced by mousePressEvent(), mouseReleaseEvent(), and selectionColors().

```
152                                     {
153     QRect rect = fillRect();
154     int res = round((selectedPos - rect.x()) * 100. / rect.width());
155     if(res < 0 || res > 100)
156         return -1;
157     return res;
158 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.19.3.12 QMap< int, QColor > GradientDialogWidget::selectionColors ( ) [private]

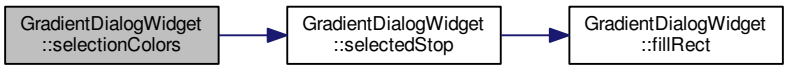
Definition at line 160 of file gradientdialogwidget.cpp.

References _colors, selected, and selectedStop().

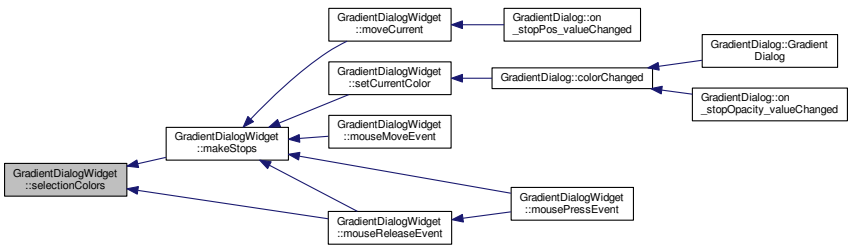
Referenced by makeStops(), and mouseReleaseEvent().

```
160                                                     {
161     if(!selected.isValid())
162         return _colors;
163     // Determine selected stop position
164     int selPos = selectedStop();
165     // Re-add if in range
166     QMap<int, QColor> res = _colors;
167     if(selPos >= 0 && selPos <= 100)
168         res[selPos] = selected;
169     return res;
170 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.19.3.13 void GradientDialogWidget::setCurrentColor (const QColor & *color*)

Definition at line 117 of file gradientdialogwidget.cpp.

References _colors, _current, makeStops(), and selected.

Referenced by GradientDialog::colorChanged().

```
117                                                          {
118     if(_current < 0 || selected.isValid())
119         return;
120     QRgb rgb = _colors.value(_current).rgb();
121     _colors[_current] = color;
122     // If any points follow with the same color but different opacity, change them to match
123     QMutableMapIterator<int, QColor> i(_colors);
124     while(i.hasNext()){
125         i.next();
126         if(i.key() <= _current)
127             continue;
128         QColor& value = i.value();
129         if(value.rgb() != rgb)
130             break;
131         int alpha = value.alpha();
132         value = color;
133         value.setAlpha(alpha);
134     }
135     makeStops();
136 }
```

Here is the call graph for this function:

Here is the caller graph for this function:

12.19.3.14 void GradientDialogWidget::setStops (const QGradientStops & *stops*)

Definition at line 12 of file gradientdialogwidget.cpp.

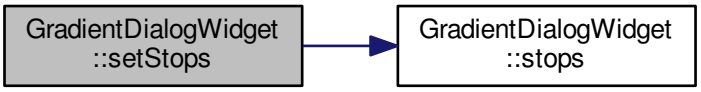
References _colors, _current, _stops, currentChanged(), and stops().

Referenced by GradientDialog::getGradient(), and GradientDialog::setPreset().

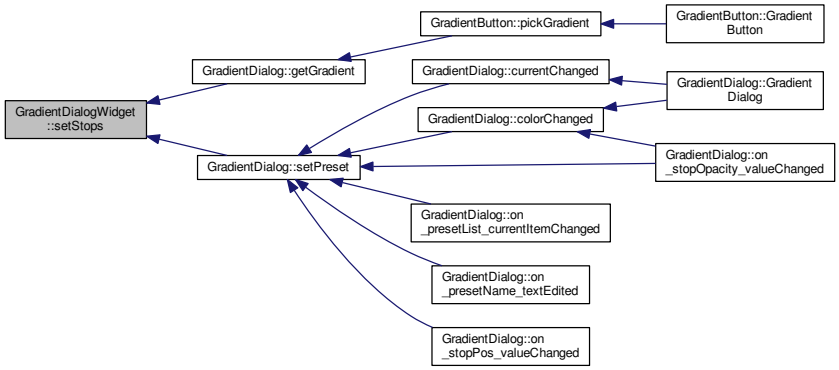
```
12                                                          {
13     _stops = stops;
14     _colors.clear();
15     foreach(const QGradientStop& stop, stops){
16         double pos = stop.first;
17         if(pos < 0. || pos > 1.)
18             continue;
19         _colors[round(pos * 100.)] = stop.second;
20     }
21     _current = 0;
```

```
22 #if QT_VERSION >= QT_VERSION_CHECK(5, 2, 0)
23     emit currentChanged(_colors.first(), false, 0);
24 #else
25     emit currentChanged(_colors.value(_colors.keys().first()), false, 0);
26 #endif
27     update();
28 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.19.3.15 int GradientDialogWidget::stopCount ( ) [inline]

Definition at line 16 of file gradientdialogwidget.h.

References _stops.

Referenced by GradientDialog::currentChanged().

```
16 { return _stops.count(); }
```

Here is the caller graph for this function:

12.19.3.16 QRect GradientDialogWidget::stopRect (int *pos*) [private]

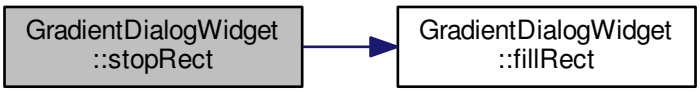
Definition at line 144 of file gradientdialogwidget.cpp.

References fillRect().

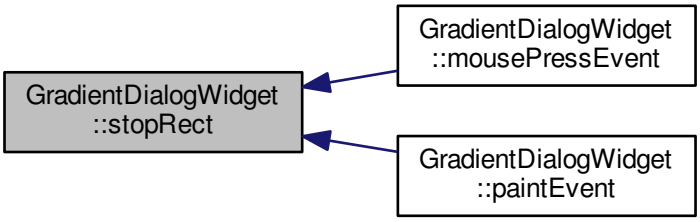
Referenced by mousePressEvent(), and paintEvent().

```
144                                             {
145     QRect fill = fillRect();
146     int sx = round(pos / 100. * fill.width()) + fill.x() - 6;
147     int sw = 12, sh = 18;
148     int sy = height() - sh - 2;
149     return QRect(sx, sy, sw, sh);
150 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.19.3.17 QGradientStops GradientDialogWidget::stops ( ) [inline]

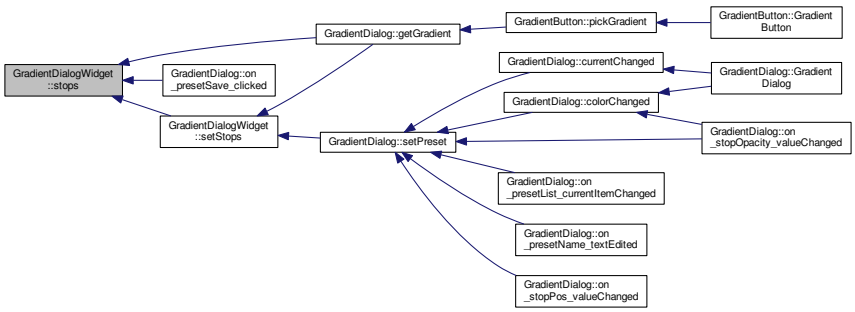
Definition at line 15 of file gradientdialogwidget.h.

References _stops.

Referenced by GradientDialog::getGradient(), GradientDialog::on_presetSave_clicked(), and setStops().

```
15 { return _stops; }
```

Here is the caller graph for this function:



12.19.4 Field Documentation

12.19.4.1 QMap<int, QColor> GradientDialogWidget::_colors [private]

Definition at line 31 of file gradientdialogwidget.h.

Referenced by mouseMoveEvent(), mousePressEvent(), mouseReleaseEvent(), moveCurrent(), paintEvent(), selectionColors(), setCurrentColor(), and setStops().

12.19.4.2 int GradientDialogWidget::_current [private]

Definition at line 30 of file gradientdialogwidget.h.

Referenced by current(), mouseReleaseEvent(), moveCurrent(), paintEvent(), setCurrentColor(), and setStops().

12.19.4.3 QGradientStops GradientDialogWidget::_stops [private]

Definition at line 32 of file gradientdialogwidget.h.

Referenced by colorAt(), makeStops(), paintEvent(), setStops(), stopCount(), and stops().

12.19.4.4 QColor GradientDialogWidget::selected [private]

Definition at line 28 of file gradientdialogwidget.h.

Referenced by mouseMoveEvent(), mousePressEvent(), mouseReleaseEvent(), moveCurrent(), paintEvent(), selectionColors(), and setCurrentColor().

12.19.4.5 int GradientDialogWidget::selectedOffset [private]

Definition at line 29 of file gradientdialogwidget.h.

Referenced by mouseMoveEvent(), and mousePressEvent().

12.19.4.6 int GradientDialogWidget::selectedPos [private]

Definition at line 29 of file gradientdialogwidget.h.

Referenced by mouseMoveEvent(), mousePressEvent(), mouseReleaseEvent(), paintEvent(), and selectedStop().

The documentation for this class was generated from the following files:

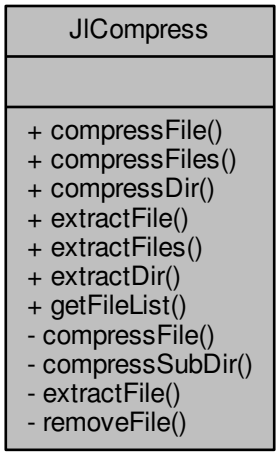
- src/ckb/gradientdialogwidget.h
- src/ckb/gradientdialogwidget.cpp
- src/ckb/release/moc_gradientdialogwidget.cpp

12.20 JlCompress Class Reference

Utility class for typical operations.

`#include <src/ckb/quazip/JlCompress.h>`

Collaboration diagram for JlCompress:



Static Public Member Functions

- static bool compressFile (QString fileCompressed, QString file)

  Compress a single file.
- static bool compressFiles (QString fileCompressed, QStringList files)

  Compress a list of files.
- static bool compressDir (QString fileCompressed, QString dir=QString(), bool recursive=true)

  Compress a whole directory.
- static QString extractFile (QString fileCompressed, QString fileName, QString fileDest=QString())

  Extract a single file.
- static QStringList extractFiles (QString fileCompressed, QStringList files, QString dir=QString())

  Extract a list of files.
- static QStringList extractDir (QString fileCompressed, QString dir=QString())

  Extract a whole archive.
- static QStringList getFileList (QString fileCompressed)

  Get the file list.

Static Private Member Functions

- static bool compressFile (QuaZip *zip, QString fileName, QString fileDest)

  Compress a single file.
- static bool compressSubDir (QuaZip *parentZip, QString dir, QString parentDir, bool recursive=true)

  Compress a subdirectory.
- static bool extractFile (QuaZip *zip, QString fileName, QString fileDest)

  Extract a single file.
- static bool removeFile (QStringList listFile)

  Remove some files.

12.20.1 Detailed Description

This class contains a number of useful static functions to perform simple operations, such as mass ZIP packing or extraction.

Definition at line 42 of file JICompress.h.

12.20.2 Member Function Documentation

12.20.2.1 bool JICompress::compressDir (QString *fileCompressed*, QString *dir* = `QString()`, bool *recursive* = `true`) [static]

OK Comprime la cartella dir nel file fileCompressed, se recursive e true allora comprime anche le sotto cartelle.

Parameters

fileCompressed	The name of the archive.
dir	The directory to compress.
recursive	Whether to pack the subdirectories as well, or just regular files.

Returns

true if success, false otherwise.

Se la funzione fallisce restituisce false e cancella il file che si e tentato di creare.

La funzione fallisce se:

- non si riesce ad aprire l'oggetto zip;
- la compressione di un file fallisce;
- non si riesce a chiudere l'oggetto zip;

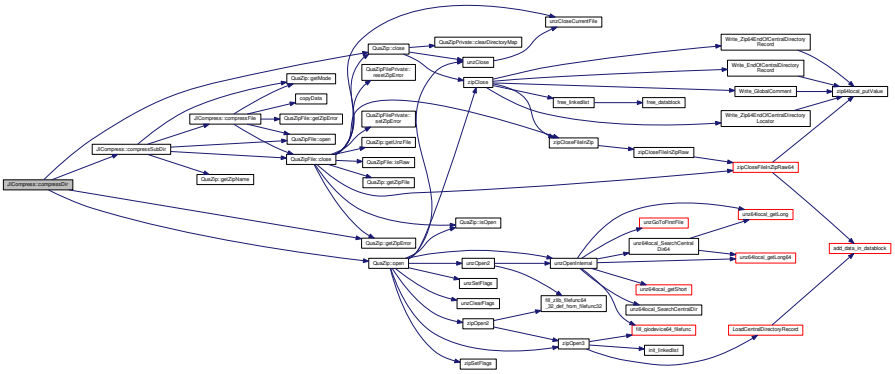
Definition at line 338 of file JICompress.cpp.

References QuaZip::close(), compressSubDir(), QuaZip::getZipError(), QuaZip::mdCreate, and QuaZip::open().

```
338                                                                                 {
339     // Creo lo zip
340     QuaZip zip(fileCompressed);
341     QDir().mkpath(QFileInfo(fileCompressed).absolutePath());
342     if(!zip.open(QuaZip::mdCreate)) {
343         QFile::remove(fileCompressed);
344         return false;
345     }
346
347     // Aggiungo i file e le sotto cartelle
348     if (!compressSubDir(&zip,dir,dir,recursive)) {
349         QFile::remove(fileCompressed);
350         return false;
```

```
351     }
352
353     // Chiudo il file zip
354     zip.close();
355     if(zip.getZipError()!=0) {
356         QFile::remove(fileCompressed);
357         return false;
358     }
359
360     return true;
361 }
```

Here is the call graph for this function:



12.20.2.2 bool JlCompress::compressFile (QuaZip * *zip,* QString *fileName,* QString *fileDest*) [static], [private]

OK Comprime il file fileName, nell'oggetto zip, con il nome fileDest.

Parameters

zip	Opened zip to compress the file to.
fileName	The full path to the source file.
fileDest	The full name of the file inside the archive.

Returns

true if success, false otherwise.

La funzione fallisce se:

- zip==NULL;
- l'oggetto zip e stato aperto in una modalita non compatibile con l'aggiunta di file;
- non e possibile aprire il file d'origine;
- non e possibile creare il file all'interno dell'oggetto zip;
- si e rilevato un errore nella copia dei dati;
- non e stato possibile chiudere il file all'interno dell'oggetto zip;

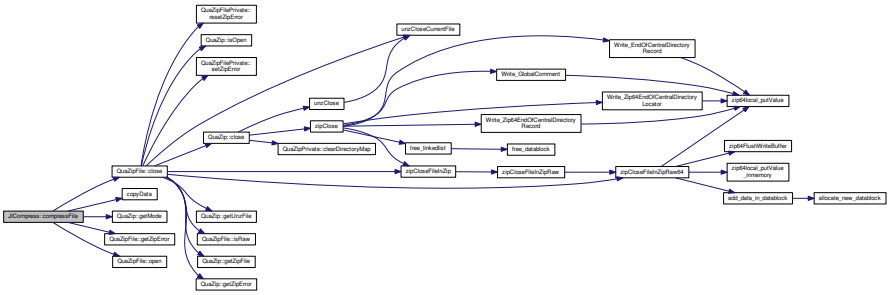
Definition at line 53 of file JlCompress.cpp.

References QuaZipFile::close(), copyData(), QuaZip::getMode(), QuaZipFile::getZipError(), QuaZip::mdAdd, QuaZip::mdAppend, QuaZip::mdCreate, QuaZipFile::open(), and UNZ_OK.

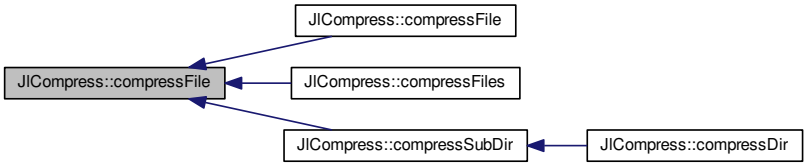
Referenced by compressFile(), compressFiles(), and compressSubDir().

```
53                                                                                  {
54      // zip: oggetto dove aggiungere il file
55      // fileName: nome del file reale
56      // fileDest: nome del file all'interno del file compresso
57
58      // Controllo l'apertura dello zip
59      if (!zip) return false;
60      if (zip->getMode()!=QuaZip::mdCreate &&
61          zip->getMode()!=QuaZip::mdAppend &&
62          zip->getMode()!=QuaZip::mdAdd) return false;
63
64      // Apro il file originale
65      QFile inFile;
66      inFile.setFileName(fileName);
67      if(!inFile.open(QIODevice::ReadOnly)) return false;
68
69      // Apro il file risulato
70      QuaZipFile outFile(zip);
71      if(!outFile.open(QIODevice::WriteOnly, QuaZipNewInfo(fileDest, inFile.fileName()))) return
        false;
72
73      // Copio i dati
74      if (!copyData(inFile, outFile) || outFile.getZipError()!=UNZ_OK) {
75          return false;
76      }
77
78      // Chiudo i file
79      outFile.close();
80      if (outFile.getZipError()!=UNZ_OK) return false;
81      inFile.close();
82
83      return true;
84  }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.20.2.3 bool JICompress::compressFile (QString *fileCompressed,* QString *file*) [static]

OK Comprime il file fileName nel file fileCompressed.

Parameters

fileCompressed	The name of the archive.
file	The file to compress.

Returns

true if success, false otherwise.

Se la funzione fallisce restituisce false e cancella il file che si e tentato di creare.

La funzione fallisce se:

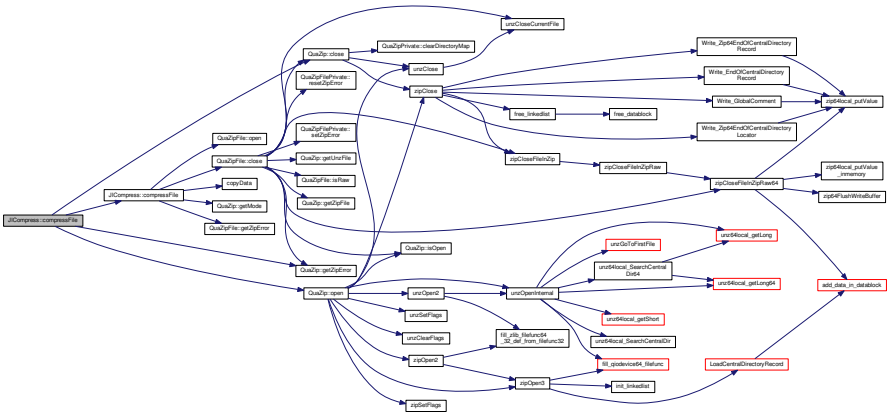
- non si riesce ad aprire l'oggetto zip;
- la compressione del file fallisce;
- non si riesce a chiudere l'oggetto zip;

Definition at line 263 of file JlCompress.cpp.

References QuaZip::close(), compressFile(), QuaZip::getZipError(), QuaZip::mdCreate, and QuaZip::open().

```
263                                                                 {
264     // Creo lo zip
265     QuaZip zip(fileCompressed);
266     QDir().mkpath(QFileInfo(fileCompressed).absolutePath());
267     if(!zip.open(QuaZip::mdCreate)) {
268         QFile::remove(fileCompressed);
269         return false;
270     }
271
272     // Aggiungo il file
273     if (!compressFile(&zip,file,QFileInfo(file).fileName())) {
274         QFile::remove(fileCompressed);
275         return false;
276     }
277
278     // Chiudo il file zip
279     zip.close();
280     if(zip.getZipError()!=0) {
281         QFile::remove(fileCompressed);
282         return false;
283     }
284
285     return true;
286 }
```

Here is the call graph for this function:



12.20.2.4 bool JICompress::compressFiles (QString *fileCompressed*, QStringList *files*) [static]

OK Comprime i file specificati in files nel file fileCompressed.

Parameters

fileCompressed	The name of the archive.
files	The file list to compress.

Returns

true if success, false otherwise.

Se la funzione fallisce restituisce false e cancella il file che si e tentato di creare.

La funzione fallisce se:

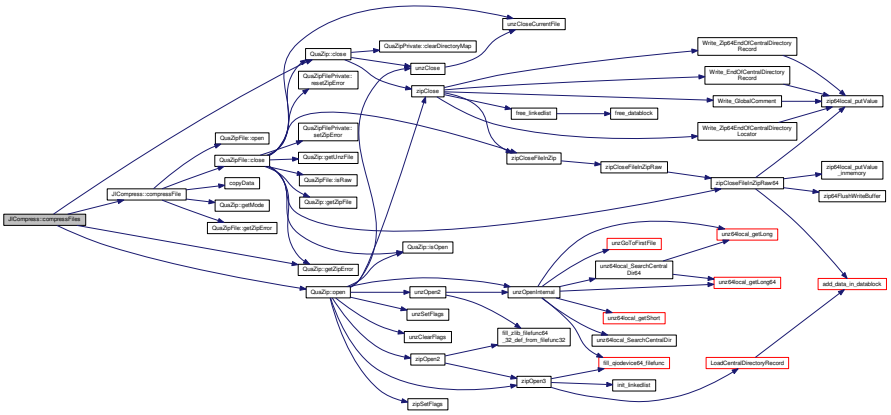
- non si riesce ad aprire l'oggetto zip;
- la compressione di un file fallisce;
- non si riesce a chiudere l'oggetto zip;

Definition at line 298 of file JlCompress.cpp.

References QuaZip::close(), compressFile(), QuaZip::getZipError(), QuaZip::mdCreate, and QuaZip::open().

```
298                                                                     {
299     // Creo lo zip
300     QuaZip zip(fileCompressed);
301     QDir().mkpath(QFileInfo(fileCompressed).absolutePath());
302     if(!zip.open(QuaZip::mdCreate)) {
303         QFile::remove(fileCompressed);
304         return false;
305     }
306
307     // Comprimo i file
308     QFileInfo info;
309     Q_FOREACH (QString file, files) {
310         info.setFile(file);
311         if (!info.exists() || !compressFile(&zip,file,info.fileName())) {
312             QFile::remove(fileCompressed);
313             return false;
314         }
315     }
316
317     // Chiudo il file zip
318     zip.close();
319     if(zip.getZipError()!=0) {
320         QFile::remove(fileCompressed);
321         return false;
322     }
323
324     return true;
325 }
```

Here is the call graph for this function:



12.20.2.5 bool JICompress::compressSubDir (QuaZip * *zip,* QString *dir,* QString *origDir,* bool *recursive* = `true`) [static], [private]

OK Comprime la cartella dir nel file fileCompressed, se recursive e true allora comprime anche le sotto cartelle.

Parameters

parentZip	Opened zip containing the parent directory.
dir	The full path to the directory to pack.
parentDir	The full path to the directory corresponding to the root of the ZIP.
recursive	Whether to pack sub-directories as well or only files.

Returns

true if success, false otherwise.

I nomi dei file preceduti dal path creato togliendo il pat della cartella origDir al path della cartella dir. Se la funzione fallisce restituisce false e cancella il file che si e tentato di creare.

La funzione fallisce se:

- zip==NULL;
- l'oggetto zip e stato aperto in una modalita non compatibile con l'aggiunta di file;
- la cartella dir non esiste;
- la compressione di una sotto cartella fallisce (1);
- la compressione di un file fallisce; (1) La funzione si richiama in maniera ricorsiva per comprimere le sotto cartelle dunque gli errori di compressione di una sotto cartella sono gli stessi di questa funzione.

Definition at line 103 of file JlCompress.cpp.

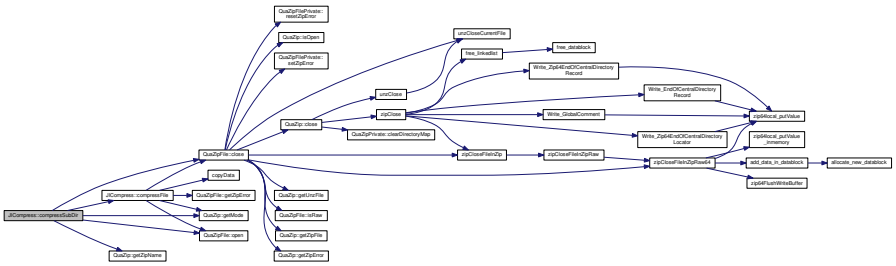
References QuaZipFile::close(), compressFile(), QuaZip::getMode(), QuaZip::getZipName(), QuaZip::mdAdd, QuaZip::mdAppend, QuaZip::mdCreate, and QuaZipFile::open().

Referenced by compressDir().

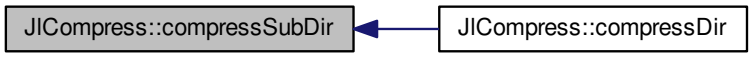
```
103                                                                                                 {
104     // zip: oggetto dove aggiungere il file
105     // dir: cartella reale corrente
106     // origDir: cartella reale originale
107     // (path(dir)-path(origDir)) = path interno all'oggetto zip
108
109     // Controllo l'apertura dello zip
110     if (!zip) return false;
111     if (zip->getMode()!=QuaZip::mdCreate &&
112         zip->getMode()!=QuaZip::mdAppend &&
113         zip->getMode()!=QuaZip::mdAdd) return false;
114
115     // Controllo la cartella
116     QDir directory(dir);
117     if (!directory.exists()) return false;
118
119     QDir origDirectory(origDir);
120     if (dir != origDir) {
121         QuaZipFile dirZipFile(zip);
122         if (!dirZipFile.open(QIODevice::WriteOnly,
123             QuaZipNewInfo(origDirectory.relativeFilePath(dir) + "/", dir), 0, 0, 0)) {
124                 return false;
125         }
126         dirZipFile.close();
127     }
128
129
130     // Se comprimo anche le sotto cartelle
131     if (recursive) {
132         // Per ogni sotto cartella
133         QFileInfoList files = directory.entryInfoList(QDir::AllDirs|QDir::NoDotAndDotDot);
134         Q_FOREACH (QFileInfo file, files) {
135             // Comprimo la sotto cartella
136             if(!compressSubDir(zip,file.absoluteFilePath(),origDir,recursive)) return false;
137         }
138     }
139
140     // Per ogni file nella cartella
141     QFileInfoList files = directory.entryInfoList(QDir::Files);
142     Q_FOREACH (QFileInfo file, files) {
143         // Se non e un file o e il file compresso che sto creando
```

```
144         if(!file.isFile()||file.absoluteFilePath()==zip->getZipName()) continue;
145
146         // Creo il nome relativo da usare all'interno del file compresso
147         QString filename = origDirectory.relativeFilePath(file.absoluteFilePath());
148
149         // Comprimo il file
150         if (!compressFile(zip,file.absoluteFilePath(),filename)) return false;
151     }
152
153     return true;
154 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.20.2.6 QStringList JICompress::extractDir (QString *fileCompressed*, QString *dir* = QString()) [static]

OK Estrae il file fileCompressed nella cartella dir.

Parameters

fileCompressed	The name of the archive.
dir	The directory to extract to, the current directory if left empty.

Returns

The list of the full paths of the files extracted, empty on failure.

Se dir = "" allora il file viene estratto nella cartella corrente. Se la funzione fallisce cancella i file che si e tentato di estrarre. Restituisce i nomi assoluti dei file estratti.

La funzione fallisce se:

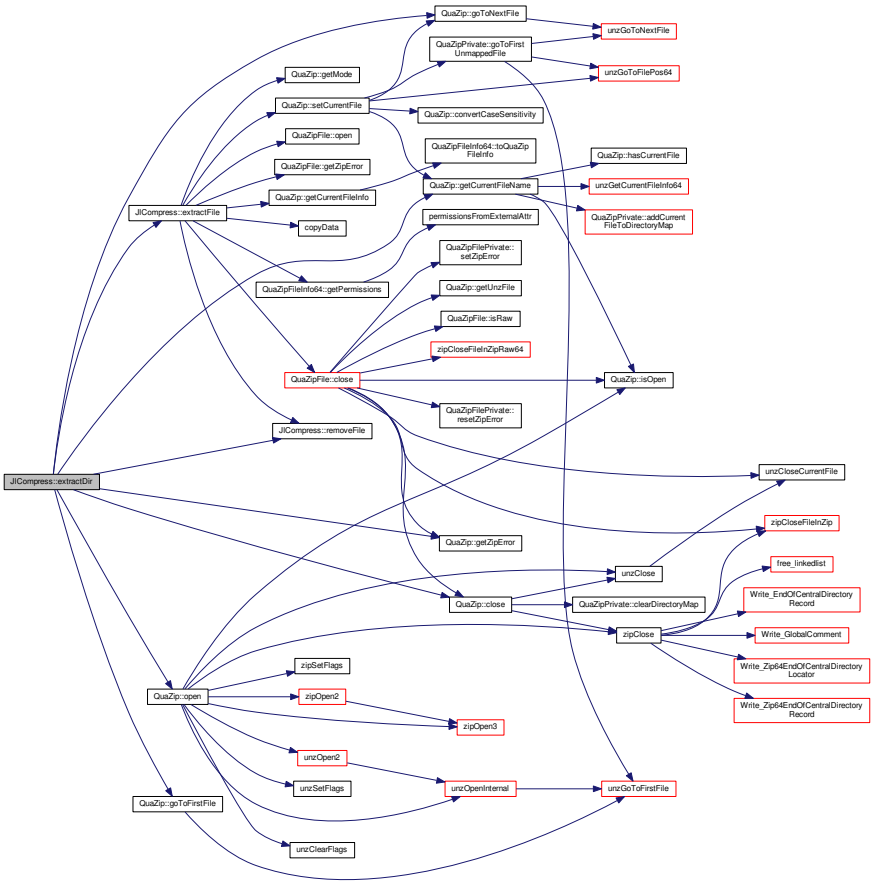
- non si riesce ad aprire l'oggetto zip;
- la compressione di un file fallisce;
- non si riesce a chiudere l'oggetto zip;

Definition at line 451 of file JlCompress.cpp.

References QuaZip::close(), extractFile(), QuaZip::getCurrentFileName(), QuaZip::getZipError(), QuaZip::goToFirstFile(), QuaZip::goToNextFile(), QuaZip::mdUnzip, QuaZip::open(), and removeFile().

```
451                                                                 {
452     // Apro lo zip
453     QuaZip zip(fileCompressed);
454     if(!zip.open(QuaZip::mdUnzip)) {
455         return QStringList();
456     }
457
458     QDir directory(dir);
459     QStringList extracted;
460     if (!zip.goToFirstFile()) {
461         return QStringList();
462     }
463     do {
464         QString name = zip.getCurrentFileName();
465         QString absFilePath = directory.absoluteFilePath(name);
466         if (!extractFile(&zip, "", absFilePath)) {
467             removeFile(extracted);
468             return QStringList();
469         }
470         extracted.append(absFilePath);
471     } while (zip.goToNextFile());
472
473     // Chiudo il file zip
474     zip.close();
475     if(zip.getZipError()!=0) {
476         removeFile(extracted);
477         return QStringList();
478     }
479
480     return extracted;
481 }
```

Here is the call graph for this function:



12.20.2.7 bool JICompress::extractFile (QuaZip * *zip*, QString *fileName*, QString *fileDest*) [static], [private]

OK Estrae il file fileName, contenuto nell'oggetto zip, con il nome fileDest.

Parameters

zip	The opened zip archive to extract from.
fileName	The full name of the file to extract.
fileDest	The full path to the destination file.

Returns

true if success, false otherwise.

Se la funzione fallisce restituisce false e cancella il file che si e tentato di estrarre.

La funzione fallisce se:

- zip==NULL;
- l'oggetto zip e stato aperto in una modalita non compatibile con l'estrazione di file;
- non e possibile aprire il file all'interno dell'oggetto zip;
- non e possibile creare il file estratto;
- si e rilevato un errore nella copia dei dati (1);
- non e stato possibile chiudere il file all'interno dell'oggetto zip (1);

(1): prima di uscire dalla funzione cancella il file estratto.

Definition at line 170 of file JICompress.cpp.

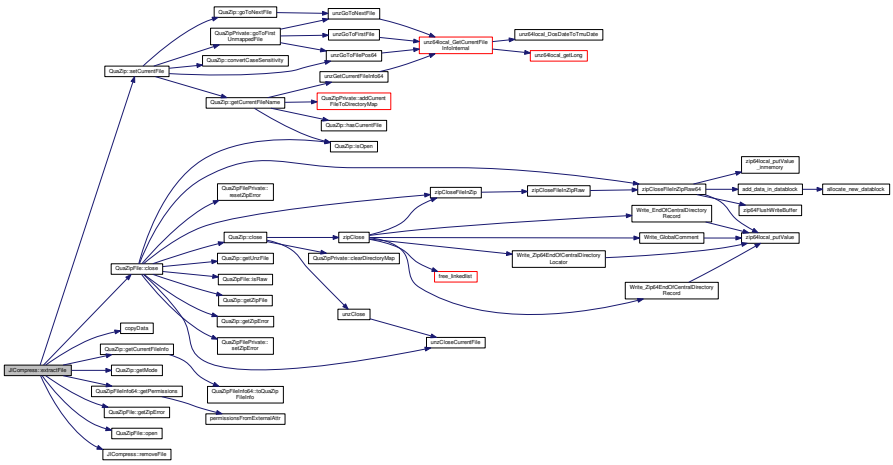
References QuaZipFile::close(), copyData(), QuaZip::getCurrentFileInfo(), QuaZip::getMode(), QuaZipFileInfo64::getPermissions(), QuaZipFile::getZipError(), QuaZip::mdUnzip, QuaZipFile::open(), removeFile(), QuaZip::setCurrentFile(), and UNZ_OK.

Referenced by extractDir(), extractFile(), and extractFiles().

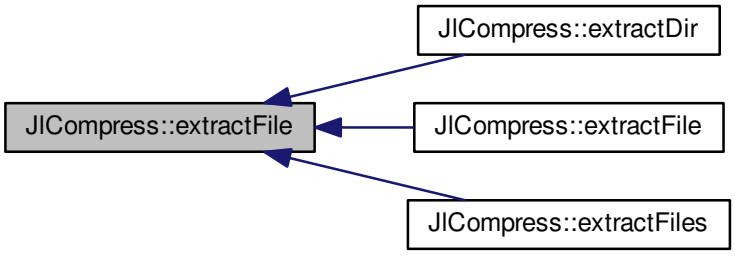
```
170                                                                                 {
171     // zip: oggetto dove aggiungere il file
172     // filename: nome del file reale
173     // fileincompress: nome del file all'interno del file compresso
174
175     // Controllo l'apertura dello zip
176     if (!zip) return false;
177     if (zip->getMode()!=QuaZip::mdUnzip) return false;
178
179     // Apro il file compresso
180     if (!fileName.isEmpty())
181         zip->setCurrentFile(fileName);
182     QuaZipFile inFile(zip);
183     if(!inFile.open(QIODevice::ReadOnly) || inFile.getZipError()!=UNZ_OK) return false;
184
185     // Controllo esistenza cartella file risultato
186     QDir curDir;
187     if (fileDest.endsWith('/')) {
188         if (!curDir.mkpath(fileDest)) {
189             return false;
190         }
191     } else {
192         if (!curDir.mkpath(QFileInfo(fileDest).absolutePath())) {
193             return false;
194         }
195     }
196
197     QuaZipFileInfo64 info;
198     if (!zip->getCurrentFileInfo(&info))
199         return false;
200
201     QFile::Permissions srcPerm = info.getPermissions();
202     if (fileDest.endsWith('/') && QFileInfo(fileDest).isDir()) {
203         if (srcPerm != 0) {
```

```
204             QFile(fileDest).setPermissions(srcPerm);
205         }
206         return true;
207     }
208
209     // Apro il file risultato
210     QFile outFile;
211     outFile.setFileName(fileDest);
212     if(!outFile.open(QIODevice::WriteOnly)) return false;
213
214     // Copio i dati
215     if (!copyData(inFile, outFile) || inFile.getZipError()!=UNZ_OK) {
216         outFile.close();
217         removeFile(QStringList(fileDest));
218         return false;
219     }
220     outFile.close();
221
222     // Chiudo i file
223     inFile.close();
224     if (inFile.getZipError()!=UNZ_OK) {
225         removeFile(QStringList(fileDest));
226         return false;
227     }
228
229     if (srcPerm != 0) {
230         outFile.setPermissions(srcPerm);
231     }
232     return true;
233 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.20.2.8 QString JICompress::extractFile (QString *fileCompressed,* QString *fileName,* QString *fileDest* = `QString()`) [static]

OK Estrae il file fileName, contenuto nel file fileCompressed, con il nome fileDest.

Parameters

fileCompressed	The name of the archive.
fileName	The file to extract.
fileDest	The destination file, assumed to be identical to *file* if left empty.

Returns

The list of the full paths of the files extracted, empty on failure.

Se fileDest = "" allora il file viene estratto con lo stesso nome con cui e stato compresso. Se la funzione fallisce cancella il file che si e tentato di estrarre. Restituisce il nome assoluto del file estratto.

La funzione fallisce se:

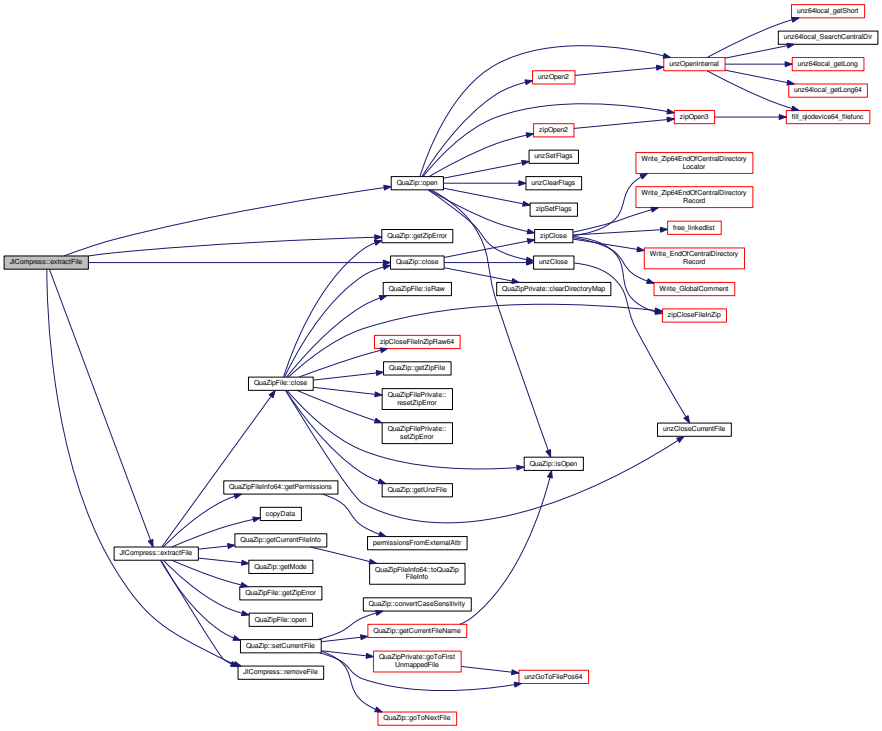
- non si riesce ad aprire l'oggetto zip;
- l'estrazione del file fallisce;
- non si riesce a chiudere l'oggetto zip;

Definition at line 377 of file JICompress.cpp.

References QuaZip::close(), extractFile(), QuaZip::getZipError(), QuaZip::mdUnzip, QuaZip::open(), and removeFile().

```
377                                                                                 {
378     // Apro lo zip
379     QuaZip zip(fileCompressed);
380     if(!zip.open(QuaZip::mdUnzip)) {
381         return QString();
382     }
383
384     // Estraggo il file
385     if (fileDest.isEmpty())
386         fileDest = fileName;
387     if (!extractFile(&zip,fileName,fileDest)) {
388         return QString();
389     }
390
391     // Chiudo il file zip
392     zip.close();
393     if(zip.getZipError()!=0) {
394         removeFile(QStringList(fileDest));
395         return QString();
396     }
397     return QFileInfo(fileDest).absoluteFilePath();
398 }
```

Here is the call graph for this function:



12.20.2.9 QStringList JICompress::extractFiles (QString *fileCompressed,* QStringList *files,* QString *dir* = `QString()`) [static]

OK Estrae i file specificati in files, contenuti nel file fileCompressed, nella cartella dir.

Parameters

fileCompressed	The name of the archive.
files	The file list to extract.
dir	The directory to put the files to, the current directory if left empty.

Returns

The list of the full paths of the files extracted, empty on failure.

La struttura a cartelle del file compresso viene rispettata. Se dir = "" allora il file viene estratto nella cartella corrente. Se la funzione fallisce cancella i file che si e tentato di estrarre. Restituisce i nomi assoluti dei file estratti.

La funzione fallisce se:

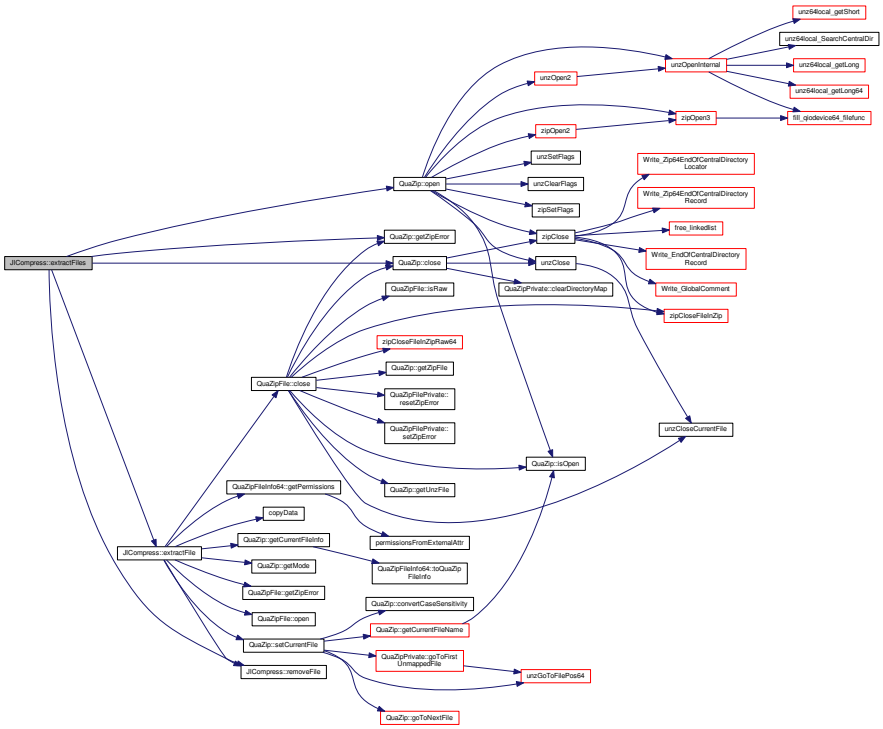
- non si riesce ad aprire l'oggetto zip;
- l'estrazione di un file fallisce;
- non si riesce a chiudere l'oggetto zip;

Definition at line 412 of file JICompress.cpp.

References QuaZip::close(), extractFile(), QuaZip::getZipError(), QuaZip::mdUnzip, QuaZip::open(), and removeFile().

```
412                                                                                                 {
413     // Creo lo zip
414     QuaZip zip(fileCompressed);
415     if(!zip.open(QuaZip::mdUnzip)) {
416         return QStringList();
417     }
418
419     // Estraggo i file
420     QStringList extracted;
421     for (int i=0; i<files.count(); i++) {
422         QString absPath = QDir(dir).absoluteFilePath(files.at(i));
423         if (!extractFile(&zip, files.at(i), absPath)) {
424             removeFile(extracted);
425             return QStringList();
426         }
427         extracted.append(absPath);
428     }
429
430     // Chiudo il file zip
431     zip.close();
432     if(zip.getZipError()!=0) {
433         removeFile(extracted);
434         return QStringList();
435     }
436
437     return extracted;
438 }
```

Here is the call graph for this function:



12.20.2.10 QStringList JICompress::getFileList (QString *fileCompressed*) [static]

OK Restituisce la lista dei file resenti nel file compresso fileCompressed.

Returns

The list of the files in the archive, or, more precisely, the list of the entries, including both files and directories if they are present separately.

Se la funzione fallisce, restituisce un elenco vuoto.

La funzione fallisce se:

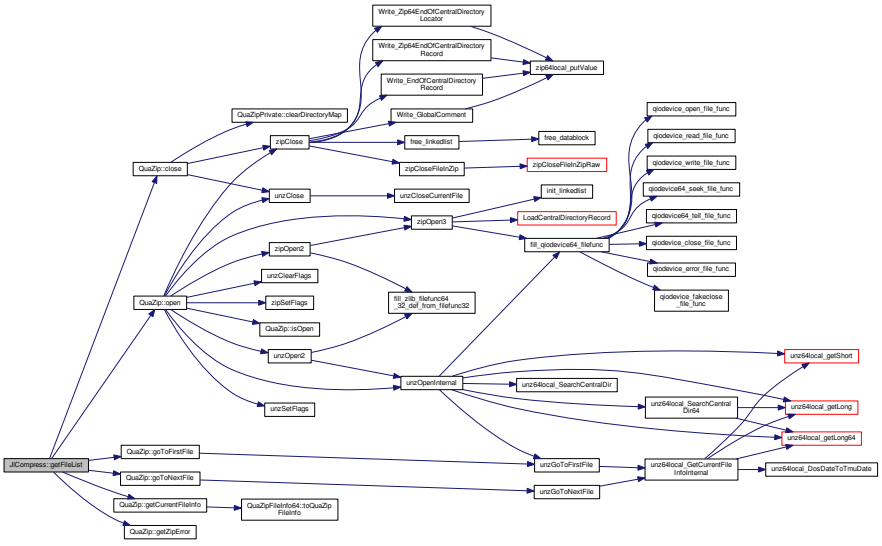
- non si riesce ad aprire l'oggetto zip;
- la richiesta di informazioni di un file fallisce;
- non si riesce a chiudere l'oggetto zip;

Definition at line 492 of file JlCompress.cpp.

References QuaZip::close(), QuaZip::getCurrentFileInfo(), QuaZip::getZipError(), QuaZip::goToFirstFile(), QuaZip::goToNextFile(), QuaZip::mdUnzip, QuaZipFileInfo64::name, and QuaZip::open().

```
492                                                     {
493     // Apro lo zip
494     QuaZip* zip = new QuaZip(QFileInfo(fileCompressed).absoluteFilePath());
495     if(!zip->open(QuaZip::mdUnzip)) {
496         delete zip;
497         return QStringList();
498     }
499
500     // Estraggo i nomi dei file
501     QStringList lst;
502     QuaZipFileInfo64 info;
503     for(bool more=zip->goToFirstFile(); more; more=zip->
      goToNextFile()) {
504         if(!zip->getCurrentFileInfo(&info)) {
505             delete zip;
506             return QStringList();
507         }
508         lst << info.name;
509         //info.name.toLocal8Bit().constData()
510     }
511
512     // Chiudo il file zip
513     zip->close();
514     if(zip->getZipError()!=0) {
515         delete zip;
516         return QStringList();
517     }
518     delete zip;
519
520     return lst;
521 }
```

Here is the call graph for this function:



12.20.2.11 bool JICompress::removeFile (QStringList *listFile*) [static],[private]

Rimuove i file il cui nome e specificato all'interno di listFile.

Parameters

listFile	The list of files to remove.

Returns

true if success, false otherwise.

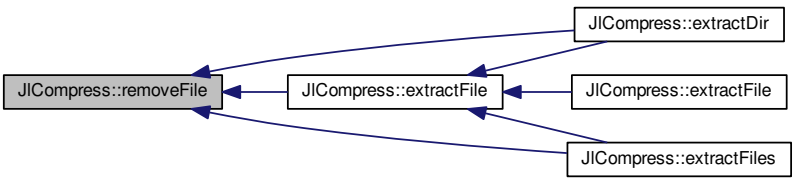
Restituisce true se tutti i file sono stati cancellati correttamente, attenzione perche puo restituire false anche se alcuni file non esistevano e si e tentato di cancellarli.

Definition at line 241 of file JlCompress.cpp.

Referenced by extractDir(), extractFile(), and extractFiles().

```
241                                                 {
242     bool ret = true;
243     // Per ogni file
244     for (int i=0; i<listFile.count(); i++) {
245         // Lo elimino
246         ret = ret && QFile::remove(listFile.at(i));
247     }
248     return ret;
249 }
```

Here is the caller graph for this function:



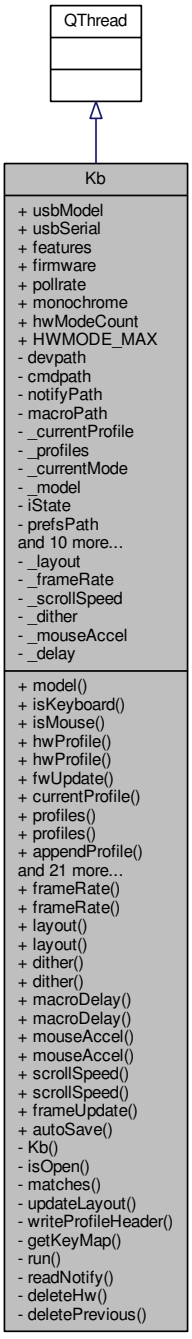
The documentation for this class was generated from the following files:

- src/ckb/quazip/JlCompress.h
- src/ckb/quazip/JlCompress.cpp

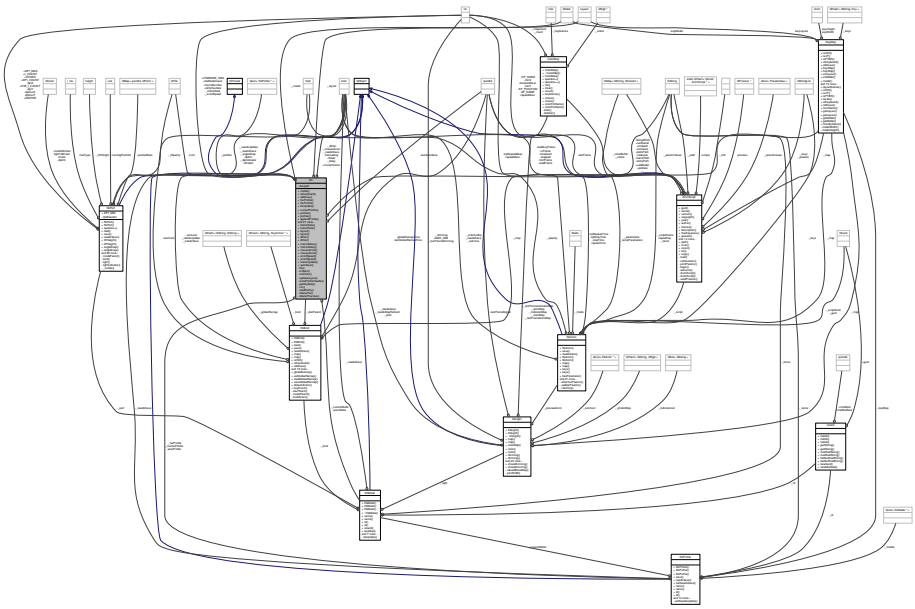
12.21 Kb Class Reference

`#include <src/ckb/kb.h>`

Inheritance diagram for Kb:



Collaboration diagram for Kb:



Public Slots

- void frameUpdate ()
- void autoSave ()

Signals

- void infoUpdated ()
- void profileAdded ()
- void profileRenamed ()
- void modeRenamed ()
- void profileChanged ()
- void modeChanged (bool spontaneous)
- void fwUpdateProgress (int current, int total)
- void fwUpdateFinished (bool succeeded)

Public Member Functions

- KeyMap::Model model () const
- bool isKeyboard () const
- bool isMouse () const
- KbProfile * hwProfile ()
- void hwProfile (KbProfile *newHwProfile)
- void fwUpdate (const QString &path)
- KbProfile * currentProfile ()
- const QList< KbProfile * > & profiles () const
- void profiles (const QList< KbProfile * > &newProfiles)
- void appendProfile (KbProfile *newProfile)
- int indexOf (KbProfile *profile)
- KbProfile * find (const QUuid &id)
- KbMode * currentMode ()

- KbLight * currentLight ()
- KbBind * currentBind ()
- KbPerf * currentPerf ()
- void setCurrentProfile (KbProfile *profile, bool spontaneous=true)
- void setCurrentMode (KbProfile *profile, KbMode *mode, bool spontaneous=true)
- void setCurrentMode (KbProfile *profile, int index, bool spontaneous=true)
- void setCurrentMode (KbMode *mode, bool spontaneous=true)
- KbProfile * newProfile ()
- KbProfile * newProfile (KbProfile *other)
- KbMode * newMode ()
- KbMode * newMode (KbMode *other)
- void load ()
- void save ()
- bool needsSave () const
- void hwSave ()
- int getMacroNumber ()

  For usage with macro definions, these two params must only be readable. So there are no setters.
- QString getMacroPath ()

  getMacroPath returns the macroPath (e.g. /dev/input/ckb1/notify), which we have saved in the constructor. For usage with macro definions, this param must only be readable. So there is no setter.
- ~Kb ()

Static Public Member Functions

- static int frameRate ()
- static void frameRate (int newFrameRate)
- static KeyMap::Layout layout ()
- static void layout (KeyMap::Layout newLayout)
- static bool dither ()
- static void dither (bool newDither)
- static bool macroDelay ()
- static void macroDelay (bool flag)

  Kb::macroDelay handles the UI-Element macroBox. Sends a command to the keyboard to switch on or off the delay function on very large macros.
- static bool mouseAccel ()
- static void mouseAccel (bool newAccel)
- static int scrollSpeed ()
- static void scrollSpeed (int newSpeed)

Data Fields

- QString usbModel
- QString usbSerial
- QString features
- QString firmware
- QString pollrate
- bool monochrome
- int hwModeCount

Static Public Attributes

- static const int HWMODE_MAX = 3

Private Slots

- void readNotify (QString line)
- void deleteHw ()
- void deletePrevious ()

Private Member Functions

- Kb (QObject *parent, const QString &path)
- bool isOpen () const
- bool matches (const QString &path, const QString &serial)
- void updateLayout ()
- void writeProfileHeader ()
- KeyMap getKeyMap ()
- void run ()

Private Attributes

- QString devpath

  pathVars devpath is the device root path (e.g. /dev/device/ckb1), cmdpath leads to the daemon input pipe for daemon commands, notifyPath is the standard input monitor for general purpose, macroPath added for a second thread to read macro input.
- QString cmdpath
- QString notifyPath
- QString macroPath
- KbProfile * _currentProfile
- QList< KbProfile * > _profiles
- KbMode * _currentMode
- KeyMap::Model _model
- bool iState [KbPerf::HW_I_COUNT]
- QString prefsPath
- quint64 lastAutoSave
- QString fwUpdPath
- KbProfile * _hwProfile
- KbProfile * prevProfile
- KbMode * prevMode
- QFile cmd
- int notifyNumber

  notifyNumber is the trailing number in the device path.
- int macroNumber
- bool _needsSave
- bool hwLoading [HWMODE_MAX+1]

Static Private Attributes

- static KeyMap::Layout _layout = KeyMap::NO_LAYOUT
- static int _frameRate = 30
- static int _scrollSpeed = 0
- static bool _dither = false
- static bool _mouseAccel = true
- static bool _delay = false

Friends

- class KbManager

12.21.1 Detailed Description

Definition at line 11 of file kb.h.

12.21.2 Constructor & Destructor Documentation

12.21.2.1 Kb::~Kb ( )

<

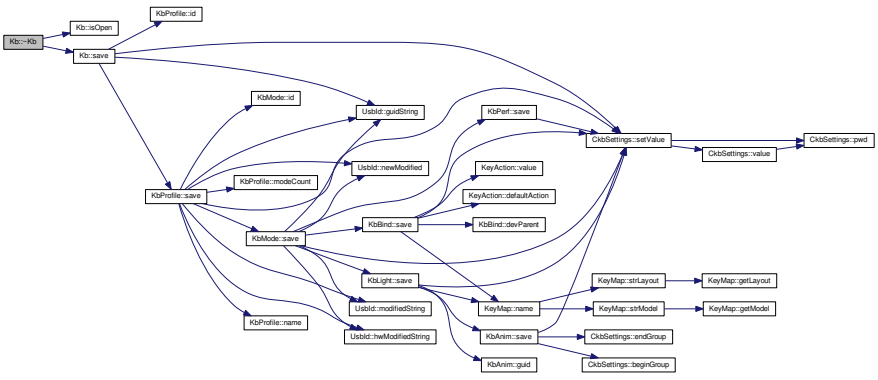
Todo I don't think, that notifypaths is used somewhere. So why do we have it? If we do not need it, searching for an ununsed notify channel can easy be refactored to a private member function.

Definition at line 141 of file kb.cpp.

References activeDevices, cmd, isOpen(), macroPath, notifyNumber, notifyPaths, and save().

```
141         {
142     // Save settings first
143     save();
144
145     // remove the notify channel from the list of notifyPaths.
148     notifyPaths.remove(macroPath);
149
150     // Kill notification thread and remove node
151     activeDevices.remove(this);
152     if(!isOpen()){
153         terminate();
154         wait(1000);
155         return;
156     }
157     if(notifyNumber > 0)
158         cmd.write(QString("idle\nnotifyoff %1\n").arg(notifyNumber).toLatin1());
159     cmd.flush();
160     terminate();
161     wait(1000);
162     cmd.close();
163 }
```

Here is the call graph for this function:



12.21.2.2 Kb::Kb (QObject * *parent*, const QString & *path*) [private]

<

Todo Is adding notify2 to the notifypaths neccessary?

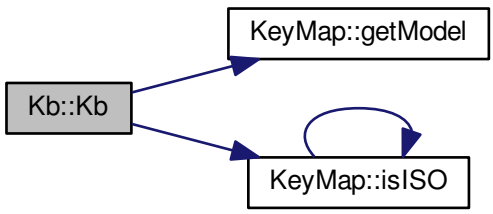
Definition at line 19 of file kb.cpp.

References _delay, _dither, _frameRate, _layout, _model, _mouseAccel, _scrollSpeed, activeDevices, cmd, cmdpath, features, firmware, KeyMap::getModel(), hwLoading, hwModeCount, infoUpdated(), KeyMap::isISO(), iState, KeyMap::K95, macroNumber, macroPath, monochrome, KeyMap::NO_MODEL, notifyNumber, notifyPath, notifyPathMutex, notifyPaths, pollrate, prefsPath, usbModel, and usbSerial.

```
19                                      :
20     QThread(parent), features("N/A"), firmware("N/A"),
      pollrate("N/A"), monochrome(false),
21     devpath(path), cmdpath(path + "/cmd"), notifyPath(path + "/notify1"),
      macroPath(path + "/notify2"),
22     _currentProfile(0), _currentMode(0), _model(
      KeyMap::NO_MODEL),
23     lastAutoSave(QDateTime::currentMSecsSinceEpoch()),
24     _hwProfile(0), prevProfile(0), prevMode(0),
25     cmd(cmdpath), notifyNumber(1), macroNumber(2),
      _needsSave(false)
26 {
27     memset(iState, 0, sizeof(iState));
28     memset(hwLoading, 0, sizeof(hwLoading));
29
30     // Get the features, model, serial number, FW version (if available), and poll rate (if available) from
       /dev nodes
31     QFile ftpath(path + "/features"), mpath(path + "/model"), spath(path + "/serial"), fwpath(path + "
      /fwversion"), ppath(path + "/pollrate");
32     if(ftpath.open(QIODevice::ReadOnly)){
33         features = ftpath.read(1000);
34         features = features.trimmed();
35         ftpath.close();
36         // Read model from features (first word: vendor, second word: product)
37         QStringList list = features.split(" ");
38         if(list.length() < 2)
39             return;
40         _model = KeyMap::getModel(list[1]);
41         if(_model == KeyMap::NO_MODEL)
42             return;
43     } else
44         // Bail if features aren't readable
45         return;
46     if(features.contains("monochrome"))
47         monochrome = true;
48     if(mpath.open(QIODevice::ReadOnly)){
49         usbModel = mpath.read(100);
50         usbModel = usbModel.remove("Corsair").remove("Gaming").remove("Keyboard").remove("
      Mouse").remove("Bootloader").trimmed();
51         mpath.close();
52     }
53     if(usbModel == "")
54         usbModel = "Keyboard";
55     if(spath.open(QIODevice::ReadOnly)){
56         usbSerial = spath.read(100);
57         usbSerial = usbSerial.trimmed().toUpper();
58         spath.close();
59     }
60     if(usbSerial == "")
61         usbSerial = "Unknown-" + usbModel;
62     if(features.contains("fwversion") && fwpath.open(QIODevice::ReadOnly)){
63         firmware = fwpath.read(100);
64         firmware = QString::number(firmware.trimmed().toInt() / 100., 'f', 2);
65         fwpath.close();
66     }
67     if(features.contains("pollrate") && ppath.open(QIODevice::ReadOnly)){
68         pollrate = ppath.read(100);
69         pollrate = pollrate.trimmed();
70         ppath.close();
71     }
72
73     prefsPath = "Devices/" + usbSerial;
74
75     hwModeCount = (_model == KeyMap::K95) ? 3 : 1;
76     // Open cmd in non-blocking mode so that it doesn't lock up if nothing is reading
77     // (e.g. if the daemon crashed and didn't clean up the node)
78     int fd = open(cmdpath.toLatin1().constData(), O_WRONLY | O_NONBLOCK);
79     if(!cmd.open(fd, QIODevice::WriteOnly, QFileDevice::AutoCloseHandle))
80         return;
```

```
81
82      // Find an available notification node (if none is found, take notify1)
83      {
84          QMutexLocker locker(&notifyPathMutex);
85          for(int i = 1; i < 10; i++){
86              QString notify = QString(path + "/notify%1").arg(i);
87              if(!QFile::exists(notify) && !notifyPaths.contains(notify)){
88                  notifyNumber = i;
89                  notifyPath = notify;
90                  break;
91              }
92          }
93          notifyPaths.insert(notifyPath);
94      }
95      cmd.write(QString("notifyon %1\n").arg(notifyNumber).toLatin1());
96      cmd.flush();
97
98      // Again, find an available notification node for macro definition
99      // (if none is found, take notify2)
100      {
101          QMutexLocker locker(&notifyPathMutex);
102          for(int i = 1; i < 10; i++){
103              QString notify = QString(path + "/notify%1").arg(i);
104              if(!QFile::exists(notify) && !notifyPaths.contains(notify)){
105                  macroNumber = i;
106                  macroPath = notify;
107                  break;
108              }
109          }
110          notifyPaths.insert(notifyPath);
111      }
112      // Activate device, apply settings, and ask for hardware profile
113      cmd.write(QString("fps %1\n").arg(_frameRate).toLatin1());
114      cmd.write(QString("dither %1\n").arg(static_cast<int>(_dither)).toLatin1());
115      cmd.write(QString("\ndelay %1\n").arg(_delay? "on" : "off").toLatin1());
116  #ifdef Q_OS_MACX
117      // Write ANSI/ISO flag to daemon (OSX only)
118      cmd.write("layout ");
119      cmd.write(KeyMap::isISO(_layout) ? "iso" : "ansi");
120      // Also OSX only: scroll speed and mouse acceleration
121      cmd.write(QString("accel %1\n").arg(QString(_mouseAccel ? "on" : "off")).toLatin1());
122      cmd.write(QString("scrollspeed %1\n").arg(_scrollSpeed).toLatin1());
123  #endif
124      cmd.write(QString("\nactive\n@%1 get :hwprofileid").arg(notifyNumber).toLatin1());
125      hwLoading[0] = true;
126      for(int i = 0; i < hwModeCount; i++){
127          cmd.write(QString(" mode %1 get :hwid").arg(i + 1).toLatin1());
128          hwLoading[i + 1] = true;
129      }
130      // Ask for current indicator and key state
131      cmd.write(" get :i :keys\n");
132      cmd.flush();
133
134      emit infoUpdated();
135      activeDevices.insert(this);
136
137      // Start a separate thread to read from the notification node
138      start();
139  }
```

Here is the call graph for this function:



12.21.3 Member Function Documentation

12.21.3.1 void Kb::appendProfile (KbProfile * *newProfile*) [inline]

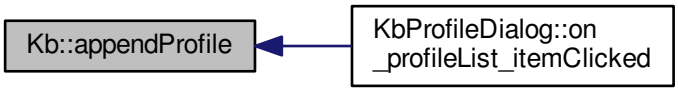
Definition at line 60 of file kb.h.

References _needsSave, and _profiles.

Referenced by KbProfileDialog::on_profileList_itemClicked().

```
60 { _needsSave = true; _profiles.append(newProfile); }
```

Here is the caller graph for this function:



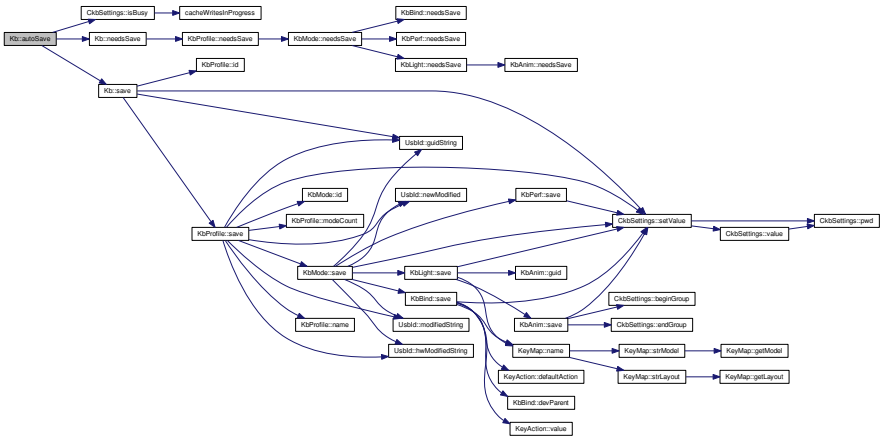
12.21.3.2 void Kb::autoSave ( ) [slot]

Definition at line 289 of file kb.cpp.

References CkbSettings::isBusy(), lastAutoSave, needsSave(), and save().

```
289                 {
290     quint64 now = QDateTime::currentMSecsSinceEpoch();
291     if(needsSave() && now >= lastAutoSave + 15 * 1000 && !
      CkbSettings::isBusy()){
292         save();
293         lastAutoSave = now;
294     }
295 }
```

Here is the call graph for this function:



12.21.3.3 KbBind* Kb::currentBind ( ) [inline]

Definition at line 67 of file kb.h.

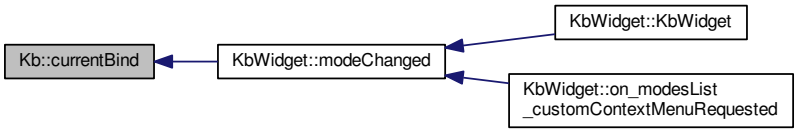
References _currentMode, and KbMode::bind().

Referenced by KbWidget::modeChanged().

```
67 { return !_currentMode ? 0 : _currentMode->bind(); }
```

Here is the call graph for this function:

Here is the caller graph for this function:



12.21.3.4 KbLight* Kb::currentLight ( ) [inline]

Definition at line 66 of file kb.h.

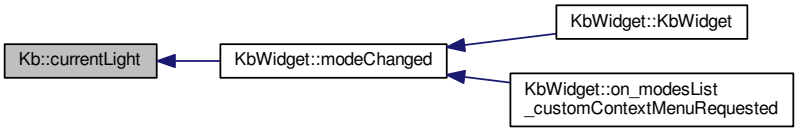
References _currentMode, and KbMode::light().

Referenced by KbWidget::modeChanged().

```
66 { return !_currentMode ? 0 : _currentMode->light(); }
```

Here is the call graph for this function:

Here is the caller graph for this function:



12.21.3.5 KbMode* Kb::currentMode ( ) [inline]

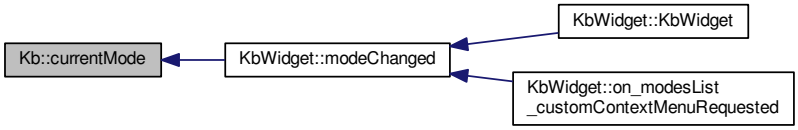
Definition at line 65 of file kb.h.

References _currentMode.

Referenced by KbWidget::modeChanged().

```
65 { return _currentMode; }
```

Here is the caller graph for this function:



12.21.3.6 KbPerf* Kb::currentPerf ( ) [inline]

Definition at line 68 of file kb.h.

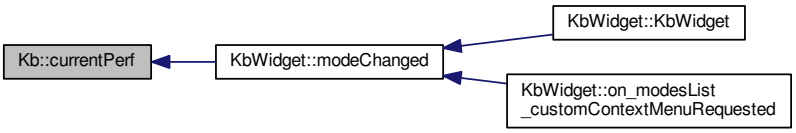
References _currentMode, and KbMode::perf().

Referenced by KbWidget::modeChanged().

```
68 { return !_currentMode ? 0 : _currentMode->perf(); }
```

Here is the call graph for this function:

Here is the caller graph for this function:



12.21.3.7 KbProfile* Kb::currentProfile ( ) [inline]

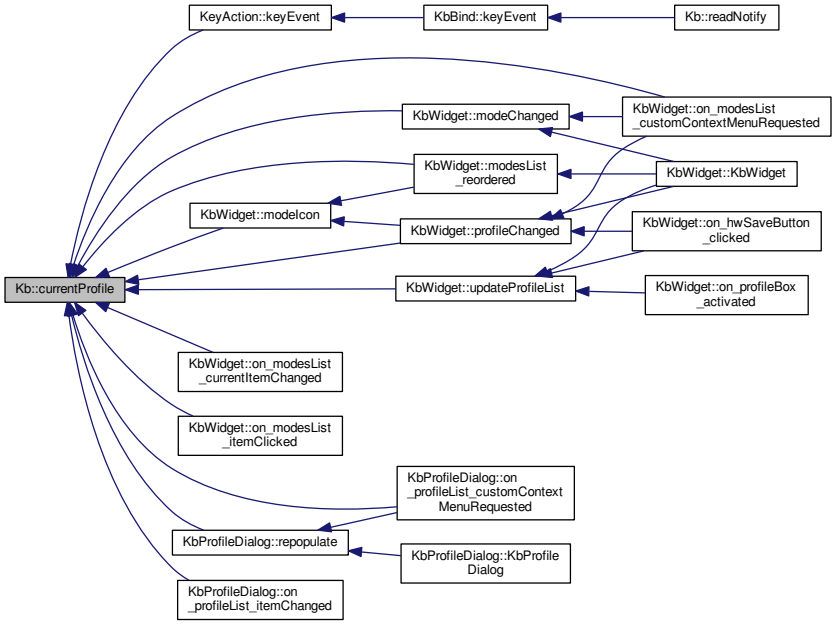
Definition at line 56 of file kb.h.

References _currentProfile.

Referenced by KeyAction::keyEvent(), KbWidget::modeChanged(), KbWidget::modeIcon(), KbWidget::modesList_reordered(), KbWidget::on_modesList_currentItemChanged(), KbWidget::on_modesList_customContextMenuRequested(), KbWidget::on_modesList_itemClicked(), KbProfileDialog::on_profileList_customContextMenuRequested(), KbProfileDialog::on_profileList_itemChanged(), KbWidget::profileChanged(), KbProfileDialog::repopulate(), and KbWidget::updateProfileList().

```
56 { return _currentProfile; }
```

Here is the caller graph for this function:



12.21.3.8 void Kb::deleteHw ( ) [private], [slot]

Definition at line 441 of file kb.cpp.

References _hwProfile.

Referenced by hwProfile().

```
441                 {
442     disconnect(_hwProfile, SIGNAL(destroyed()), this, SLOT(deleteHw()));
443     _hwProfile = 0;
444 }
```

Here is the caller graph for this function:

12.21.3.9 void Kb::deletePrevious ( ) [private], [slot]

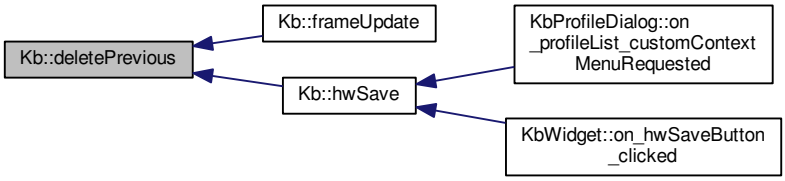
Definition at line 426 of file kb.cpp.

References prevMode.

Referenced by frameUpdate(), and hwSave().

```
426                     {
427     disconnect(prevMode, SIGNAL(destroyed()), this, SLOT(deletePrevious()));
428     prevMode = 0;
429 }
```

Here is the caller graph for this function:



12.21.3.10 static bool Kb::dither ( ) [inline], [static]

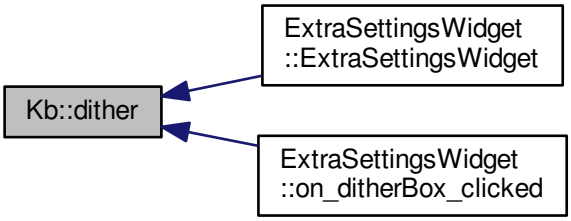
Definition at line 33 of file kb.h.

References _dither.

Referenced by ExtraSettingsWidget::ExtraSettingsWidget(), and ExtraSettingsWidget::on_ditherBox_clicked().

```
33 { return _dither; }
```

Here is the caller graph for this function:



12.21.3.11 void Kb::dither (bool *newDither*) [static]

Definition at line 204 of file kb.cpp.

References _dither, activeDevices, and cmd.

```
204                                {
205     if(newDither == _dither)
206         return;
207     _dither = newDither;
208     // Update all devices
209     foreach(Kb* kb, activeDevices){
210         kb->cmd.write(QString("dither %1\n").arg(static_cast<int>(newDither)).toLatin1());
211         kb->cmd.flush();
212     }
213 }
```

12.21.3.12 KbProfile* Kb::find (const QUuid & *id*) [inline]

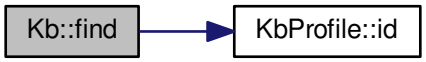
Definition at line 62 of file kb.h.

References _profiles, UsbId::guid, and KbProfile::id().

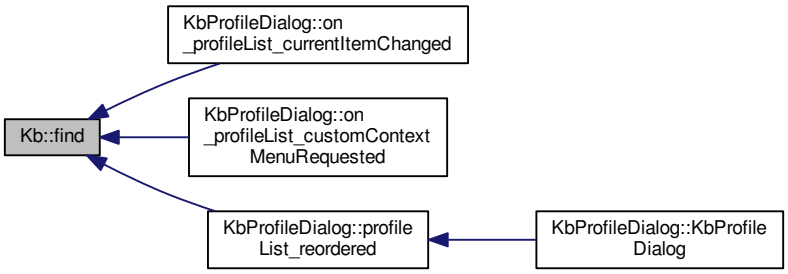
Referenced by KbProfileDialog::on_profileList_currentItemChanged(), KbProfileDialog::on_profileList_customContextMenuRequested(), and KbProfileDialog::profileList_reordered().

```
62 { foreach(KbProfile* profile, _profiles) { if(profile->id().
      guid == id) return profile; } return 0; }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.21.3.13 static int Kb::frameRate ( ) [inline], [static]

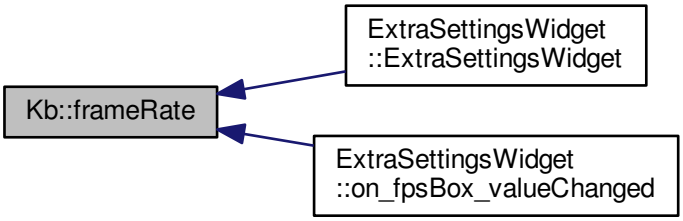
Definition at line 27 of file kb.h.

References _frameRate.

Referenced by ExtraSettingsWidget::ExtraSettingsWidget(), and ExtraSettingsWidget::on_fpsBox_valueChanged().

```
27 { return _frameRate; }
```

Here is the caller graph for this function:



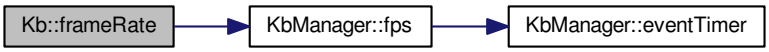
12.21.3.14 void Kb::frameRate (int *newFrameRate*) [static]

Definition at line 165 of file kb.cpp.

References _frameRate, activeDevices, cmd, and KbManager::fps().

```
165                                {
166     KbManager::fps(newFrameRate);
167     // If the rate has changed, send to all devices
168     if(newFrameRate == _frameRate)
169         return;
170     _frameRate = newFrameRate;
171     foreach(Kb* kb, activeDevices){
172         kb->cmd.write(QString("fps %1\n").arg(newFrameRate).toLatin1());
173         kb->cmd.flush();
174     }
175 }
```

Here is the call graph for this function:



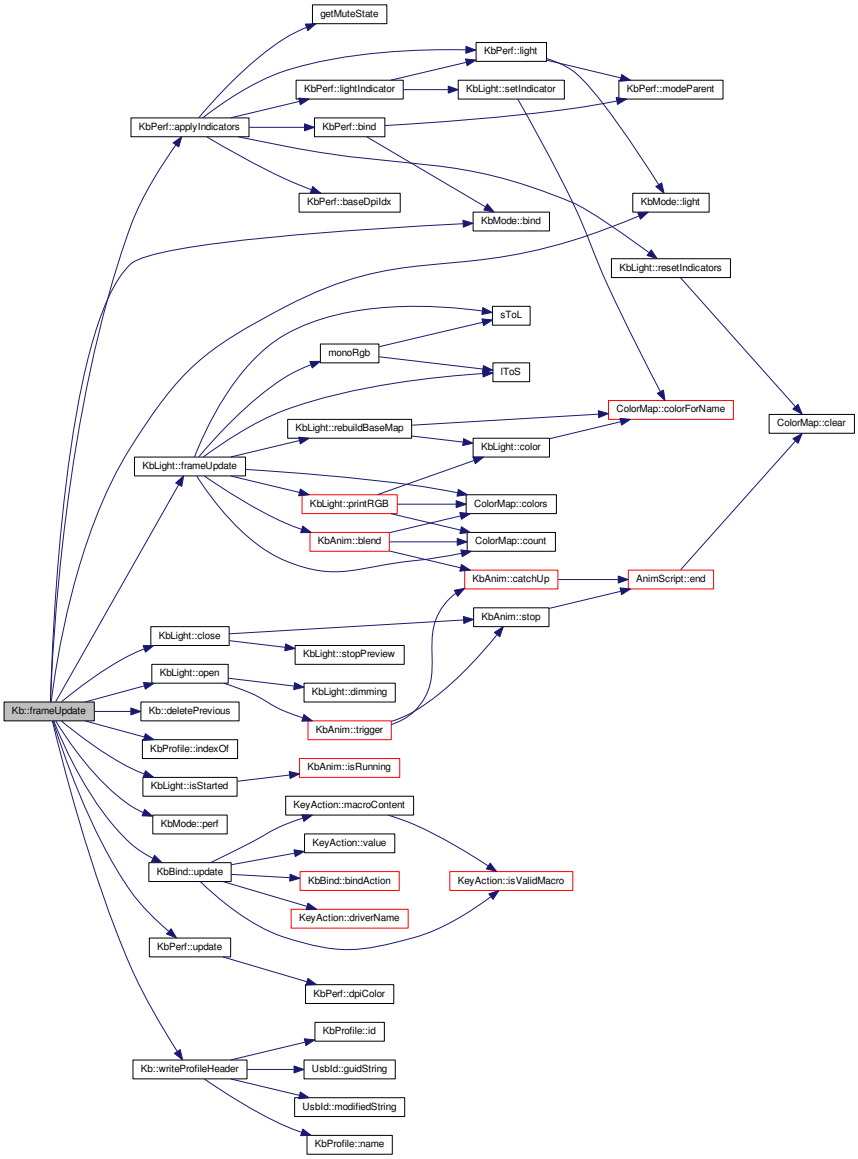
12.21.3.15 void Kb::frameUpdate ( ) [slot]

Definition at line 375 of file kb.cpp.

References _currentMode, _currentProfile, KbPerf::applyIndicators(), KbMode::bind(), KbLight::close(), cmd, deletePrevious(), KbLight::frameUpdate(), KbProfile::indexOf(), KbLight::isStarted(), iState, KbMode::light(), monochrome, notifyNumber, KbLight::open(), KbMode::perf(), prevMode, prevProfile, KbBind::update(), KbPerf::update(), and writeProfileHeader().

```
375                     {
376     // Advance animation frame
377     if(!_currentMode)
378         return;
379     KbLight* light = _currentMode->light();
380     KbBind* bind = _currentMode->bind();
381     KbPerf* perf = _currentMode->perf();
382     if(!light->isStarted()){
383         // Don't do anything until the animations are started
384         light->open();
385         return;
386     }
387
388     // Stop animations on the previously active mode (if any)
389     bool changed = false;
390     if(prevMode != _currentMode){
391         if(prevMode){
392             prevMode->light()->close();
393             disconnect(prevMode, SIGNAL(destroyed()), this, SLOT(
    deletePrevious()));
394         }
395         prevMode = _currentMode;
396         connect(prevMode, SIGNAL(destroyed()), this, SLOT(deletePrevious()));
397         changed = true;
398     }
399
400     // If the profile has changed, update it
401     if(prevProfile != _currentProfile){
402         writeProfileHeader();
403         cmd.write(" ");
404         prevProfile = _currentProfile;
405     }
406
407     // Update current mode
408     int index = _currentProfile->indexOf(_currentMode);
409     // ckb-daemon only has 6 modes: 3 hardware, 3 non-hardware. Beyond mode six, switch back to four.
410     // e.g. 1, 2, 3, 4, 5, 6, 4, 5, 6, 4, 5, 6 ...
411     if(index >= 6)
412         index = 3 + index % 3;
413
414     // Send lighting/binding to driver
415     cmd.write(QString("mode %1 switch ").arg(index + 1).toLatin1());
416     perf->applyIndicators(index, iState);
417     light->frameUpdate(cmd, monochrome);
418     cmd.write(QString("\n@%1 ").arg(notifyNumber).toLatin1());
419     bind->update(cmd, changed);
420     cmd.write(" ");
421     perf->update(cmd, notifyNumber, changed, true);
422     cmd.write("\n");
423     cmd.flush();
424 }
```

Here is the call graph for this function:



12.21.3.16 void Kb::fwUpdate (const QString & *path*)

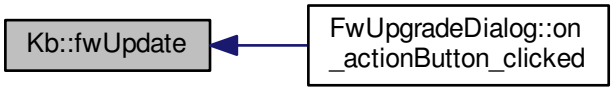
Definition at line 365 of file kb.cpp.

References cmd, fwUpdPath, and notifyNumber.

Referenced by FwUpgradeDialog::on_actionButton_clicked().

```
365                                     {
366     fwUpdPath = path;
367     // Write the active command to ensure it's not ignored
368     cmd.write("active");
369     cmd.write(QString(" @%1 ").arg(notifyNumber).toLatin1());
370     cmd.write("fwupdate ");
371     cmd.write(path.toLatin1());
372     cmd.write("\n");
373 }
```

Here is the caller graph for this function:



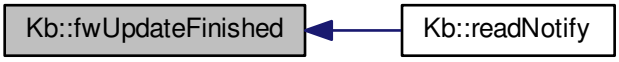
12.21.3.17 void Kb::fwUpdateFinished (bool *succeeded*) [signal]

Definition at line 277 of file moc_kb.cpp.

Referenced by readNotify().

```
278 {
279     void *_a[] = { Q_NULLPTR, const_cast<void*>(reinterpret_cast<const void*>(&_t1)) };
280     QMetaObject::activate(this, &staticMetaObject, 7, _a);
281 }
```

Here is the caller graph for this function:



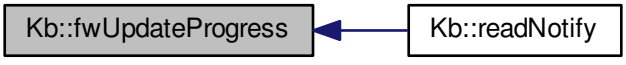
12.21.3.18 void Kb::fwUpdateProgress (int *current*, int *total*) [signal]

Definition at line 270 of file moc_kb.cpp.

Referenced by readNotify().

```
271 {
272     void *_a[] = { Q_NULLPTR, const_cast<void*>(reinterpret_cast<const void*>(&_t1)), const_cast<void*>(
      reinterpret_cast<const void*>(&_t2)) };
273     QMetaObject::activate(this, &staticMetaObject, 6, _a);
274 }
```

Here is the caller graph for this function:



12.21.3.19 KeyMap Kb::getKeyMap ( ) [private]

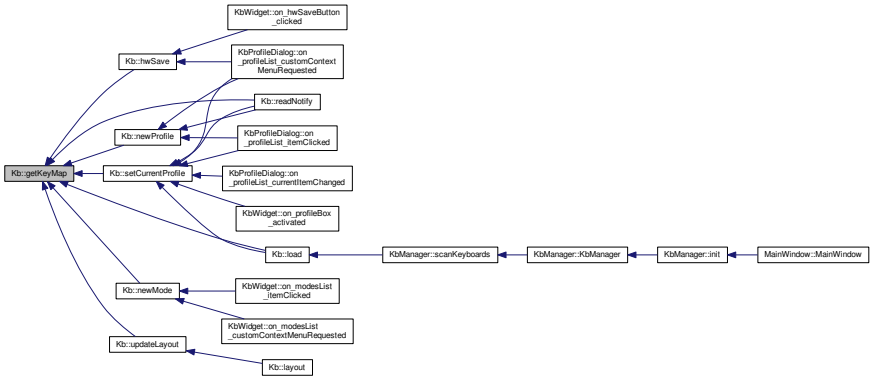
Definition at line 756 of file kb.cpp.

References _layout, and _model.

Referenced by hwSave(), load(), newMode(), newProfile(), readNotify(), setCurrentProfile(), and updateLayout().

```
756                         {
757     return KeyMap(_model, _layout);
758 }
```

Here is the caller graph for this function:



12.21.3.20 int Kb::getMacroNumber ( ) [inline]

getMacroNumber returns the macroNumber, which we have saved in the constructor. For usage with macro definions, this param must only be readable. So there is no setter.

Returns

The Number is returned as int.

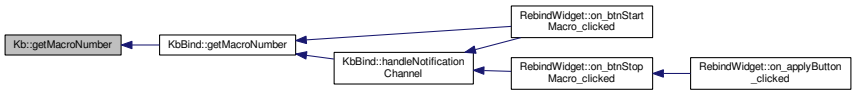
Definition at line 97 of file kb.h.

References macroNumber.

Referenced by KbBind::getMacroNumber().

```
97 { return macroNumber; }
```

Here is the caller graph for this function:



12.21.3.21 QString Kb::getMacroPath ( ) [inline]

Returns

The absolute path as String

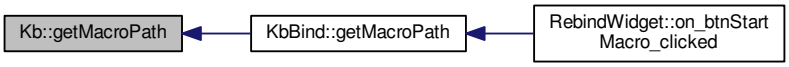
Definition at line 106 of file kb.h.

References macroPath.

Referenced by KbBind::getMacroPath().

```
106 { return macroPath; }
```

Here is the caller graph for this function:



12.21.3.22 KbProfile* Kb::hwProfile ( ) [inline]

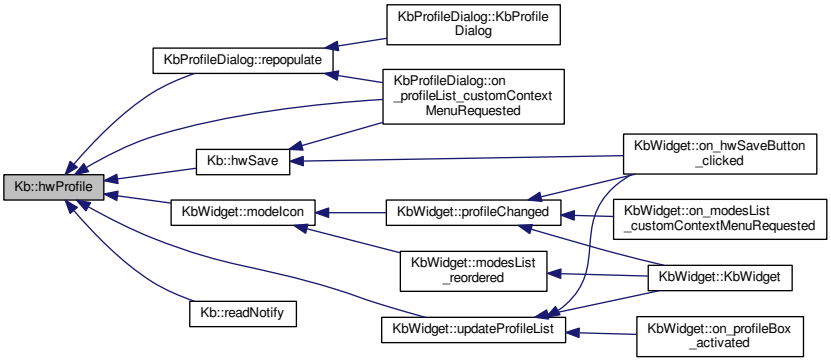
Definition at line 46 of file kb.h.

References _hwProfile.

Referenced by hwSave(), KbWidget::modeIcon(), KbProfileDialog::on_profileList_customContextMenuRequested(), readNotify(), KbProfileDialog::repopulate(), and KbWidget::updateProfileList().

```
46 { return _hwProfile; }
```

Here is the caller graph for this function:



12.21.3.23 void Kb::hwProfile (KbProfile * *newHwProfile*)

Definition at line 431 of file kb.cpp.

References _hwProfile, and deleteHw().

```
431                                         {
432     if(_hwProfile == newHwProfile)
433         return;
434     if(_hwProfile)
435         disconnect(_hwProfile, SIGNAL(destroyed()), this, SLOT(
      deleteHw()));
436     _hwProfile = newHwProfile;
437     if(_hwProfile)
438         connect(_hwProfile, SIGNAL(destroyed()), this, SLOT(deleteHw()));
439 }
```

Here is the call graph for this function:

12.21.3.24 void Kb::hwSave ( )

Definition at line 297 of file kb.cpp.

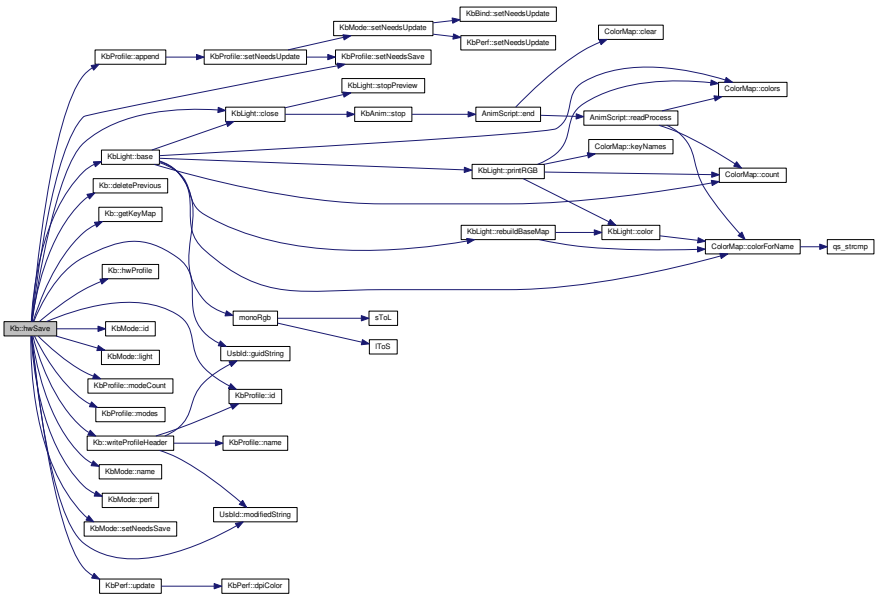
References _currentMode, _currentProfile, _hwProfile, KbProfile::append(), KbLight::base(), KbLight::close(), cmd, deletePrevious(), getKeyMap(), UsbId::guidString(), hwModeCount, UsbId::hwModified, hwProfile(), KbProfile::id(), KbMode::id(), KbMode::light(), KbProfile::modeCount(), KbProfile::modes(), UsbId::modified, UsbId::modifiedString(), monochrome, KbMode::name(), notifyNumber, KbMode::perf(), prevMode, KbProfile::setNeedsSave(), KbMode::setNeedsSave(), KbPerf::update(), and writeProfileHeader().

Referenced by KbWidget::on_hwSaveButton_clicked(), and KbProfileDialog::on_profileList_customContextMenuRequested().

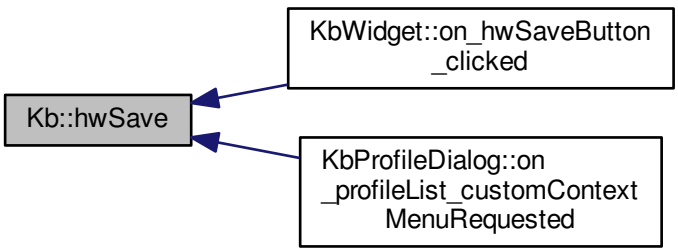
```
297                 {
298     if(!_currentProfile)
299         return;
300     // Close active lighting (if any)
301     if(prevMode){
302         prevMode->light()->close();
303         deletePrevious();
304     }
305     hwProfile(_currentProfile);
306     _hwProfile->id().hwModified = _hwProfile->id().
      modified;
307     _hwProfile->setNeedsSave();
308     // Re-send the current profile from scratch to ensure consistency
309     writeProfileHeader();
310     // Make sure there are enough modes
311     while(_currentProfile->modeCount() < hwModeCount)
312         _currentProfile->append(new KbMode(this,
      getKeyMap()));
313     // Write only the base colors of each mode, no animations
314     for(int i = 0; i < hwModeCount; i++){
315         KbMode* mode = _currentProfile->modes()[i];
316         cmd.write(QString("\nmode %1").arg(i + 1).toLatin1());
317         KbLight* light = mode->light();
318         KbPerf* perf = mode->perf();
319         if(mode == _currentMode)
320             cmd.write(" switch");
321         // Write the mode name and ID
322         cmd.write(" name ");
323         cmd.write(QUrl::toPercentEncoding(mode->name()));
324         cmd.write(" id ");
325         cmd.write(mode->id().guidString().toLatin1());
326         cmd.write(" ");
327         cmd.write(mode->id().modifiedString().toLatin1());
328         cmd.write(" ");
329         // Write lighting and performance
330         light->base(cmd, true, monochrome);
```

```
331         cmd.write(" ");
332         perf->update(cmd, notifyNumber, true, false);
333         // Update mode ID
334         mode->id().hwModified = mode->id().modified;
335         mode->setNeedsSave();
336     }
337     cmd.write("\n");
338
339     // Save the profile to memory
340     cmd.write("hwsave\n");
341     cmd.flush();
342 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.21.3.25 int Kb::indexOf (KbProfile * *profile*) [inline]

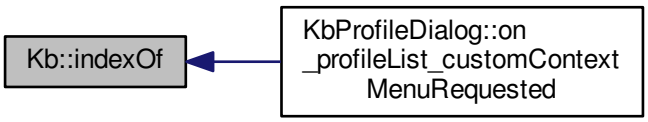
Definition at line 61 of file kb.h.

References _profiles.

Referenced by KbProfileDialog::on_profileList_customContextMenuRequested().

```
61 { return _profiles.indexOf(profile); }
```

Here is the caller graph for this function:



12.21.3.26 void Kb::infoUpdated ( ) [signal]

Definition at line 233 of file moc_kb.cpp.

Referenced by Kb(), load(), and updateLayout().

```
234 {
235     QMetaObject::activate(this, &staticMetaObject, 0, Q_NULLPTR);
236 }
```

Here is the caller graph for this function:

12.21.3.27 bool Kb::isKeyboard ( ) const [inline]

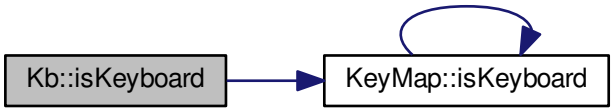
Definition at line 23 of file kb.h.

References _model, and KeyMap::isKeyboard().

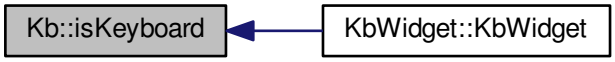
Referenced by KbWidget::KbWidget().

```
23 { return KeyMap::isKeyboard(_model); }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.21.3.28 bool Kb::isMouse ( ) const [inline]

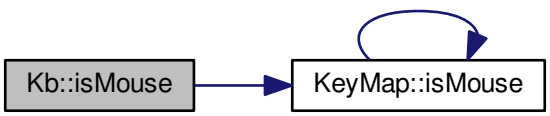
Definition at line 24 of file kb.h.

References _model, and KeyMap::isMouse().

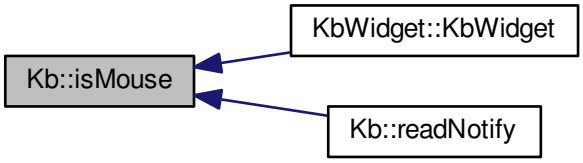
Referenced by KbWidget::KbWidget(), and readNotify().

```
24 { return KeyMap::isMouse(_model); }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.21.3.29 bool Kb::isOpen ( ) const [inline], [private]

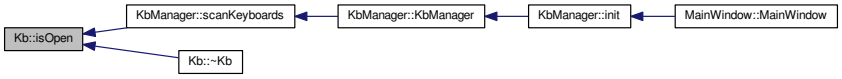
Definition at line 147 of file kb.h.

References cmd.

Referenced by KbManager::scanKeyboards(), and ~Kb().

```
147 { return cmd.isOpen(); }
```

Here is the caller graph for this function:



12.21.3.30 static KeyMap::Layout Kb::layout ( ) [inline], [static]

Definition at line 30 of file kb.h.

References _layout.

Referenced by LayoutDialog::LayoutDialog(), SettingsWidget::on_layoutBox_activated(), and SettingsWidget::~SettingsWidget().

```
30 { return _layout; }
```

Here is the caller graph for this function:

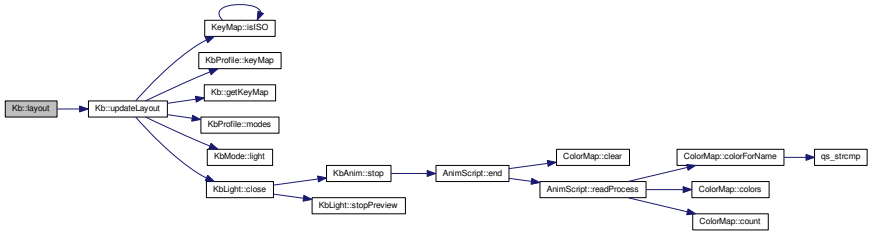
12.21.3.31 void Kb::layout (KeyMap::Layout *newLayout*) [static]

Definition at line 177 of file kb.cpp.

References _layout, activeDevices, KeyMap::NO_LAYOUT, and updateLayout().

```
177                                     {
178     if(newLayout == KeyMap::NO_LAYOUT || newLayout == _layout)
179         return;
180     _layout = newLayout;
181     // Update all devices
182     foreach(Kb* kb, activeDevices)
183         kb->updateLayout();
184 }
```

Here is the call graph for this function:



12.21.3.32 void Kb::load ( )

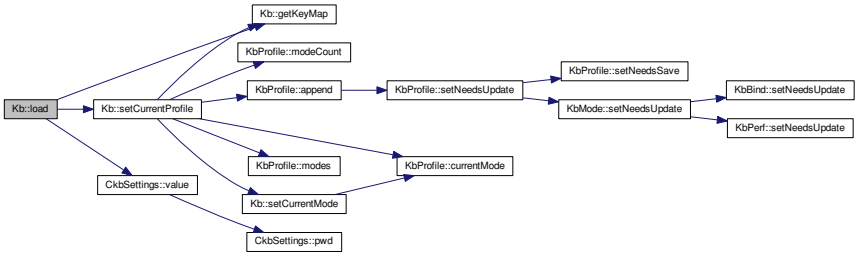
Definition at line 241 of file kb.cpp.

References _needsSave, _profiles, getKeyMap(), infoUpdated(), prefsPath, profileAdded(), setCurrentProfile(), and CkbSettings::value().

Referenced by KbManager::scanKeyboards().

```
241             {
242     if(prefsPath.isEmpty())
243         return;
244     _needsSave = false;
245     CkbSettings settings(prefsPath);
246     // Read profiles
247     KbProfile* newCurrentProfile = 0;
248     QString current = settings.value("CurrentProfile").toString().trimmed().toUpper();
249     foreach(QString guid, settings.value("Profiles").toString().split(" ")){
250         guid = guid.trimmed().toUpper();
251         if(guid != ""){
252             KbProfile* profile = new KbProfile(this, getKeyMap(), settings, guid
    );
253             _profiles.append(profile);
254             if(guid == current || !newCurrentProfile)
255                 newCurrentProfile = profile;
256         }
257     }
258     if(newCurrentProfile)
259         setCurrentProfile(newCurrentProfile);
260     else {
261         // If nothing was loaded, load the demo profile
262         QSettings demoSettings(":/txt/demoprofile.conf", QSettings::IniFormat, this);
263         CkbSettings cSettings(demoSettings);
264         KbProfile* demo = new KbProfile(this, getKeyMap(), cSettings, "
    {BA7FC152-2D51-4C26-A7A6-A036CC93D924}");
265         _profiles.append(demo);
266         setCurrentProfile(demo);
267     }
268
269     emit infoUpdated();
270     emit profileAdded();
271 }
```

Here is the call graph for this function:



Here is the caller graph for this function:

12.21.3.33 static bool Kb::macroDelay ( ) [inline], [static]

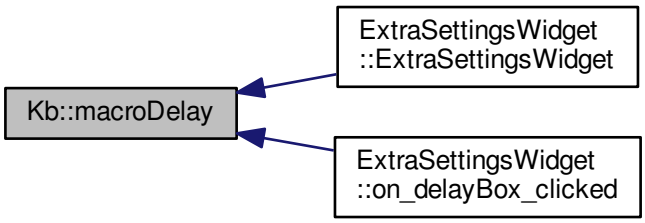
Definition at line 36 of file kb.h.

References _delay.

Referenced by ExtraSettingsWidget::ExtraSettingsWidget(), and ExtraSettingsWidget::on_delayBox_clicked().

```
36 { return _delay; }
```

Here is the caller graph for this function:



12.21.3.34 void Kb::macroDelay (bool *flag*) [static]

Parameters

flag	true: Switch on delay function, else switch off

Definition at line 787 of file kb.cpp.

References _delay, activeDevices, and cmd.

```
787                             {
788     _delay = flag;
789
790     foreach(Kb* kb, activeDevices){
791         kb->cmd.write(QString("\ndelay %1\n").arg(flag? "on" : "off").toLatin1());
792     }
793 }
```

12.21.3.35 bool Kb::matches (const QString & *path,* const QString & *serial*) [inline],[private]

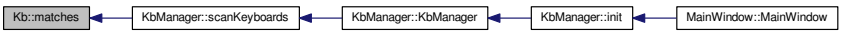
Definition at line 157 of file kb.h.

References devpath, and usbSerial.

Referenced by KbManager::scanKeyboards().

```
157 { return path.trimmed() == devpath.trimmed() && usbSerial == serial.trimmed().toUpper(); }
```

Here is the caller graph for this function:



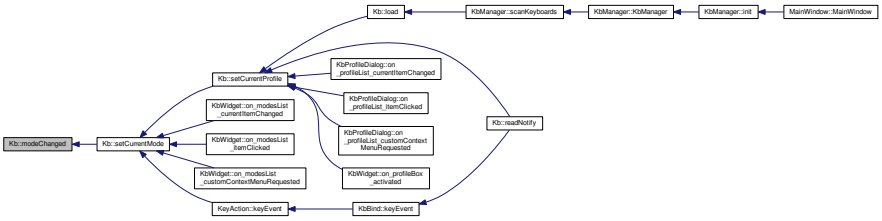
12.21.3.36 void Kb::modeChanged (bool *spontaneous*) [signal]

Definition at line 263 of file moc_kb.cpp.

Referenced by setCurrentMode().

```
264 {
265     void *_a[] = { Q_NULLPTR, const_cast<void*>(reinterpret_cast<const void*>(&_t1)) };
266     QMetaObject::activate(this, &staticMetaObject, 5, _a);
267 }
```

Here is the caller graph for this function:



12.21.3.37 KeyMap::Model Kb::model ( ) const [inline]

Definition at line 22 of file kb.h.

References _model.

```
22 { return _model; }
```

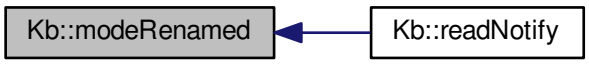
12.21.3.38 void Kb::modeRenamed ( ) [signal]

Definition at line 251 of file moc_kb.cpp.

Referenced by readNotify().

```
252 {
253     QMetaObject::activate(this, &staticMetaObject, 3, Q_NULLPTR);
254 }
```

Here is the caller graph for this function:



12.21.3.39 static bool Kb::mouseAccel ( ) [inline], [static]

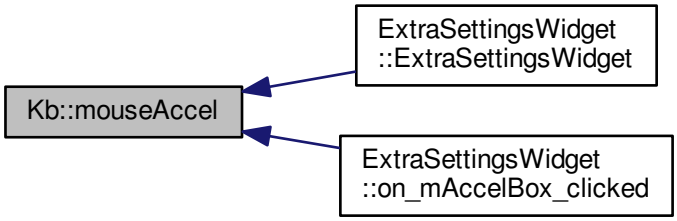
Definition at line 39 of file kb.h.

References _mouseAccel.

Referenced by ExtraSettingsWidget::ExtraSettingsWidget(), and ExtraSettingsWidget::on_mAccelBox_clicked().

```
39 { return _mouseAccel; }
```

Here is the caller graph for this function:



12.21.3.40 void Kb::mouseAccel (bool *newAccel*) [static]

Definition at line 215 of file kb.cpp.

References _mouseAccel, activeDevices, and cmd.

```
215                                 {
216     if(newAccel == _mouseAccel)
217         return;
218     _mouseAccel = newAccel;
219 #ifdef Q_OS_MACX
220     // Update all devices
221     foreach(Kb* kb, activeDevices){
222         kb->cmd.write(QString("accel %1\n").arg(QString(newAccel ? "on" : "off")).toLatin1());
223         kb->cmd.flush();
224     }
225 #endif
226 }
```

12.21.3.41 bool Kb::needsSave ( ) const

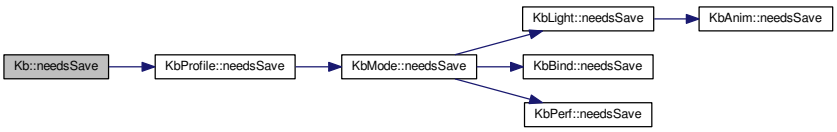
Definition at line 344 of file kb.cpp.

References _needsSave, _profiles, and KbProfile::needsSave().

Referenced by autoSave().

```
344                           {
345     if(_needsSave)
346         return true;
347     foreach(const KbProfile* profile, _profiles){
348         if(profile->needsSave())
349             return true;
350     }
351     return false;
352 }
```

Here is the call graph for this function:



Here is the caller graph for this function:

12.21.3.42 KbMode* Kb::newMode ( ) [inline]

Definition at line 79 of file kb.h.

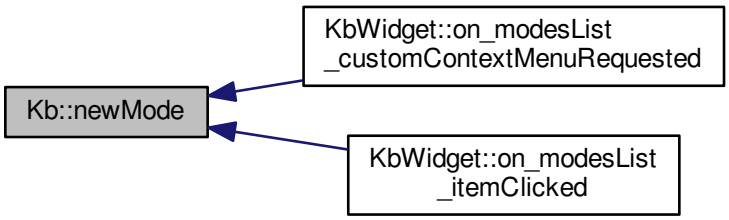
References getKeyMap().

Referenced by KbWidget::on_modesList_customContextMenuRequested(), and KbWidget::on_modesList_itemClicked().

```
79 { return new KbMode(this, getKeyMap()); }
```

Here is the call graph for this function:

Here is the caller graph for this function:



12.21.3.43 KbMode* Kb::newMode (KbMode * *other*) [inline]

Definition at line 80 of file kb.h.

References getKeyMap().

```
80 { return new KbMode(this, getKeyMap(), *other); }
```

Here is the call graph for this function:

12.21.3.44 KbProfile* Kb::newProfile ( ) [inline]

Definition at line 77 of file kb.h.

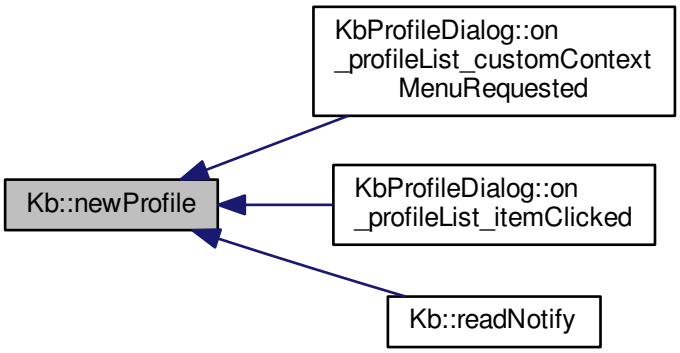
References getKeyMap().

Referenced by KbProfileDialog::on_profileList_customContextMenuRequested(), KbProfileDialog::on_profileList_itemClicked(), and readNotify().

```
77 { return new KbProfile(this, getKeyMap()); }
```

Here is the call graph for this function:

Here is the caller graph for this function:



12.21.3.45 KbProfile* Kb::newProfile (KbProfile * *other*) [inline]

Definition at line 78 of file kb.h.

References getKeyMap().

```
78 { return new KbProfile(this, getKeyMap(), *other); }
```

Here is the call graph for this function:

12.21.3.46 void Kb::profileAdded ( ) [signal]

Definition at line 239 of file moc_kb.cpp.

Referenced by load(), and readNotify().

```
240 {
241     QMetaObject::activate(this, &staticMetaObject, 1, Q_NULLPTR);
242 }
```

Here is the caller graph for this function:

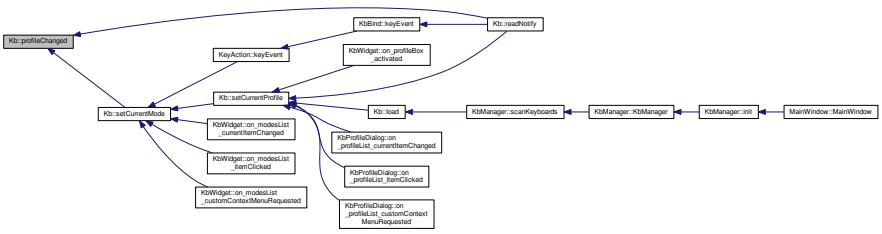
12.21.3.47 void Kb::profileChanged ( ) [signal]

Definition at line 257 of file moc_kb.cpp.

Referenced by readNotify(), and setCurrentMode().

```
258 {
259     QMetaObject::activate(this, &staticMetaObject, 4, Q_NULLPTR);
260 }
```

Here is the caller graph for this function:



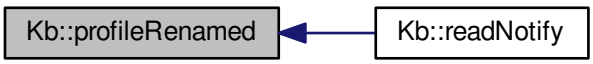
12.21.3.48 void Kb::profileRenamed ( ) [signal]

Definition at line 245 of file moc_kb.cpp.

Referenced by readNotify().

```
246 {
247     QMetaObject::activate(this, &staticMetaObject, 2, Q_NULLPTR);
248 }
```

Here is the caller graph for this function:



12.21.3.49 const QList<KbProfile*>& Kb::profiles ( ) const [inline]

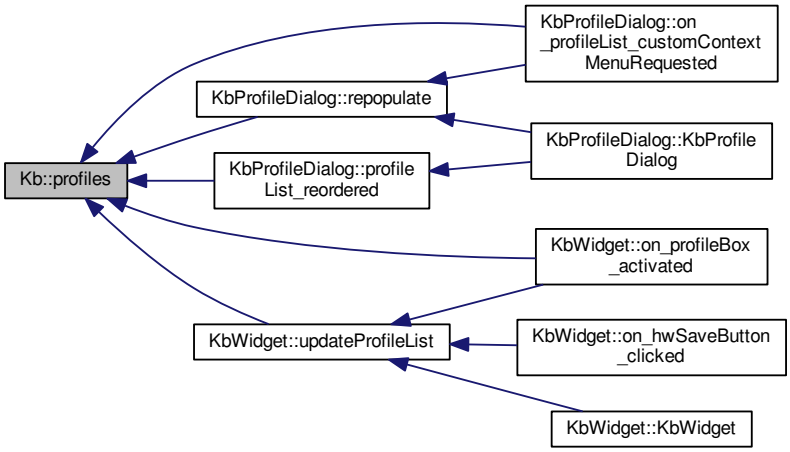
Definition at line 58 of file kb.h.

References _profiles.

Referenced by KbWidget::on_profileBox_activated(), KbProfileDialog::on_profileList_customContextMenuRequested(), KbProfileDialog::profileList_reordered(), KbProfileDialog::repopulate(), and KbWidget::updateProfileList().

```
58 { return _profiles; }
```

Here is the caller graph for this function:



12.21.3.50 void Kb::profiles (const QList< KbProfile * > & *newProfiles*) [inline]

Definition at line 59 of file kb.h.

References _needsSave, and _profiles.

```
59 { _needsSave = true; _profiles = newProfiles; }
```

12.21.3.51 void Kb::readNotify (QString *line*) [private], [slot]

Definition at line 471 of file kb.cpp.

References _currentMode, _currentProfile, _hwProfile, _needsSave, _profiles, KbPerf::angleSnap(), KbLight::animKeypress(), KbProfile::append(), KbPerf::baseDpiIdx(), KbMode::bind(), cmd, KbLight::color(), KbPerf::dpi(), KbPerf::DPI_COUNT, KbPerf::dpiColor(), KbPerf::dpiEnabled(), fwUpdateFinished(), fwUpdateProgress(), fwUpdPath, getKeyMap(), UsbId::guid, hwLoading, HWMODE_MAX, hwModeCount, UsbId::hwModifiedString(), hwProfile(), KbProfile::id(), KbMode::id(), KbProfile::indexOf(), isMouse(), iState, KbBind::keyEvent(), KbPerf::liftHeight(), KbMode::light(), KbProfile::modeCount(), modeRenamed(), KbProfile::modes(), UsbId::modifiedString(), KbProfile::move(), KbProfile::name(), KbMode::name(), newProfile(), notifyNumber, KbMode::perf(), profileAdded(), profileChanged(), profileRenamed(), setCurrentProfile(), KbProfile::setNeedsSave(), and KbMode::setNeedsSave().

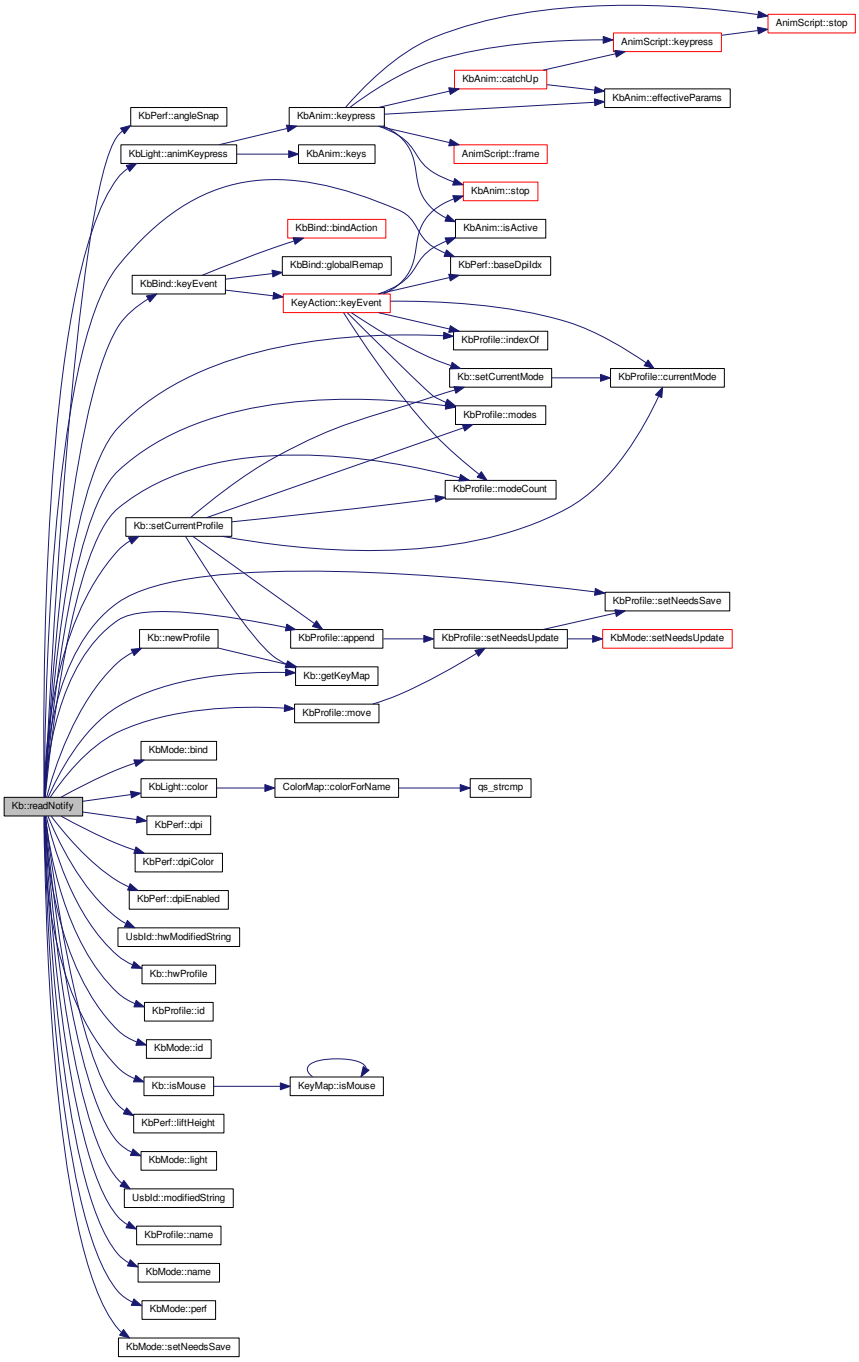
```
471                                {
472     QStringList components = line.trimmed().split(" ");
473     if(components.count() < 2)
474         return;
475     if(components[0] == "key"){
476         // Key event
477         QString key = components[1];
478         if(key.length() < 2)
479             return;
480         QString keyName = key.mid(1);
481         bool keyPressed = (key[0] == '+');
482         KbMode* mode = _currentMode;
483         if(mode){
484             mode->light()->animKeypress(keyName, keyPressed);
485             mode->bind()->keyEvent(keyName, keyPressed);
486         }
487     } else if(components[0] == "i"){
488         // Indicator event
489         QString i = components[1];
490         if(i.length() < 2)
491             return;
492         QString iName = i.mid(1);
493         bool on = (i[0] == '+');
494         if(iName == "num")
495             iState[0] = on;
496         else if(iName == "caps")
497             iState[1] = on;
498         else if(iName == "scroll")
499             iState[2] = on;
500     } else if(components[0] == "hwprofileid"){
501         // Hardware profile ID
502         if(components.count() < 3)
503             return;
504         // Find the hardware profile in the list of profiles
505         QString guid = components[1];
506         QString modified = components[2];
507         KbProfile* newProfile = 0;
508         foreach(KbProfile* profile, _profiles){
509             if(profile->id().guid == guid){
510                 newProfile = profile;
511                 break;
512             }
513         }
514         // If it wasn't found, create it
515         if(!newProfile){
516             newProfile = new KbProfile(this, getKeyMap(), guid, modified);
517             hwLoading[0] = true;
518             cmd.write(QString("@%1 get :hwprofilename\n").arg(notifyNumber).toLatin1());
519             cmd.flush();
520         } else {
521             // If it's been updated, fetch its name
522             if(newProfile->id().hwModifiedString() != modified){
523                 newProfile->id().modifiedString(modified);
524                 newProfile->id().hwModifiedString(modified);
525                 newProfile->setNeedsSave();
```

```
526                 if(hwLoading[0]){
527                     cmd.write(QString("@%1 get :hwprofilename\n").arg(
    notifyNumber).toLatin1());
528                     cmd.flush();
529                 }
530             } else {
531                 hwLoading[0] = false;
532             }
533         }
534         hwProfile(newProfile);
535         emit profileAdded();
536         if(_hwProfile == _currentProfile)
537             emit profileChanged();
538     } else if(components[0] == "hwprofilename"){
539         // Hardware profile name
540         QString name = QUrl::fromPercentEncoding(components[1].toUtf8());
541         if(!_hwProfile || !hwLoading[0])
542             return;
543         QString oldName = _hwProfile->name();
544         if(!(oldName.length() >= name.length() && oldName.left(name.length()) == name)){
545             // Don't change the name if it's a truncated version of what we already have
546             _hwProfile->name(name);
547             emit profileRenamed();
548         }
549     } else if(components[0] == "mode"){
550         // Mode-specific data
551         if(components.count() < 4)
552             return;
553         int mode = components[1].toInt() - 1;
554         if(components[2] == "hwid"){
555             if(components.count() < 5 || mode >= HWMODE_MAX || !
    _hwProfile)
556                 return;
557             // Hardware mode ID
558             QString guid = components[3];
559             QString modified = components[4];
560             // Look for this mode in the hardware profile
561             KbMode* hwMode = 0;
562             bool isUpdated = false;
563             foreach(KbMode* kbMode, _hwProfile->modes()){
564                 if(kbMode->id().guid == guid){
565                     hwMode = kbMode;
566                     if(kbMode->id().hwModifiedString() != modified){
567                         // Update modification time
568                         hwMode->id().modifiedString(modified);
569                         hwMode->id().hwModifiedString(modified);
570                         hwMode->setNeedsSave();
571                         isUpdated = true;
572                     } else {
573                         hwLoading[mode + 1] = false;
574                     }
575                     break;
576                 }
577             }
578             // If it wasn't found, add it
579             if(!hwMode){
580                 isUpdated = true;
581                 hwMode = new KbMode(this, getKeyMap(), guid, modified);
582                 _hwProfile->append(hwMode);
583                 // If the hardware profile now contains enough modes to be added to the list, do so
584                 if(!_profiles.contains(_hwProfile) &&
    _hwProfile->modeCount() >= hwModeCount){
585                     _profiles.append(_hwProfile);
586                     _needsSave = true;
587                     emit profileAdded();
588                     if(!_currentProfile)
589                         setCurrentProfile(_hwProfile);
590                 }
591             }
592             if(hwLoading[mode + 1] && isUpdated){
593                 // If the mode isn't in the right place, move it
594                 int index = _hwProfile->indexOf(hwMode);
595                 if(mode < _hwProfile->modeCount() && index != mode)
596                     _hwProfile->move(index, mode);
597                 // Fetch the updated data
598                 cmd.write(QString("@%1 mode %2 get :hwname :hwrgb").arg(
    notifyNumber).arg(mode + 1).toLatin1());
599                 if(isMouse())
600                     cmd.write(" :hwdpi :hwdpisel :hwlift :hwsnap");
601                 cmd.write("\n");
602                 cmd.flush();
603             }
604         } else if(components[2] == "hwname"){
605             // Mode name - update list
606             if(!_hwProfile || _hwProfile->modeCount() <= mode || mode >=
    HWMODE_MAX || !hwLoading[mode + 1])
607                 return;
```

```
608             KbMode* hwMode = _hwProfile->modes()[mode];
609             QString name = QUrl::fromPercentEncoding(components[3].toUtf8());
610             QString oldName = hwMode->name();
611             if(!(oldName.length() >= name.length() && oldName.left(name.length()) == name)){
612                 // Don't change the name if it's a truncated version of what we already have
613                 hwMode->name(name);
614                 if(_hwProfile == _currentProfile)
615                     emit modeRenamed();
616             }
617         } else if(components[2] == "hwrgb"){
618             // RGB - set mode lighting
619             if(!_hwProfile || _hwProfile->modeCount() <= mode || mode >=
    HWMODE_MAX || !hwLoading[mode + 1])
620                 return;
621             KbMode* kbmode = _hwProfile->modes()[mode];
622             KbLight* light = kbmode->light();
623             // Scan the input for colors
624             QColor lightColor = QColor();
625             for(int i = 3; i < components.count(); i++){
626                 QString comp = components[i];
627                 if(comp.indexOf(":") < 0){
628                     // No ":" - single hex constant
629                     bool ok;
630                     int rgb = comp.toInt(&ok, 16);
631                     if(ok)
632                         light->color(QColor::fromRgb((QRgb)rgb));
633                 } else {
634                     // List of keys ("a,b:xxxxxx"). Parse color first
635                     QStringList set = comp.split(":");
636                     bool ok;
637                     int rgb = set[1].toInt(&ok, 16);
638                     if(ok){
639                         QColor color = QColor::fromRgb((QRgb)rgb);
640                         // Parse keys
641                         QStringList keys = set[0].split(",");
642                         foreach(QString key, keys){
643                             if(key == "light")
644                                 // Extrapolate the Light key to the M-keys and Lock key, since those will
     be set to black on hwsave
645                                 lightColor = color;
646                             if(key.startsWith("dpi") && key.length() > 3){
647                                 // DPI levels go to the KbPerf object instead of KbLight
648                                 bool ok = false;
649                                 int index = key.mid(3).toInt(&ok);
650                                 if(ok)
651                                     kbmode->perf()->dpiColor(index, color);
652                                 continue;
653                             }
654                             light->color(key, color);
655                         }
656                     }
657                 }
658             }
659             if(lightColor.isValid()){
660                 light->color("mr", lightColor);
661                 light->color("m1", lightColor);
662                 light->color("m2", lightColor);
663                 light->color("m3", lightColor);
664                 light->color("lock", lightColor);
665             }
666         } else if(components[2] == "hwdpi"){
667             // DPI settings
668             if(!_hwProfile || _hwProfile->modeCount() <= mode || mode >=
    HWMODE_MAX || !hwLoading[mode + 1])
669                 return;
670             KbPerf* perf = _hwProfile->modes()[mode]->perf();
671             // Read the rest of the line as stage:x,y
672             foreach(QString comp, components.mid(3)){
673                 QStringList dpi = comp.split(':');
674                 if(dpi.length() != 2)
675                     continue;
676                 QStringList xy = dpi[1].split(',');
677                 int x, y;
678                 bool off = false;
679                 if(xy.length() < 2){
680                     // If the right side only has one parameter, set both X and Y
681                     if(xy[0] == "off")
682                         off = true;
683                     else
684                         x = y = xy[0].toInt();
685                 } else {
686                     x = xy[0].toInt();
687                     y = xy[1].toInt();
688                 }
689                 // Set DPI for this stage
690                 int index = dpi[0].toInt();
691                 if(off){
```

```
692                     perf->dpiEnabled(index, false);
693                     // If all DPIs have been disabled, turn them back on
694                     bool allOff = true;
695                     for(int i = 1; i < KbPerf::DPI_COUNT; i++){
696                         if(perf->dpiEnabled(i)){
697                             allOff = false;
698                             break;
699                         }
700                     }
701                     if(allOff){
702                         for(int i = 1; i < KbPerf::DPI_COUNT; i++)
703                             perf->dpiEnabled(i, true);
704                     }
705                 } else {
706                     perf->dpiEnabled(index, true);
707                     perf->dpi(index, QPoint(x, y));
708                 }
709             }
710         } else if(components[2] == "hwdpisel"){
711             // Hardware DPI selection (0...5)
712             if(!_hwProfile || _hwProfile->modeCount() <= mode || mode >=
    HWMODE_MAX || !hwLoading[mode + 1])
713                 return;
714             KbPerf* perf = _hwProfile->modes()[mode]->perf();
715             int idx = components[3].toInt();
716             if(idx < 1)
717                 idx = 1;
718             if(idx >= KbPerf::DPI_COUNT)
719                 idx = KbPerf::DPI_COUNT - 1;
720             perf->baseDpiIdx(idx);
721         } else if(components[2] == "hwlift"){
722             // Mouse lift height (1...5)
723             if(!_hwProfile || _hwProfile->modeCount() <= mode || mode >=
    HWMODE_MAX || !hwLoading[mode + 1])
724                 return;
725             KbPerf* perf = _hwProfile->modes()[mode]->perf();
726             perf->liftHeight((KbPerf::height)components[3].toInt());
727         } else if(components[3] == "hwsnap"){
728             // Mouse angle snapping ("on" or "off")
729             if(!_hwProfile || _hwProfile->modeCount() <= mode || mode >=
    HWMODE_MAX || !hwLoading[mode + 1])
730                 return;
731             KbPerf* perf = _hwProfile->modes()[mode]->perf();
732             perf->angleSnap(components[3] == "on");
733         }
734     } else if(components[0] == "fwupdate"){
735         // Firmware update progress
736         if(components.count() < 3)
737             return;
738         // Make sure path is the same
739         if(components[1] != fwUpdPath)
740             return;
741         QString res = components[2];
742         if(res == "invalid" || res == "fail")
743             emit fwUpdateFinished(false);
744         else if(res == "ok")
745             emit fwUpdateFinished(true);
746         else {
747             // "xx/yy" indicates progress
748             if(!res.contains("/"))
749                 return;
750             QStringList numbers = res.split("/");
751             emit fwUpdateProgress(numbers[0].toInt(), numbers[1].toInt());
752         }
753     }
754 }
```

Here is the call graph for this function:



12.21.3.52 void Kb::run ( ) [private]

Definition at line 446 of file kb.cpp.

References notifyPath, notifyPathMutex, and notifyPaths.

```
446             {
447     QFile notify(notifyPath);
448     // Wait a small amount of time for the node to open (100ms)
```

```
449     QThread::usleep(100000);
450     if(!notify.open(QIODevice::ReadOnly)){
451         // If it's still not open, try again before giving up (1s at a time, 10s total)
452         QThread::usleep(900000);
453         for(int i = 1; i < 10; i++){
454             if(notify.open(QIODevice::ReadOnly))
455                 break;
456             QThread::sleep(1);
457         }
458         if(!notify.isOpen())
459             return;
460     }
461     // Read data from notification node
462     QByteArray line;
463     while(notify.isOpen() && (line = notify.readLine()).length() > 0){
464         QString text = QString::fromUtf8(line);
465         metaObject()->invokeMethod(this, "readNotify", Qt::QueuedConnection, Q_ARG(QString, text));
466     }
467     QMutexLocker locker(&notifyPathMutex);
468     notifyPaths.remove(notifyPath);
469 }
```

12.21.3.53 void Kb::save ( )

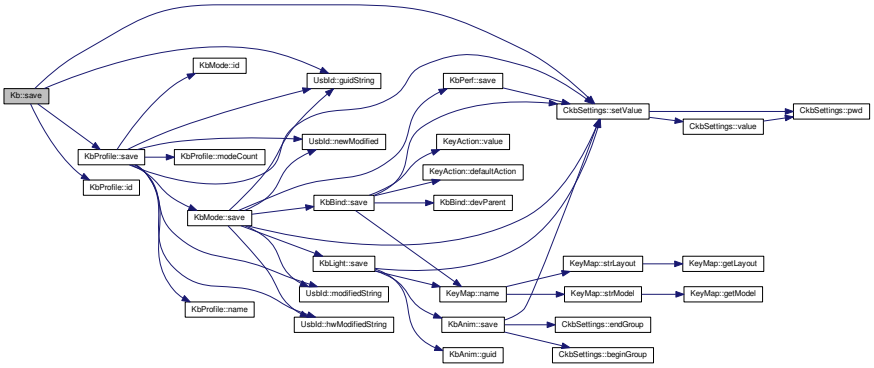
Definition at line 273 of file kb.cpp.

References _currentProfile, _needsSave, _profiles, UsbId::guidString(), KbProfile::id(), prefsPath, KbProfile::save(), and CkbSettings::setValue().

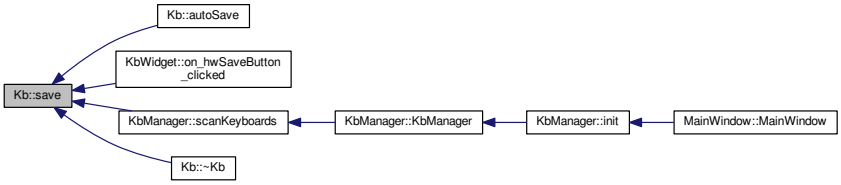
Referenced by autoSave(), KbWidget::on_hwSaveButton_clicked(), KbManager::scanKeyboards(), and ~Kb().

```
273             {
274     if(prefsPath.isEmpty())
275         return;
276     _needsSave = false;
277     CkbSettings settings(prefsPath, true);
278     QString guids, currentGuid;
279     foreach(KbProfile* profile, _profiles){
280         guids.append(" " + profile->id().guidString());
281         if(profile == _currentProfile)
282             currentGuid = profile->id().guidString();
283         profile->save(settings);
284     }
285     settings.setValue("CurrentProfile", currentGuid);
286     settings.setValue("Profiles", guids.trimmed());
287 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.21.3.54 static int Kb::scrollSpeed ( ) [inline], [static]

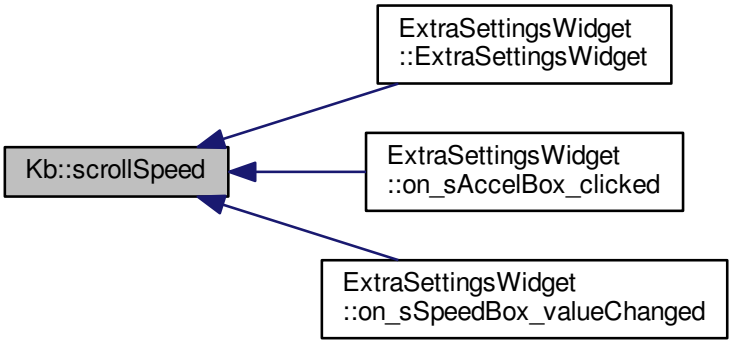
Definition at line 42 of file kb.h.

References _scrollSpeed.

Referenced by ExtraSettingsWidget::ExtraSettingsWidget(), ExtraSettingsWidget::on_sAccelBox_clicked(), and ExtraSettingsWidget::on_sSpeedBox_valueChanged().

```
42 { return _scrollSpeed; }
```

Here is the caller graph for this function:



12.21.3.55 void Kb::scrollSpeed (int *newSpeed*) [static]

Definition at line 228 of file kb.cpp.

References _scrollSpeed, activeDevices, and cmd.

```
228                                {
229     if(newSpeed == _scrollSpeed)
230         return;
231     _scrollSpeed = newSpeed;
232 #ifdef Q_OS_MACX
233     // Update all devices
234     foreach(Kb* kb, activeDevices){
235         kb->cmd.write(QString("scrollspeed %1\n").arg(newSpeed).toLatin1());
```

```
236         kb->cmd.flush();
237     }
238 #endif
239 }
```

12.21.3.56 void Kb::setCurrentMode (KbProfile * *profile*, KbMode * *mode*, bool *spontaneous* = true)

Definition at line 769 of file kb.cpp.

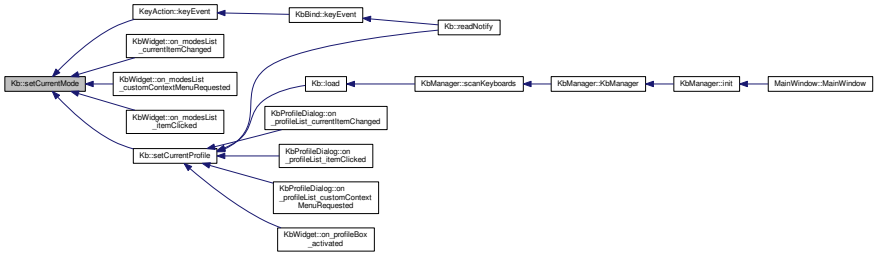
References _currentMode, _currentProfile, _needsSave, KbProfile::currentMode(), modeChanged(), and profileChanged().

Referenced by KeyAction::keyEvent(), KbWidget::on_modesList_currentItemChanged(), KbWidget::on_modesList_customContextMenuRequested(), KbWidget::on_modesList_itemClicked(), and setCurrentProfile().

```
769                                                                 {
770     if(_currentProfile != profile){
771         _currentProfile = profile;
772         _needsSave = true;
773         emit profileChanged();
774     }
775     if(_currentMode != mode || _currentProfile->
    currentMode() != mode){
776         _currentProfile->currentMode(_currentMode = mode);
777         _needsSave = true;
778         emit modeChanged(spontaneous);
779     }
780 }
```

Here is the call graph for this function:

Here is the caller graph for this function:



12.21.3.57 void Kb::setCurrentMode (KbProfile * *profile*, int *index*, bool *spontaneous* = true) [inline]

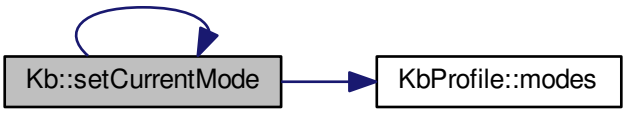
Definition at line 73 of file kb.h.

References KbProfile::modes(), and setCurrentMode().

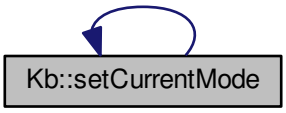
Referenced by setCurrentMode().

```
73 { setCurrentMode(profile, profile->modes()[index], spontaneous); }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.21.3.58 void Kb::setCurrentMode (KbMode * *mode*, bool *spontaneous* = true) [inline]

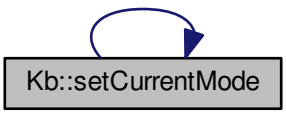
Definition at line 74 of file kb.h.

References _currentProfile, and setCurrentMode().

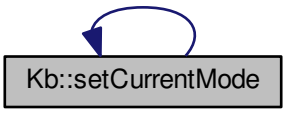
Referenced by setCurrentMode().

```
74 { setCurrentMode(_currentProfile, mode, spontaneous); }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.21.3.59 void Kb::setCurrentProfile (KbProfile * *profile*, bool *spontaneous* = true)

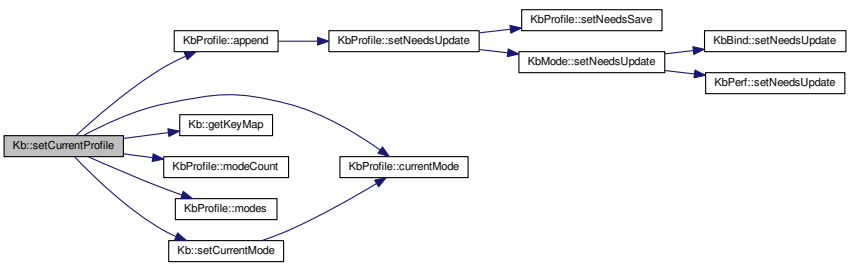
Definition at line 760 of file kb.cpp.

References KbProfile::append(), KbProfile::currentMode(), getKeyMap(), hwModeCount, KbProfile::modeCount(), KbProfile::modes(), and setCurrentMode().

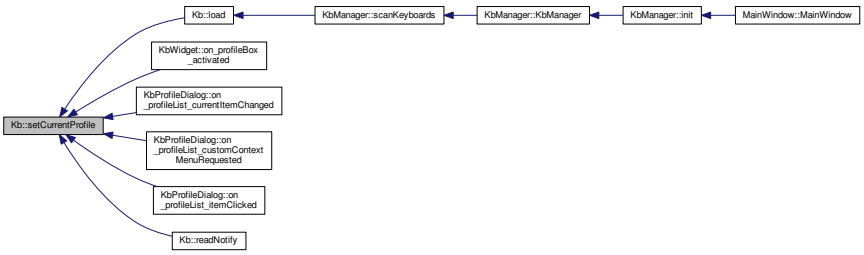
Referenced by load(), KbWidget::on_profileBox_activated(), KbProfileDialog::on_profileList_currentItemChanged(), KbProfileDialog::on_profileList_customContextMenuRequested(), KbProfileDialog::on_profileList_itemClicked(), and readNotify().

```
760                                                                 {
761     while(profile->modeCount() < hwModeCount)
762         profile->append(new KbMode(this, getKeyMap()));
763     KbMode* mode = profile->currentMode();
764     if(!mode)
765         profile->currentMode(mode = profile->modes().first());
766     setCurrentMode(profile, mode, spontaneous);
767 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.21.3.60 void Kb::updateLayout ( ) [private]

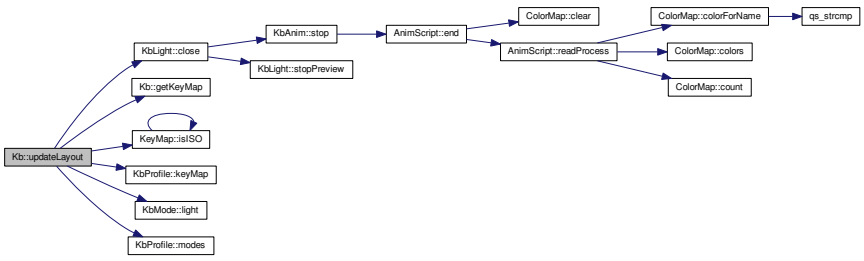
Definition at line 186 of file kb.cpp.

References _currentProfile, _hwProfile, _layout, _profiles, KbLight::close(), cmd, getKeyMap(), infoUpdated(), KeyMap::isISO(), KbProfile::keyMap(), KbMode::light(), and KbProfile::modes().

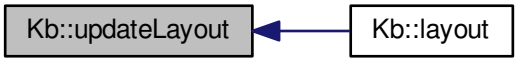
Referenced by layout().

```
186                     {
187 #ifdef Q_OS_MACX
188     // Write ANSI/ISO flag to daemon (OSX only)
189     cmd.write("layout ");
190     cmd.write(KeyMap::isISO(_layout) ? "iso" : "ansi");
191     cmd.write("\n");
192     cmd.flush();
193 #endif
194     foreach(KbProfile* profile, _profiles)
195         profile->keyMap(getKeyMap());
196     if(_hwProfile && !_profiles.contains(_hwProfile))
197         _hwProfile->keyMap(getKeyMap());
198     // Stop all animations as they'll need to be restarted
199     foreach(KbMode* mode, _currentProfile->modes())
200         mode->light()->close();
201     emit infoUpdated();
202 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.21.3.61 void Kb::writeProfileHeader ( ) [private]

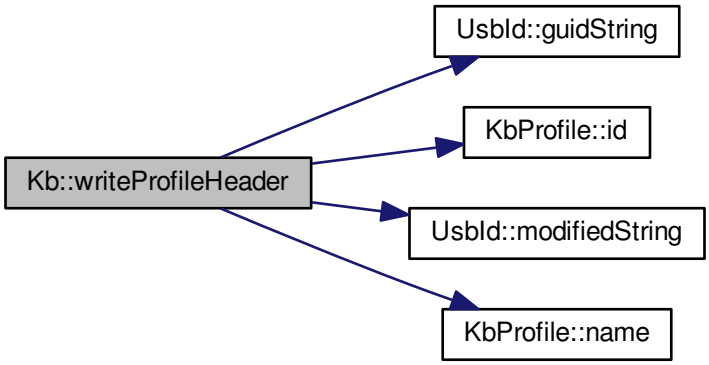
Definition at line 354 of file kb.cpp.

References _currentProfile, cmd, UsbId::guidString(), KbProfile::id(), UsbId::modifiedString(), and KbProfile::name().

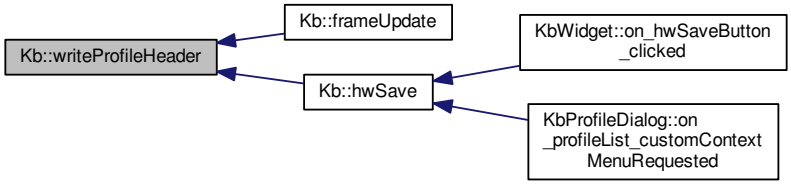
Referenced by frameUpdate(), and hwSave().

```
354                             {
355     cmd.write("eraseprofile");
356     // Write the profile name and ID
357     cmd.write(" profilename ");
358     cmd.write(QUrl::toPercentEncoding(_currentProfile->name()));
359     cmd.write(" profileid ");
360     cmd.write(_currentProfile->id().guidString().toLatin1());
361     cmd.write(" ");
362     cmd.write(_currentProfile->id().modifiedString().toLatin1());
363 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.21.4 Friends And Related Function Documentation

12.21.4.1 friend class KbManager [friend]

Definition at line 142 of file kb.h.

12.21.5 Field Documentation

12.21.5.1 KbMode* Kb::_currentMode [private]

Definition at line 169 of file kb.h.

Referenced by currentBind(), currentLight(), currentMode(), currentPerf(), frameUpdate(), hwSave(), readNotify(), and setCurrentMode().

12.21.5.2 KbProfile* Kb::_currentProfile [private]

Definition at line 167 of file kb.h.

Referenced by currentProfile(), frameUpdate(), hwSave(), readNotify(), save(), setCurrentMode(), updateLayout(), and writeProfileHeader().

12.21.5.3 bool Kb::_delay = false [static], [private]

Definition at line 198 of file kb.h.

Referenced by Kb(), and macroDelay().

12.21.5.4 bool Kb::_dither = false [static], [private]

Definition at line 165 of file kb.h.

Referenced by dither(), and Kb().

12.21.5.5 int Kb::_frameRate = 30 [static], [private]

Definition at line 164 of file kb.h.

Referenced by frameRate(), and Kb().

12.21.5.6 KbProfile* Kb::_hwProfile [private]

Definition at line 183 of file kb.h.

Referenced by deleteHw(), hwProfile(), hwSave(), readNotify(), and updateLayout().

12.21.5.7 KeyMap::Layout Kb::_layout = KeyMap::NO_LAYOUT [static], [private]

Definition at line 161 of file kb.h.

Referenced by getKeyMap(), Kb(), layout(), and updateLayout().

12.21.5.8 KeyMap::Model Kb::_model [private]

Definition at line 171 of file kb.h.

Referenced by getKeyMap(), isKeyboard(), isMouse(), Kb(), and model().

12.21.5.9 bool Kb::_mouseAccel = true [static], [private]

Definition at line 165 of file kb.h.

Referenced by Kb(), and mouseAccel().

12.21.5.10 bool Kb::_needsSave [private]

Definition at line 201 of file kb.h.

Referenced by appendProfile(), load(), needsSave(), profiles(), readNotify(), save(), and setCurrentMode().

12.21.5.11 QList<KbProfile*> Kb::_profiles [private]

Definition at line 168 of file kb.h.

Referenced by appendProfile(), find(), indexOf(), load(), needsSave(), profiles(), readNotify(), save(), and updateLayout().

12.21.5.12 int Kb::_scrollSpeed = 0 [static], [private]

Definition at line 164 of file kb.h.

Referenced by Kb(), and scrollSpeed().

12.21.5.13 QFile Kb::cmd [private]

Definition at line 191 of file kb.h.

Referenced by dither(), frameRate(), frameUpdate(), fwUpdate(), hwSave(), isOpen(), Kb(), macroDelay(), mouseAccel(), readNotify(), scrollSpeed(), updateLayout(), writeProfileHeader(), and ~Kb().

12.21.5.14 QString Kb::cmdpath [private]

Definition at line 155 of file kb.h.

Referenced by Kb().

12.21.5.15 QString Kb::devpath [private]

Definition at line 155 of file kb.h.

Referenced by matches().

12.21.5.16 QString Kb::features

Definition at line 18 of file kb.h.

Referenced by MainWindow::checkFwUpdates(), FwUpgradeDialog::exec(), Kb(), KbWidget::KbWidget(), KbWidget::on_fwUpdButton_clicked(), and KbWidget::updateFwButton().

12.21.5.17 QString Kb::firmware

Definition at line 18 of file kb.h.

Referenced by MainWindow::checkFwUpdates(), KbWidget::devUpdate(), FwUpgradeDialog::FwUpgradeDialog(), Kb(), KbWidget::on_fwUpdButton_clicked(), and KbWidget::updateFwButton().

12.21.5.18 QString Kb::fwUpdPath [private]

Definition at line 181 of file kb.h.

Referenced by fwUpdate(), and readNotify().

12.21.5.19 bool Kb::hwLoading[HWMODE_MAX+1] [private]

Definition at line 205 of file kb.h.

Referenced by Kb(), and readNotify().

12.21.5.20 const int Kb::HWMODE_MAX = 3 [static]

Definition at line 50 of file kb.h.

Referenced by KbPerf::applyIndicators(), and readNotify().

12.21.5.21 int Kb::hwModeCount

Definition at line 49 of file kb.h.

Referenced by hwSave(), Kb(), KbWidget::modeIcon(), KbWidget::on_modesList_customContextMenuRequested(), readNotify(), and setCurrentProfile().

12.21.5.22 bool Kb::iState[KbPerf::HW_I_COUNT] [private]

Definition at line 174 of file kb.h.

Referenced by frameUpdate(), Kb(), and readNotify().

12.21.5.23 quint64 Kb::lastAutoSave [private]

Definition at line 178 of file kb.h.

Referenced by autoSave().

12.21.5.24 int Kb::macroNumber [private]

Definition at line 196 of file kb.h.

Referenced by getMacroNumber(), and Kb().

12.21.5.25 QString Kb::macroPath [private]

Definition at line 155 of file kb.h.

Referenced by getMacroPath(), Kb(), and ~Kb().

12.21.5.26 bool Kb::monochrome

Definition at line 19 of file kb.h.

Referenced by frameUpdate(), hwSave(), Kb(), and KbWidget::KbWidget().

12.21.5.27 int Kb::notifyNumber [private]

Definition at line 194 of file kb.h.

Referenced by frameUpdate(), fwUpdate(), hwSave(), Kb(), readNotify(), and ~Kb().

12.21.5.28 QString Kb::notifyPath [private]

Definition at line 155 of file kb.h.

Referenced by Kb(), and run().

12.21.5.29 QString Kb::pollrate

Definition at line 18 of file kb.h.

Referenced by KbWidget::devUpdate(), and Kb().

12.21.5.30 QString Kb::prefsPath [private]

Definition at line 177 of file kb.h.

Referenced by Kb(), load(), and save().

12.21.5.31 KbMode* Kb::prevMode [private]

Definition at line 186 of file kb.h.

Referenced by deletePrevious(), frameUpdate(), and hwSave().

12.21.5.32 KbProfile* Kb::prevProfile [private]

Definition at line 185 of file kb.h.

Referenced by frameUpdate().

12.21.5.33 QString Kb::usbModel

Definition at line 16 of file kb.h.

Referenced by FwUpgradeDialog::FwUpgradeDialog(), Kb(), KbWidget::name(), and MainWindow::showFwUpdateNotification().

12.21.5.34 QString Kb::usbSerial

Definition at line 16 of file kb.h.

Referenced by KbWidget::devUpdate(), Kb(), and matches().

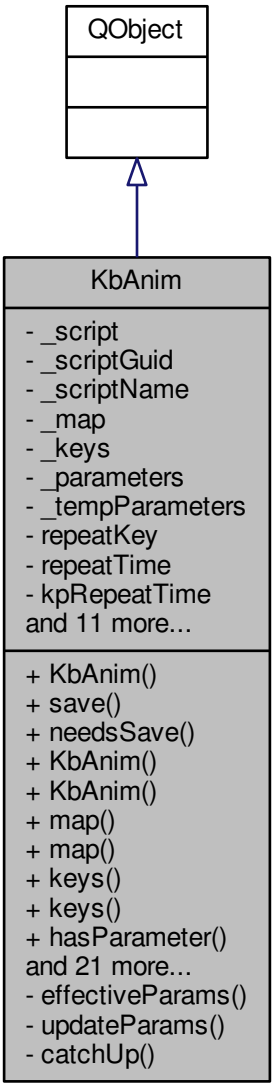
The documentation for this class was generated from the following files:

- src/ckb/kb.h
- src/ckb/kb.cpp
- src/ckb/release/moc_kb.cpp

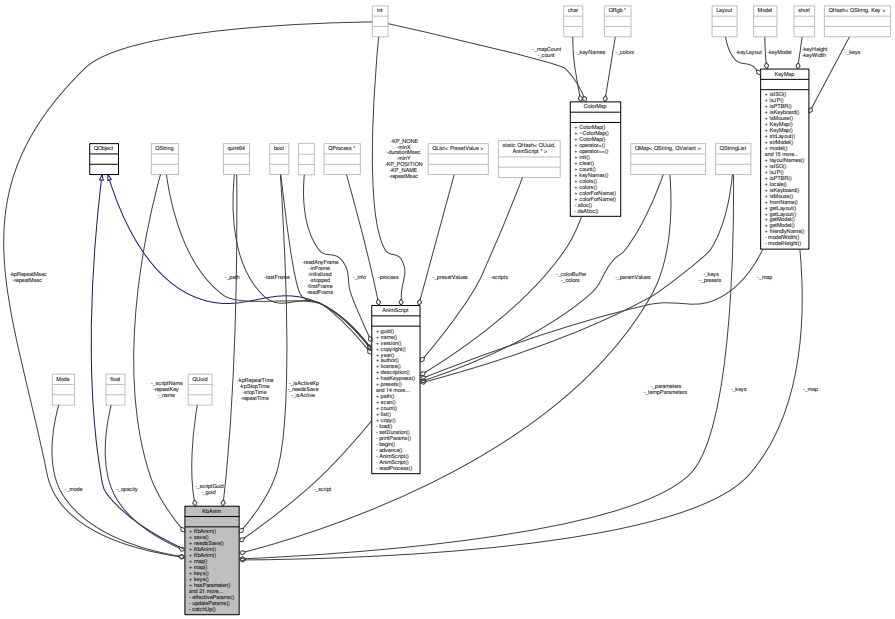
12.22 KbAnim Class Reference

`#include <src/ckb/kbanim.h>`

Inheritance diagram for KbAnim:



Collaboration diagram for KbAnim:



Public Types

- enum Mode {
  Normal, Add, Subtract, Multiply,
  Divide }

Public Member Functions

- KbAnim (QObject *parent, const KeyMap &map, const QUuid id, CkbSettings &settings)
- void save (CkbSettings &settings)
- bool needsSave () const
- KbAnim (QObject *parent, const KeyMap &map, const QString &name, const QStringList &keys, const AnimScript *script)
- KbAnim (QObject *parent, const KeyMap &map, const KbAnim &other)
- const KeyMap & map ()
- void map (const KeyMap &newMap)
- const QStringList & keys ()
- void keys (const QStringList &newKeys)
- bool hasParameter (const QString &name) const
- QVariant parameter (const QString &name) const
- void parameter (const QString &name, const QVariant &value)
- void commitParams ()
- void resetParams ()
- void reInit ()
- void trigger (quint64 timestamp, bool ignoreParameter=false)
- void keypress (const QString &key, bool pressed, quint64 timestamp)
- void stop ()
- bool isActive () const
- bool isRunning () const
- void blend (ColorMap &animMap, quint64 timestamp)
- const QUuid & guid () const

- void newId ()
- const QString & name () const
- void name (const QString &newName)
- float opacity () const
- void opacity (float newOpacity)
- Mode mode () const
- void mode (Mode newMode)
- const AnimScript * script () const
- const QString & scriptName () const

Private Member Functions

- QMap< QString, QVariant > effectiveParams ()
- void updateParams ()
- void catchUp (quint64 timestamp)

Private Attributes

- AnimScript * _script
- QUuid _scriptGuid
- QString _scriptName
- KeyMap _map
- QStringList _keys
- QMap< QString, QVariant > _parameters
- QMap< QString, QVariant > _tempParameters
- QString repeatKey
- quint64 repeatTime
- quint64 kpRepeatTime
- quint64 stopTime
- quint64 kpStopTime
- int repeatMsec
- int kpRepeatMsec
- QUuid _guid
- QString _name
- float _opacity
- Mode _mode
- bool _isActive
- bool _isActiveKp
- bool _needsSave

12.22.1 Detailed Description

Definition at line 11 of file kbanim.h.

12.22.2 Member Enumeration Documentation

12.22.2.1 enum KbAnim::Mode

Enumerator

Normal

Add

Subtract

Multiply

Divide

Definition at line 17 of file kbanim.h.

```
17              {
18         Normal,
19         Add,
20         Subtract,
21         Multiply,
22         Divide
23     };
```

12.22.3 Constructor & Destructor Documentation

12.22.3.1 KbAnim::KbAnim (QObject * *parent,* const KeyMap & *map,* const QUuid *id,* CkbSettings & *settings*)

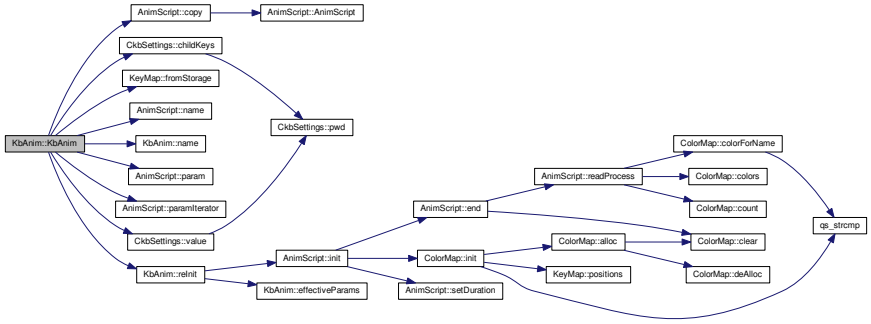
Definition at line 8 of file kbanim.cpp.

References _guid, _keys, _map, _mode, _name, _opacity, _parameters, _script, _scriptGuid, _scriptName, CkbSettings::childKeys(), AnimScript::copy(), AnimScript::Param::defaultValue, KeyMap::fromStorage(), AnimScript::Param::INVALID, AnimScript::Param::LABEL, AnimScript::Param::name, AnimScript::name(), name(), Normal, AnimScript::param(), AnimScript::paramIterator(), reInit(), AnimScript::Param::type, and CkbSettings::value().

```
8                                                                                                   :
9      QObject(parent), _script(0), _map(map),
10      repeatTime(0), kpRepeatTime(0), stopTime(0),
       kpStopTime(0), repeatMsec(0), kpRepeatMsec(0),
11      _guid(id), _isActive(false), _isActiveKp(false),
       _needsSave(false)
12 {
13     SGroup group(settings, _guid.toString().toUpper());
14     _keys = settings.value("Keys").toStringList();
15     // Convert key list from storage names if needed
16     if(!settings.value("UseRealNames").toBool()){
17         QMutableListIterator<QString> i(_keys);
18         while(i.hasNext()){
19             i.next();
20             QString& key = i.value();
21             key = _map.fromStorage(key);
22         }
23     }
24     _name = settings.value("Name").toString().trimmed();
25     _opacity = settings.value("Opacity").toString().toDouble();
26     if(_opacity < 0.)
27         _opacity = 0.;
28     else if(_opacity > 1.)
29         _opacity = 1.;
30     bool modeOk = false;
31     _mode = (Mode)metaObject()->enumerator(metaObject()->indexOfEnumerator("Mode")).keysToValue(
       settings.value("BlendMode").toString().toLatin1(), &modeOk);
32     if(!modeOk)
33         _mode = Normal;
34     _scriptName = settings.value("ScriptName").toString().trimmed();
35     _scriptGuid = settings.value("ScriptGuid").toString();
36     {
37         SGroup group(settings, "Parameters");
38         foreach(const QString& param, settings.childKeys())
39             _parameters[param.toLower()] = settings.value(param);
40     }
41
42     if(!_scriptGuid.isNull()){
43         _script = AnimScript::copy(this, _scriptGuid);
44         if(_script){
45             // Remove nonexistant parameters
46             foreach(const QString& name, _parameters.keys()){
47                 AnimScript::Param param = _script->param(name);
48                 if(param.type == AnimScript::Param::INVALID || param.
       type == AnimScript::Param::LABEL)
49                     _parameters.remove(name);
50             }
51             // Add defaults for unset parameters
52             QListIterator<AnimScript::Param> i = _script->paramIterator();
53             while(i.hasNext()){
54                 AnimScript::Param param = i.next();
55                 if(!_parameters.contains(param.name) && param.
       type != AnimScript::Param::LABEL)
```

```
56                      _parameters[param.name] = param.defaultValue;
57              }
58              _scriptName = _script->name();
59              reInit();
60          }
61      }
62  }
```

Here is the call graph for this function:



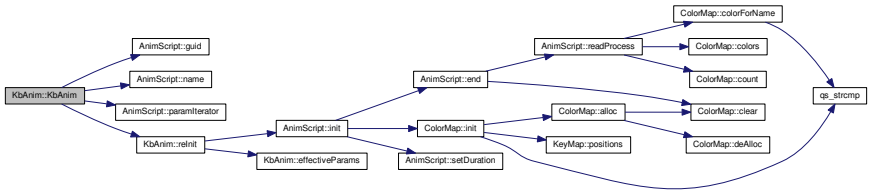
12.22.3.2 KbAnim::KbAnim (QObject * *parent*, const KeyMap & *map*, const QString & *name*, const QStringList & *keys*, const AnimScript * *script*)

Definition at line 84 of file kbanim.cpp.

References _parameters, _script, _scriptGuid, _scriptName, AnimScript::Param::defaultValue, AnimScript::guid(), AnimScript::Param::LABEL, AnimScript::Param::name, AnimScript::name(), AnimScript::paramIterator(), reInit(), and AnimScript::Param::type.

```
84
                    :
85      QObject(parent),
86      _script(AnimScript::copy(this, script->guid())),
        _map(map), _keys(keys),
87      repeatTime(0), kpRepeatTime(0), stopTime(0),
        kpStopTime(0), repeatMsec(0), kpRepeatMsec(0),
88      _guid(QUuid::createUuid()), _name(name), _opacity(1.),
        _mode(Normal), _isActive(false), _isActiveKp(false),
        _needsSave(true)
89  {
90      if(_script){
91          // Set default parameters
92          QListIterator<AnimScript::Param> i = _script->paramIterator();
93          while(i.hasNext()){
94              AnimScript::Param param = i.next();
95              if(param.type != AnimScript::Param::LABEL)
96                  _parameters[param.name] = param.defaultValue;
97          }
98          _scriptGuid = script->guid();
99          _scriptName = script->name();
100         reInit();
101     }
102 }
```

Here is the call graph for this function:



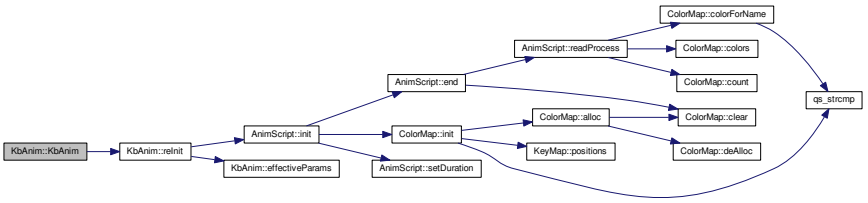
12.22.3.3 KbAnim::KbAnim (QObject * *parent,* const KeyMap & *map,* const KbAnim & *other*)

Definition at line 104 of file kbanim.cpp.

References reInit().

```
104                                                                    :
105     QObject(parent),
106     _script(AnimScript::copy(this, other.script()->
      guid())), _scriptGuid(_script->guid()), _scriptName(
      _script->name()),
107     _map(map), _keys(other._keys), _parameters(other.
      _parameters),
108     repeatTime(0), kpRepeatTime(0), stopTime(0),
      kpStopTime(0), repeatMsec(0), kpRepeatMsec(0),
109     _guid(other._guid), _name(other._name), _opacity(other.
      _opacity), _mode(other._mode), _isActive(false),
      _isActiveKp(false), _needsSave(true)
110 {
111     reInit();
112 }
```

Here is the call graph for this function:



12.22.4 Member Function Documentation

12.22.4.1 void KbAnim::blend (ColorMap & *animMap,* quint64 *timestamp*)

Definition at line 365 of file kbanim.cpp.

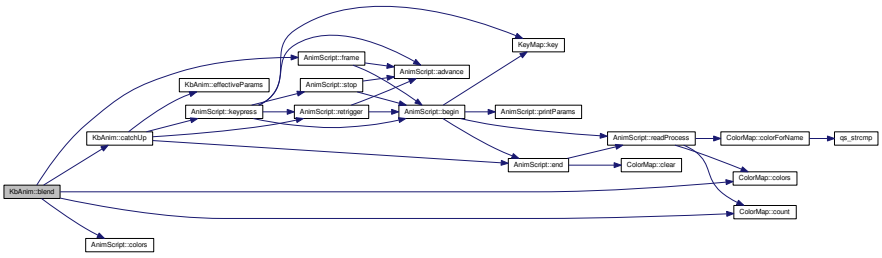
References _mode, _opacity, _script, catchUp(), ColorMap::colors(), AnimScript::colors(), ColorMap::count(), AnimScript::frame(), and functions.

Referenced by KbLight::frameUpdate().

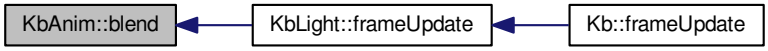
```
365                                                    {
366     if(!_script)
367         return;
368
369     // Fetch the next frame from the script
370     catchUp(timestamp);
```

```
371     _script->frame(timestamp);
372
373     // Blend the script's map with the current map
374     int blendMode = (int)_mode;
375     blendFunc blend = functions[blendMode];
376     float fOpacity = _opacity / 255.f;  // save some math by pre-dividing the 255 for qAlpha
377
378     const ColorMap& scriptMap = _script->colors();
379     int count = animMap.count();
380     if(scriptMap.count() != count){
381         qDebug() << "Script map didn't match base map (" << count << " vs " << scriptMap.
      count() << "). This should never happen.";
382         return;
383     }
384     QRgb* background = animMap.colors();
385     const QRgb* foreground = scriptMap.colors();
386     for(int i = 0; i < count; i++){
387         // Mix the colors in with the color map according to blend mode and alpha
388         QRgb& bg = background[i];
389         QRgb fg = foreground[i];
390         int alpha = qAlpha(fg);
391         if(alpha == 0)
392             continue;
393         if(blendMode == 0){
394             // Blend: normal
395             // This is the most common use case and it requires much less arithmetic
396             if(alpha == 255){
397                 bg = fg;
398             } else {
399                 float r = qRed(bg), g = qGreen(bg), b = qBlue(bg);
400                 float a = alpha * fOpacity;
401                 r = r * (1.f - a) + qRed(fg) * a;
402                 g = g * (1.f - a) + qGreen(fg) * a;
403                 b = b * (1.f - a) + qBlue(fg) * a;
404                 bg = qRgb(round(r), round(g), round(b));
405             }
406         } else {
407             // Use blend function
408             float r = qRed(bg) / 255.f, g = qGreen(bg) / 255.f, b = qBlue(bg) / 255.f;
409             float a = alpha * fOpacity;
410             r = r * (1.f - a) + blend(r, qRed(fg) / 255.f) * a;
411             g = g * (1.f - a) + blend(g, qGreen(fg) / 255.f) * a;
412             b = b * (1.f - a) + blend(b, qBlue(fg) / 255.f) * a;
413             bg = qRgb(round(r * 255.f), round(g * 255.f), round(b * 255.f));
414         }
415     }
416 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.22.4.2 void KbAnim::catchUp (quint64 *timestamp*) [private]

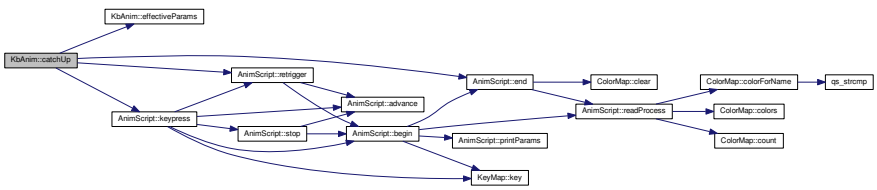
Definition at line 166 of file kbanim.cpp.

References _isActive, _isActiveKp, _script, effectiveParams(), AnimScript::end(), AnimScript::keypress(), kpRepeatMsec, kpRepeatTime, kpStopTime, repeatKey, repeatMsec, repeatTime, AnimScript::retrigger(), and stopTime.

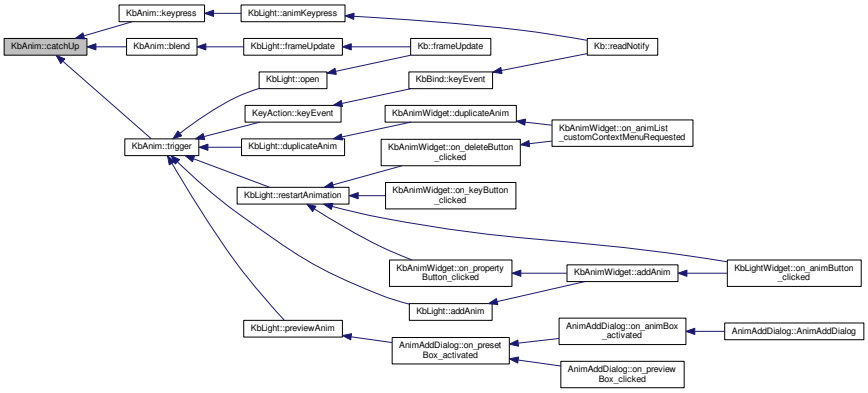
Referenced by blend(), keypress(), and trigger().

```
166                                     {
167     QMap<QString, QVariant> parameters = effectiveParams();
168     // Stop the animation if its time has run out
169     if(stopTime != 0 && timestamp >= stopTime){
170         repeatMsec = repeatTime = 0;
171         if(!parameters.contains("repeat")){
172             // If repeats aren't allowed, stop the animation entirely
173             _script->end();
174             _isActive = false;
175             return;
176         } else
177             // Otherwise, simply stop repeating
178             stopTime = 0;
179     }
180     if(kpStopTime != 0 && timestamp >= kpStopTime){
181         kpRepeatMsec = kpRepeatTime = 0;
182         if(!parameters.contains("kprepeat")){
183             _script->end();
184             _isActiveKp = false;
185             return;
186         } else
187             kpStopTime = 0;
188     }
189
190     // Restart (or start, if there was a delay) the animation if its repeat time is up
191     while(repeatTime > 0 && timestamp >= repeatTime){
192         _script->retrigger(repeatTime);
193         if(repeatMsec <= 0){
194             repeatTime = 0;
195             break;
196         }
197         repeatTime += repeatMsec;
198     }
199     while(kpRepeatTime > 0 && timestamp >= kpRepeatTime){
200         _script->keypress(repeatKey, 1, kpRepeatTime);
201         if(kpRepeatMsec <= 0){
202             kpRepeatTime = 0;
203             break;
204         }
205         kpRepeatTime += kpRepeatMsec;
206     }
207 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.22.4.3 void KbAnim::commitParams ( )

Definition at line 121 of file kbanim.cpp.

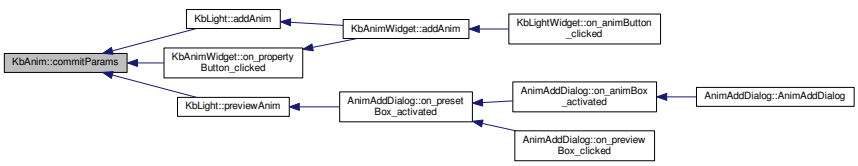
References _needsSave, _parameters, _tempParameters, and effectiveParams().

Referenced by KbLight::addAnim(), KbAnimWidget::on_propertyButton_clicked(), and KbLight::previewAnim().

```
121                          {
122     _needsSave = true;
123     _parameters = effectiveParams();
124     _tempParameters.clear();
125 }
```

Here is the call graph for this function:

Here is the caller graph for this function:



12.22.4.4 QMap< QString, QVariant > KbAnim::effectiveParams ( ) [private]

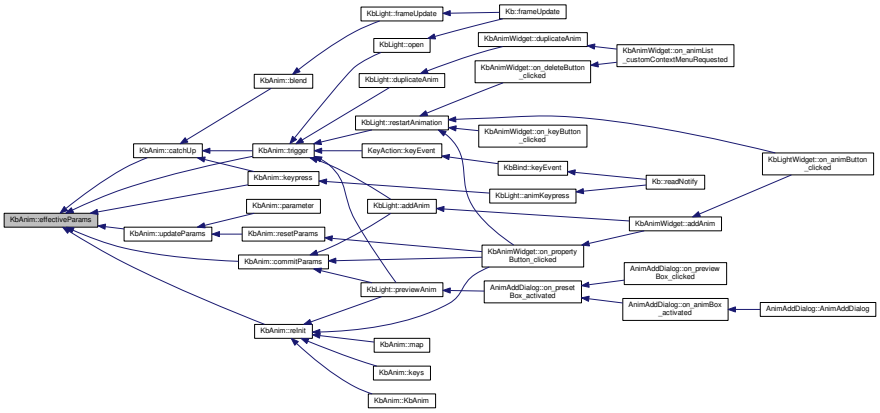
Definition at line 138 of file kbanim.cpp.

References _parameters, and _tempParameters.

Referenced by catchUp(), commitParams(), keypress(), reInit(), trigger(), and updateParams().

```
138                                                 {
139     QMap<QString, QVariant> res = _parameters;
140     // Apply all uncommited parameters
141     QMapIterator<QString, QVariant> i(_tempParameters);
142     while(i.hasNext()){
143         i.next();
144         res[i.key()] = i .value();
145     }
146     return res;
147 }
```

Here is the caller graph for this function:



12.22.4.5 const QUuid& KbAnim::guid ( ) const [inline]

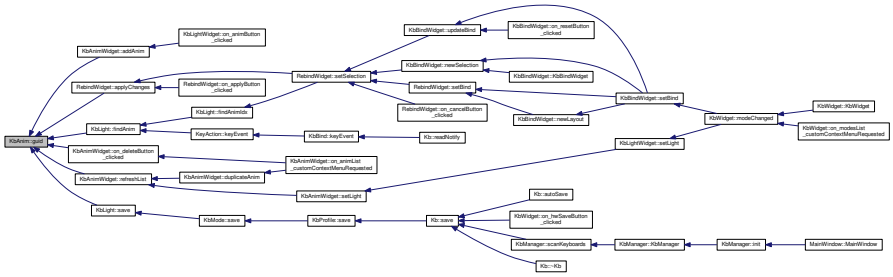
Definition at line 71 of file kbanim.h.

References _guid.

Referenced by KbAnimWidget::addAnim(), RebindWidget::applyChanges(), KbLight::findAnim(), KbAnimWidget::on_deleteButton_clicked(), KbAnimWidget::refreshList(), and KbLight::save().

```
71 { return _guid; }
```

Here is the caller graph for this function:



12.22.4.6 bool KbAnim::hasParameter (const QString & *name*) const [inline]

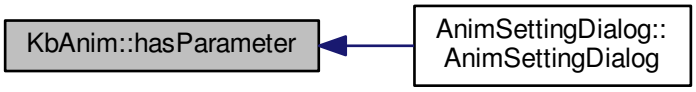
Definition at line 44 of file kbanim.h.

References _parameters.

Referenced by AnimSettingDialog::AnimSettingDialog().

```
44 { return _parameters.contains(name); }
```

Here is the caller graph for this function:



12.22.4.7 bool KbAnim::isActive ( ) const [inline]

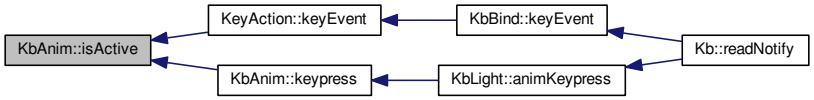
Definition at line 63 of file kbanim.h.

References _isActive, and _isActiveKp.

Referenced by KeyAction::keyEvent(), and keypress().

```
63 { return _isActive || _isActiveKp; }
```

Here is the caller graph for this function:



12.22.4.8 bool KbAnim::isRunning ( ) const

Definition at line 325 of file kbanim.cpp.

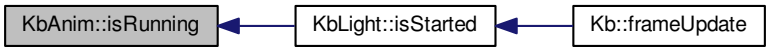
References _script, and AnimScript::hasFrame().

Referenced by KbLight::isStarted().

```
325                             {
326     if(!_script)
327         return true;    // If the script wasn't loaded, pretend it's running anyway so it won't lock up
328     return _script->hasFrame();
329 }
```

Here is the call graph for this function:

Here is the caller graph for this function:



12.22.4.9 void KbAnim::keypress (const QString & *key,* bool *pressed,* quint64 *timestamp*)

Definition at line 250 of file kbanim.cpp.

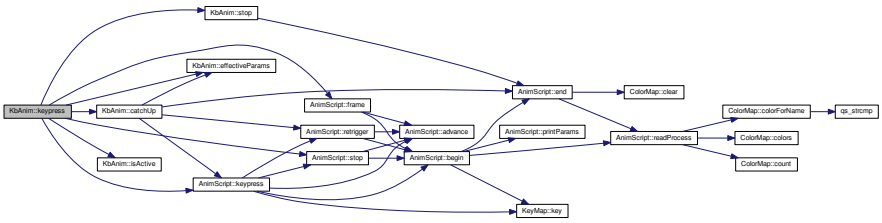
References _isActive, _isActiveKp, _script, catchUp(), effectiveParams(), AnimScript::frame(), isActive(), AnimScript::keypress(), kpRepeatMsec, kpRepeatTime, kpStopTime, repeatKey, repeatMsec, repeatTime, stop(), AnimScript::stop(), and stopTime.

Referenced by KbLight::animKeypress().

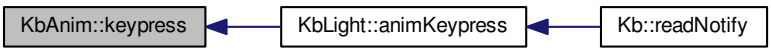
```
250                                                                     {
251     if(!_script)
252         return;
253     QMap<QString, QVariant> parameters = effectiveParams();
254     if(pressed && parameters.value("kpmodestop").toBool()){
255         // If stop on key press is enabled, stop mode-wide animation
256         catchUp(timestamp);
257         _script->stop(timestamp);
258         stopTime = repeatTime = repeatMsec = 0;
259         _isActive = false;
260     } else {
261         if(!parameters.value("kptrigger").toBool())
262             return;
263         catchUp(timestamp);
264     }
265
266     if(pressed){
267         // Key pressed
268         _isActiveKp = true;
269         if(parameters.value("kpmode", 0).toInt() == 2 && isActive())
270             // If mode is start once and a key has already been pressed, do nothing
271             return;
272         int delay = round(parameters.value("kpdelay").toDouble() * 1000.);
273         if(delay > 0){
274             // If delay is enabled, wait to trigger the event
275             timestamp += delay;
276             kpRepeatTime = timestamp;
277         } else
278             _script->keypress(key, pressed, timestamp);
279
280         int repeat = round(parameters.value("kprepeat").toDouble() * 1000.);
281         if(repeat <= 0){
282             // If no repeat allowed, calculate stop time in seconds
283             kpRepeatMsec = -1;
284             double stop = parameters.value("kpstop").toDouble();
285             if(stop <= 0.)
```

```
286                 kpStopTime = 0;
287             else
288                 kpStopTime = timestamp + round(stop * 1000.);
289         } else {
290             // If repeat is allowed, calculate stop time in repetitions
291             kpRepeatMsec = repeat;
292             if(delay <= 0)
293                 kpRepeatTime = timestamp + repeat;
294             int stop = parameters.value("kpstop").toInt();
295             if(stop < 0)
296                 kpStopTime = 0;
297             else
298                 kpStopTime = timestamp + repeat * (1 + stop);
299         }
300         repeatKey = key;
301     } else {
302         // Key released
303         _isActiveKp = false;
304         _script->keypress(key, pressed, timestamp);
305         if(parameters.value("kprelease").toBool())
306             // Stop repeating keypress if "Stop on key release" is enabled
307             kpStopTime = timestamp;
308     }
309     _script->frame(timestamp);
310 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.22.4.10 const QStringList& KbAnim::keys ( ) [inline]

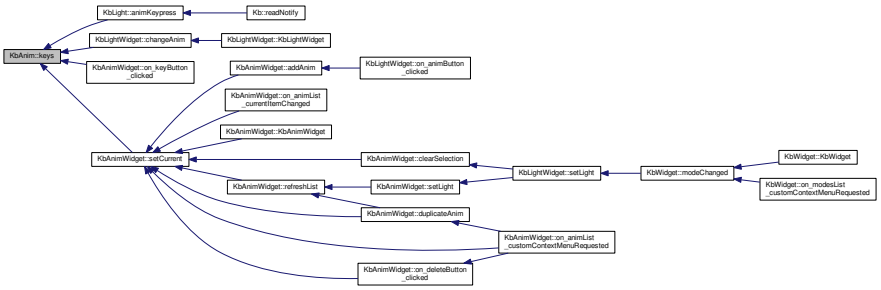
Definition at line 40 of file kbanim.h.

References _keys.

Referenced by KbLight::animKeypress(), KbLightWidget::changeAnim(), KbAnimWidget::on_keyButton_clicked(), and KbAnimWidget::setCurrent().

```
40 { return _keys; }
```

Here is the caller graph for this function:



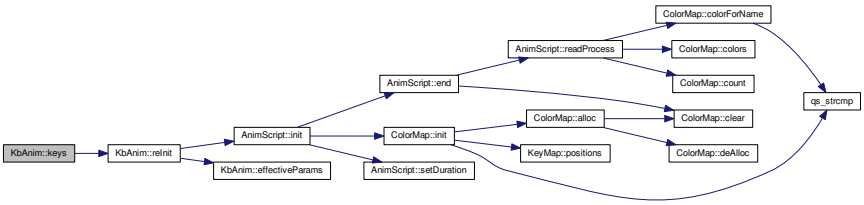
12.22.4.11 void KbAnim::keys (const QStringList & *newKeys*)

Definition at line 161 of file kbanim.cpp.

References _keys, and reInit().

```
161                                             {
162     _keys = newKeys;
163     reInit();
164 }
```

Here is the call graph for this function:



12.22.4.12 const KeyMap& KbAnim::map ( ) [inline]

Definition at line 37 of file kbanim.h.

References _map.

Referenced by KbLight::map().

```
37 { return _map; }
```

Here is the caller graph for this function:

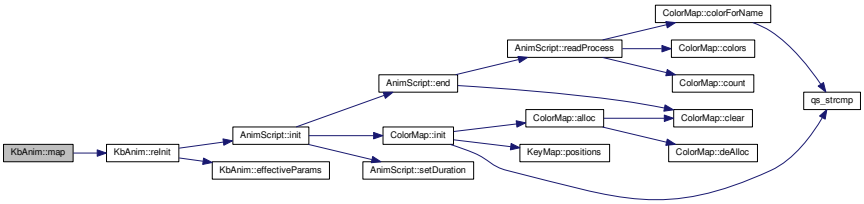
12.22.4.13 void KbAnim::map (const KeyMap & *newMap*)

Definition at line 156 of file kbanim.cpp.

References _map, and reInit().

```
156                                    {
157     _map = newMap;
158     reInit();
159 }
```

Here is the call graph for this function:



12.22.4.14 Mode KbAnim::mode ( ) const [inline]

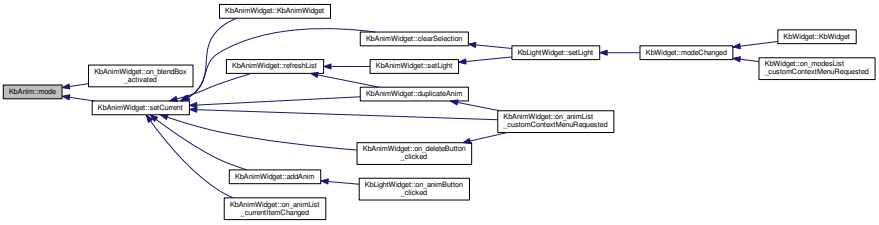
Definition at line 77 of file kbanim.h.

References _mode.

Referenced by KbAnimWidget::on_blendBox_activated(), and KbAnimWidget::setCurrent().

```
77 { return _mode; }
```

Here is the caller graph for this function:



12.22.4.15 void KbAnim::mode (Mode *newMode*) [inline]

Definition at line 78 of file kbanim.h.

References _mode, and _needsSave.

```
78 { _needsSave = true; _mode = newMode; }
```

12.22.4.16 const QString& KbAnim::name ( ) const [inline]

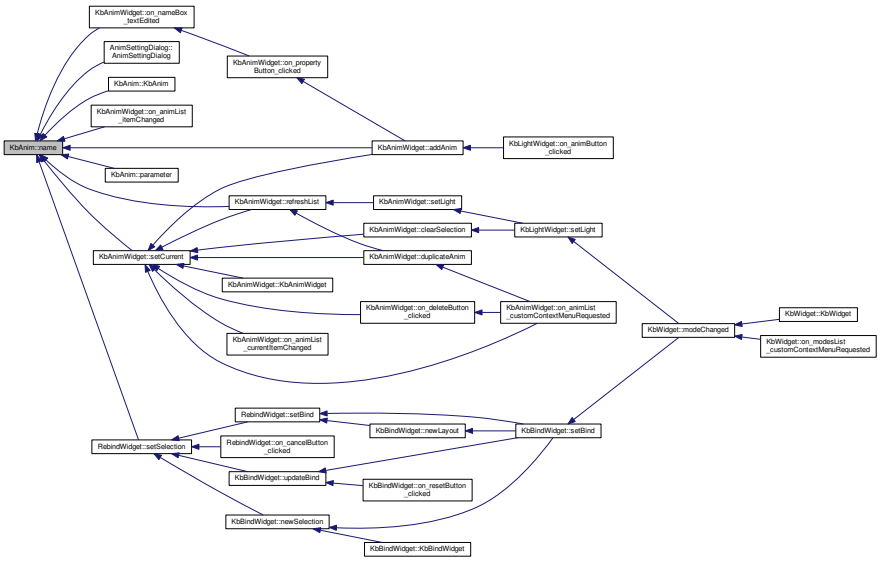
Definition at line 73 of file kbanim.h.

References _name.

Referenced by KbAnimWidget::addAnim(), AnimSettingDialog::AnimSettingDialog(), KbAnim(), KbAnimWidget::on_animList_itemChanged(), KbAnimWidget::on_nameBox_textEdited(), parameter(), KbAnimWidget::refreshList(), KbAnimWidget::setCurrent(), and RebindWidget::setSelection().

```
73 { return _name; }
```

Here is the caller graph for this function:



12.22.4.17 void KbAnim::name (const QString & *newName*) [inline]

Definition at line 74 of file kbanim.h.

References _name, and _needsSave.

```
74 { _needsSave = true; _name = newName; }
```

12.22.4.18 bool KbAnim::needsSave ( ) const [inline]

Definition at line 29 of file kbanim.h.

References _needsSave.

Referenced by KbLight::needsSave().

```
29 { return _needsSave; }
```

Here is the caller graph for this function:

12.22.4.19 void KbAnim::newId ( ) [inline]

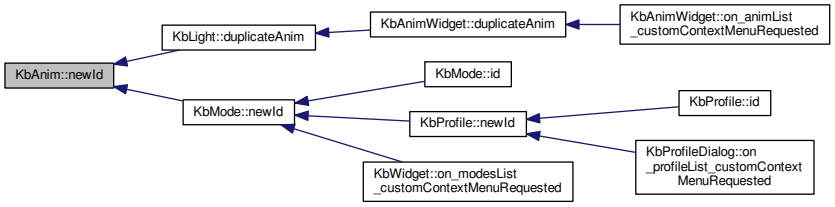
Definition at line 72 of file kbanim.h.

References _guid, and _needsSave.

Referenced by KbLight::duplicateAnim(), and KbMode::newId().

```
72 { _needsSave = true; _guid = QUuid::createUuid(); }
```

Here is the caller graph for this function:



12.22.4.20 float KbAnim::opacity ( ) const [inline]

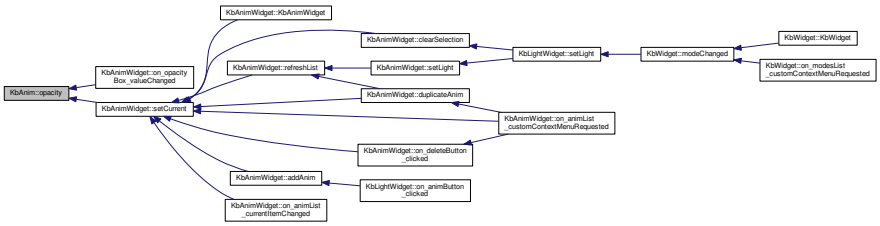
Definition at line 75 of file kbanim.h.

References _opacity.

Referenced by KbAnimWidget::on_opacityBox_valueChanged(), and KbAnimWidget::setCurrent().

```
75 { return _opacity; }
```

Here is the caller graph for this function:



12.22.4.21 void KbAnim::opacity (float *newOpacity*) [inline]

Definition at line 76 of file kbanim.h.

References _needsSave, and _opacity.

```
76 { _needsSave = true; _opacity = newOpacity; }
```

12.22.4.22 QVariant KbAnim::parameter (const QString & *name*) const [inline]

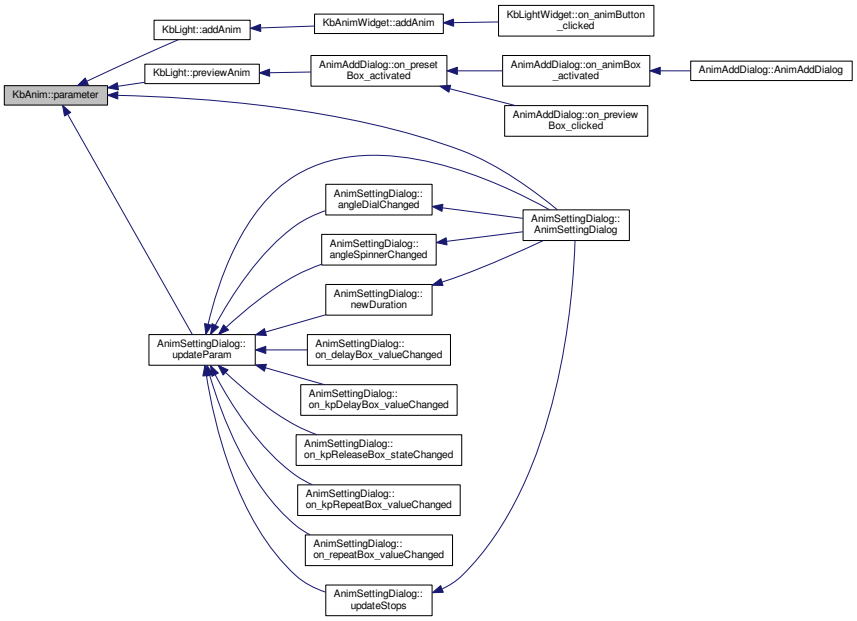
Definition at line 45 of file kbanim.h.

References _parameters.

Referenced by KbLight::addAnim(), AnimSettingDialog::AnimSettingDialog(), KbLight::previewAnim(), and AnimSettingDialog::updateParam().

```
45 { return _parameters.value(name); }
```

Here is the caller graph for this function:



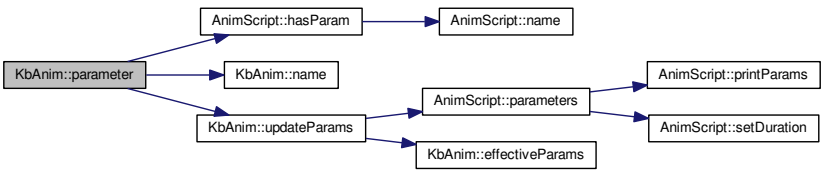
12.22.4.23 void KbAnim::parameter (const QString & *name,* const QVariant & *value*)

Definition at line 114 of file kbanim.cpp.

References _script, _tempParameters, AnimScript::hasParam(), name(), and updateParams().

```
114                                                         {
115     if(!_script->hasParam(name))
116         return;
117     _tempParameters[name] = value;
118     updateParams();
119 }
```

Here is the call graph for this function:



12.22.4.24 void KbAnim::relnit ( )

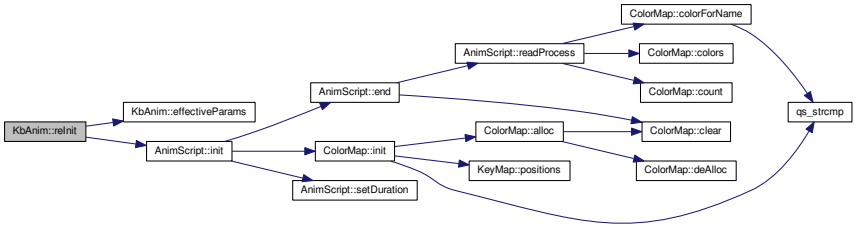
Definition at line 149 of file kbanim.cpp.

References _isActive, _isActiveKp, _keys, _map, _script, effectiveParams(), AnimScript::init(), and repeatKey.

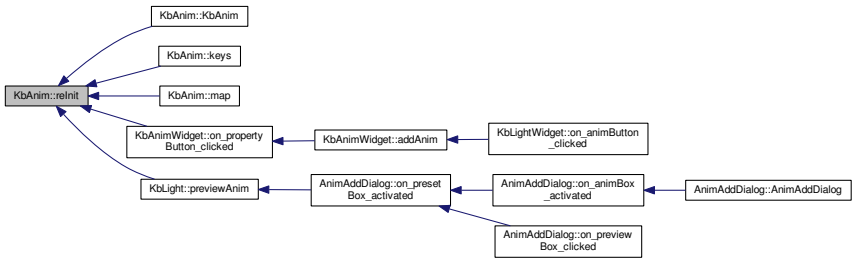
Referenced by KbAnim(), keys(), map(), KbAnimWidget::on_propertyButton_clicked(), and KbLight::previewAnim().

```
149                     {
150     if(_script)
151         _script->init(_map, _keys, effectiveParams());
152     repeatKey = "";
153     _isActive = _isActiveKp = false;
154 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.22.4.25 void KbAnim::resetParams ( )

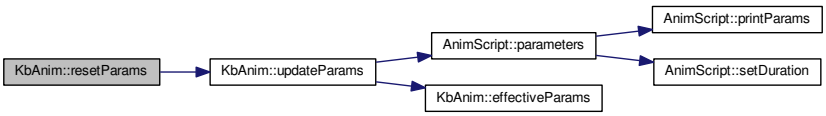
Definition at line 127 of file kbanim.cpp.

References _tempParameters, and updateParams().

Referenced by KbAnimWidget::on_propertyButton_clicked().

```
127                       {
128     _tempParameters.clear();
129     updateParams();
130 }
```

Here is the call graph for this function:



Here is the caller graph for this function:

12.22.4.26 void KbAnim::save (CkbSettings & *settings*)

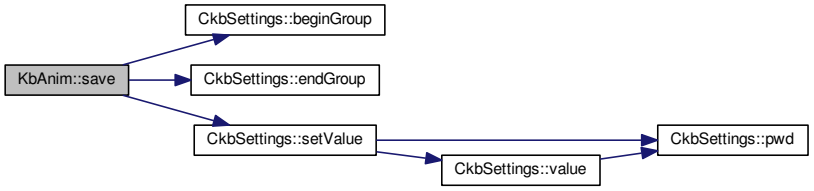
Definition at line 64 of file kbanim.cpp.

References _guid, _keys, _mode, _name, _needsSave, _opacity, _parameters, _scriptGuid, _scriptName, CkbSettings::beginGroup(), CkbSettings::endGroup(), and CkbSettings::setValue().

Referenced by KbLight::save().

```
64                                      {
65     _needsSave = false;
66     settings.beginGroup(_guid.toString().toUpper());
67     settings.setValue("UseRealNames", true);
68     settings.setValue("Keys", _keys);
69     settings.setValue("Name", _name);
70     settings.setValue("Opacity", QString::number(_opacity));
71     settings.setValue("BlendMode", metaObject()->enumerator(metaObject()->indexOfEnumerator("Mode")
       ).valueToKey(_mode));
72     settings.setValue("ScriptName", _scriptName);
73     settings.setValue("ScriptGuid", _scriptGuid.toString().toUpper());
74     settings.beginGroup("Parameters");
75     QMapIterator<QString, QVariant> i(_parameters);
76     while(i.hasNext()){
77         i.next();
78         settings.setValue(i.key(), i.value());
79     }
80     settings.endGroup();
81     settings.endGroup();
82 }
```

Here is the call graph for this function:



Here is the caller graph for this function:

12.22.4.27 const AnimScript* KbAnim::script ( ) const [inline]

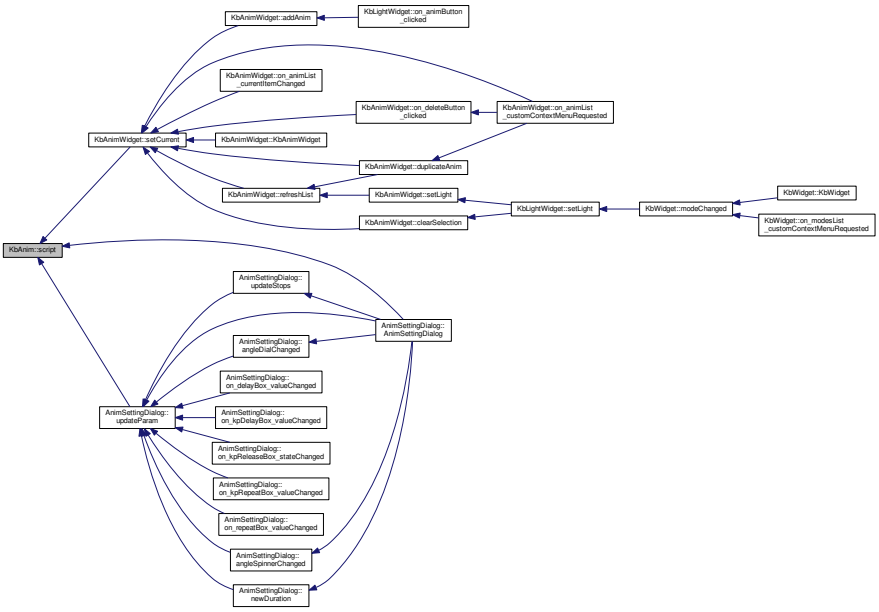
Definition at line 81 of file kbanim.h.

References _script.

Referenced by AnimSettingDialog::AnimSettingDialog(), KbAnimWidget::setCurrent(), and AnimSettingDialog::updateParam().

```
81 { return _script; }
```

Here is the caller graph for this function:



12.22.4.28 const QString& KbAnim::scriptName ( ) const [inline]

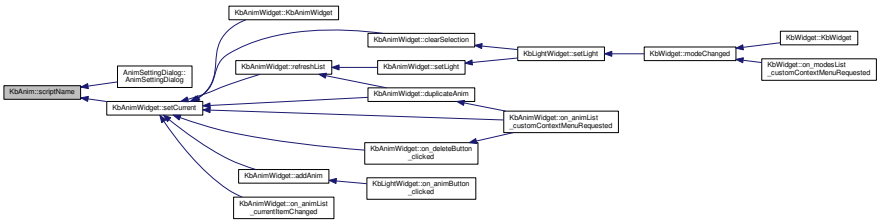
Definition at line 82 of file kbanim.h.

References _scriptName.

Referenced by AnimSettingDialog::AnimSettingDialog(), and KbAnimWidget::setCurrent().

```
82 { return _scriptName; }
```

Here is the caller graph for this function:



12.22.4.29 void KbAnim::stop ( )

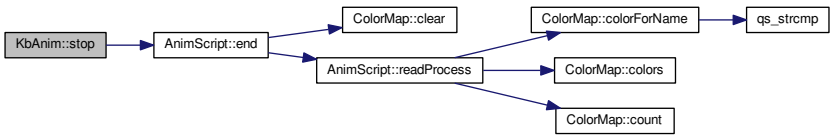
Definition at line 312 of file kbanim.cpp.

References _isActive, _isActiveKp, _script, AnimScript::end(), kpRepeatMsec, kpRepeatTime, kpStopTime, repeatKey, repeatMsec, repeatTime, and stopTime.

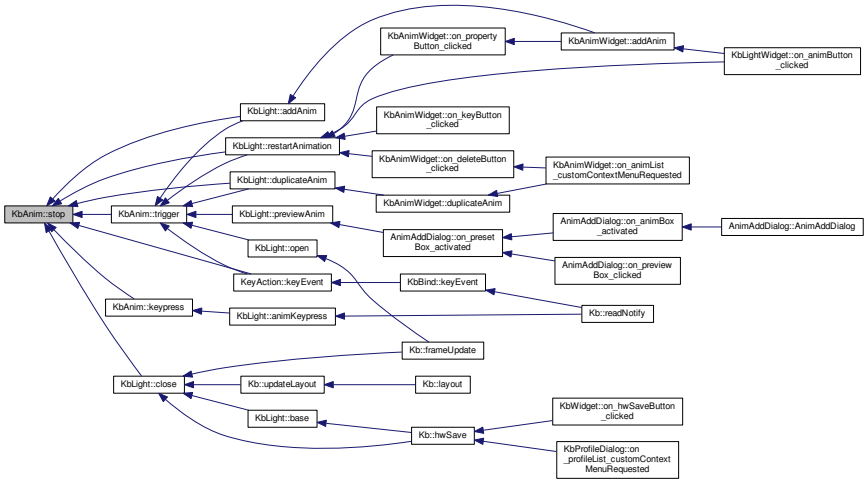
Referenced by KbLight::addAnim(), KbLight::close(), KbLight::duplicateAnim(), KeyAction::keyEvent(), keypress(), KbLight::restartAnimation(), and trigger().

```
312                 {
313     if(_script)
314         _script->end();
315     repeatTime = 0;
316     kpRepeatTime = 0;
317     repeatMsec = 0;
318     kpRepeatMsec = 0;
319     stopTime = 0;
320     kpStopTime = 0;
321     repeatKey = "";
322     _isActive = _isActiveKp = false;
323 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.22.4.30 void KbAnim::trigger (quint64 *timestamp*, bool *ignoreParameter* = `false`)

Definition at line 209 of file kbanim.cpp.

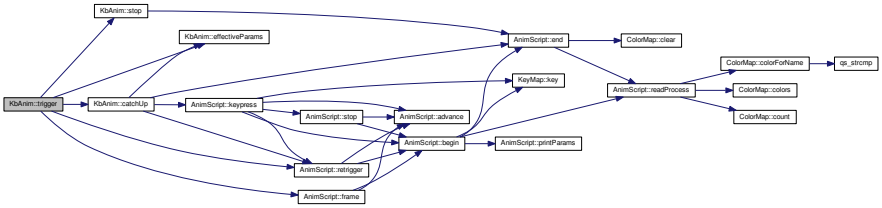
References _isActive, _script, catchUp(), effectiveParams(), AnimScript::frame(), repeatMsec, repeatTime, AnimScript::retrigger(), stop(), and stopTime.

Referenced by KbLight::addAnim(), KbLight::duplicateAnim(), KeyAction::keyEvent(), KbLight::open(), KbLight::previewAnim(), and KbLight::restartAnimation().

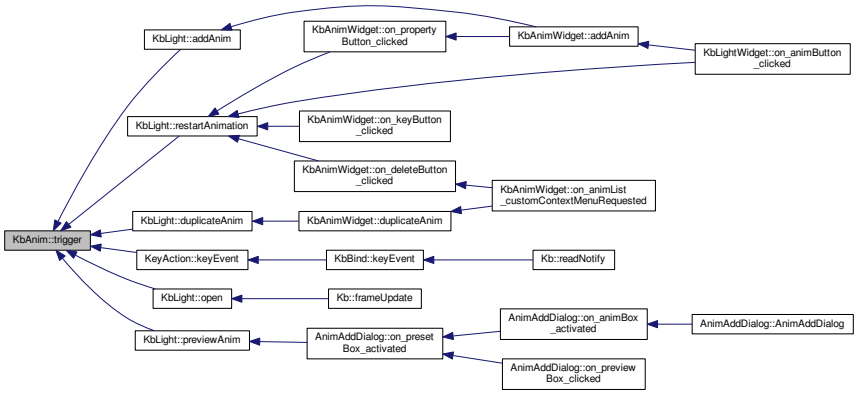
```
209                                                    {
210     if(!_script)
211         return;
212     QMap<QString, QVariant> parameters = effectiveParams();
213
214     catchUp(timestamp);
215     if(parameters.value("trigger").toBool() || ignoreParameter){
216         _isActive = true;
217         int delay = round(parameters.value("delay").toDouble() * 1000.);
218         if(delay > 0){
219             // If delay is enabled, wait to trigger the event
220             timestamp += delay;
221             repeatTime = timestamp;
222         } else
223             _script->retrigger(timestamp, true);
224
225         int repeat = round(parameters.value("repeat").toDouble() * 1000.);
226         if(repeat <= 0){
227             // If no repeat allowed, calculate stop time in seconds
228             repeatMsec = -1;
229             double stop = parameters.value("stop").toDouble();
230             if(stop <= 0)
231                 stopTime = 0;
232             else
233                 stopTime = timestamp + round(stop * 1000.);
234         } else {
235             // If repeat is allowed, calculate stop time in repetitions
236             repeatMsec = repeat;
237             if(delay <= 0)
238                 repeatTime = timestamp + repeat;
239             int stop = parameters.value("stop").toInt();
240             if(stop < 0)
241                 stopTime = 0;
242             else
243                 stopTime = timestamp + repeat * (1 + stop);
244         }
245     }
246     // Ask the script for a frame even if we didn't do anything here. This way ckb knows the script is
    responding.
247     _script->frame(timestamp);
```

```
248 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.22.4.31 void KbAnim::updateParams ( ) [private]

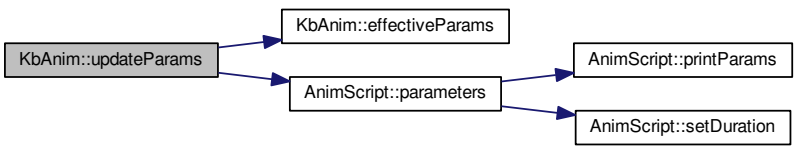
Definition at line 132 of file kbanim.cpp.

References _script, effectiveParams(), AnimScript::parameters(), and repeatKey.

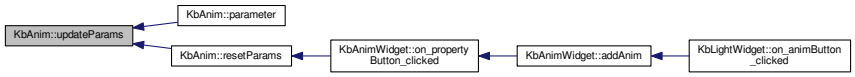
Referenced by parameter(), and resetParams().

```
132                          {
133     if(_script)
134         _script->parameters(effectiveParams());
135     repeatKey = "";
136 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.22.5 Field Documentation

12.22.5.1 QUuid KbAnim::_guid [private]

Definition at line 110 of file kbanim.h.

Referenced by guid(), KbAnim(), newId(), and save().

12.22.5.2 bool KbAnim::_isActive [private]

Definition at line 114 of file kbanim.h.

Referenced by catchUp(), isActive(), keypress(), reInit(), stop(), and trigger().

12.22.5.3 bool KbAnim::_isActiveKp [private]

Definition at line 114 of file kbanim.h.

Referenced by catchUp(), isActive(), keypress(), reInit(), and stop().

12.22.5.4 QStringList KbAnim::_keys [private]

Definition at line 92 of file kbanim.h.

Referenced by KbAnim(), keys(), reInit(), and save().

12.22.5.5 KeyMap KbAnim::_map [private]

Definition at line 91 of file kbanim.h.

Referenced by KbAnim(), map(), and reInit().

12.22.5.6 Mode KbAnim::_mode [private]

Definition at line 113 of file kbanim.h.

Referenced by blend(), KbAnim(), mode(), and save().

12.22.5.7 QString KbAnim::_name [private]

Definition at line 111 of file kbanim.h.

Referenced by KbAnim(), name(), and save().

12.22.5.8 bool KbAnim::_needsSave [private]

Definition at line 115 of file kbanim.h.

Referenced by commitParams(), mode(), name(), needsSave(), newId(), opacity(), and save().

12.22.5.9 float KbAnim::_opacity [private]

Definition at line 112 of file kbanim.h.

Referenced by blend(), KbAnim(), opacity(), and save().

12.22.5.10 QMap<QString, QVariant> KbAnim::_parameters [private]

Definition at line 94 of file kbanim.h.

Referenced by commitParams(), effectiveParams(), hasParameter(), KbAnim(), parameter(), and save().

12.22.5.11 AnimScript* KbAnim::_script [private]

Definition at line 86 of file kbanim.h.

Referenced by blend(), catchUp(), isRunning(), KbAnim(), keypress(), parameter(), reInit(), script(), stop(), trigger(), and updateParams().

12.22.5.12 QUuid KbAnim::_scriptGuid [private]

Definition at line 88 of file kbanim.h.

Referenced by KbAnim(), and save().

12.22.5.13 QString KbAnim::_scriptName [private]

Definition at line 89 of file kbanim.h.

Referenced by KbAnim(), save(), and scriptName().

12.22.5.14 QMap<QString, QVariant> KbAnim::_tempParameters [private]

Definition at line 96 of file kbanim.h.

Referenced by commitParams(), effectiveParams(), parameter(), and resetParams().

12.22.5.15 int KbAnim::kpRepeatMsec [private]

Definition at line 106 of file kbanim.h.

Referenced by catchUp(), keypress(), and stop().

12.22.5.16 quint64 KbAnim::kpRepeatTime [private]

Definition at line 105 of file kbanim.h.

Referenced by catchUp(), keypress(), and stop().

12.22.5.17 quint64 KbAnim::kpStopTime [private]

Definition at line 105 of file kbanim.h.

Referenced by catchUp(), keypress(), and stop().

12.22.5.18 QString KbAnim::repeatKey [private]

Definition at line 104 of file kbanim.h.

Referenced by catchUp(), keypress(), reInit(), stop(), and updateParams().

12.22.5.19 int KbAnim::repeatMsec [private]

Definition at line 106 of file kbanim.h.

Referenced by catchUp(), keypress(), stop(), and trigger().

12.22.5.20 quint64 KbAnim::repeatTime [private]

Definition at line 105 of file kbanim.h.

Referenced by catchUp(), keypress(), stop(), and trigger().

12.22.5.21 quint64 KbAnim::stopTime [private]

Definition at line 105 of file kbanim.h.

Referenced by catchUp(), keypress(), stop(), and trigger().

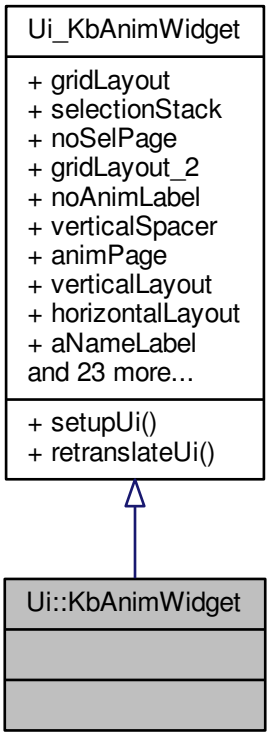
The documentation for this class was generated from the following files:

- src/ckb/kbanim.h
- src/ckb/kbanim.cpp

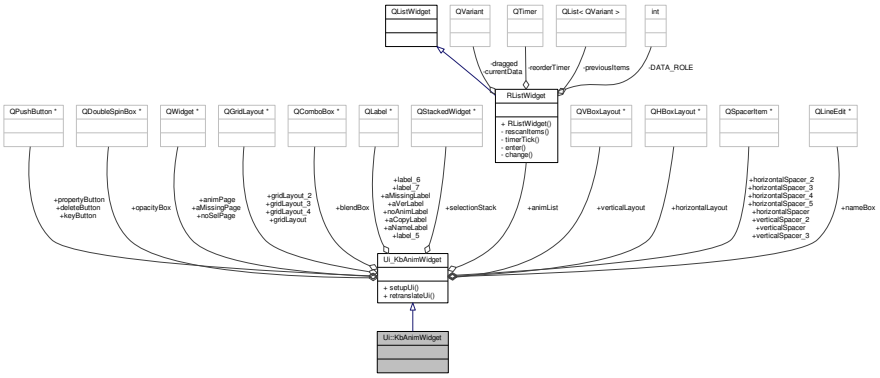
12.23 Ui::KbAnimWidget Class Reference

`#include <src/ckb/ui_kbanimwidget.h>`

Inheritance diagram for Ui::KbAnimWidget:



Collaboration diagram for Ui::KbAnimWidget:



Public Member Functions

- void setupUi (QWidget *KbAnimWidget)
- void retranslateUi (QWidget *KbAnimWidget)

Data Fields

- QGridLayout * gridLayout
- QStackedWidget * selectionStack
- QWidget * noSelPage
- QGridLayout * gridLayout_2
- QLabel * noAnimLabel
- QSpacerItem * verticalSpacer
- QWidget * animPage
- QVBoxLayout * verticalLayout
- QHBoxLayout * horizontalLayout
- QLabel * aNameLabel
- QLabel * aVerLabel
- QSpacerItem * horizontalSpacer
- QLabel * aCopyLabel
- QSpacerItem * verticalSpacer_3
- QGridLayout * gridLayout_3
- QLineEdit * nameBox
- QDoubleSpinBox * opacityBox
- QComboBox * blendBox
- QLabel * label_6
- QSpacerItem * horizontalSpacer_4
- QSpacerItem * horizontalSpacer_2
- QLabel * label_5
- QSpacerItem * horizontalSpacer_3
- QLabel * label_7
- QPushButton * keyButton
- QPushButton * deleteButton
- QPushButton * propertyButton
- QSpacerItem * horizontalSpacer_5
- QSpacerItem * verticalSpacer_2
- QWidget * aMissingPage
- QGridLayout * gridLayout_4
- QLabel * aMissingLabel
- RListWidget * animList

12.23.1 Detailed Description

Definition at line 269 of file ui_kbanimwidget.h.

12.23.2 Member Function Documentation

12.23.2.1 void Ui_KbAnimWidget::retranslateUi (QWidget * *KbAnimWidget*) [inline], [inherited]

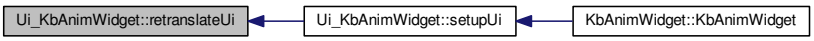
Definition at line 241 of file ui_kbanimwidget.h.

References Ui_KbAnimWidget::aCopyLabel, Ui_KbAnimWidget::aMissingLabel, Ui_KbAnimWidget::aNameLabel, Ui_KbAnimWidget::aVerLabel, Ui_KbAnimWidget::blendBox, Ui_KbAnimWidget::deleteButton, Ui_KbAnimWidget::keyButton, Ui_KbAnimWidget::label_5, Ui_KbAnimWidget::label_6, Ui_KbAnimWidget::label_7, Ui_KbAnimWidget::noAnimLabel, Ui_KbAnimWidget::opacityBox, and Ui_KbAnimWidget::propertyButton.

Referenced by Ui_KbAnimWidget::setupUi().

```
242     {
243         KbAnimWidget->setWindowTitle(QApplication::translate("KbAnimWidget", "Form", Q_NULLPTR));
244         noAnimLabel->setText(QApplication::translate("KbAnimWidget", "No animations in this mode
    ", Q_NULLPTR));
245         aNameLabel->setText(QApplication::translate("KbAnimWidget", "Animation name", Q_NULLPTR))
    ;
246         aVerLabel->setText(QApplication::translate("KbAnimWidget", "v0.0", Q_NULLPTR));
247         aCopyLabel->setText(QApplication::translate("KbAnimWidget", "Copyright \302\251",
    Q_NULLPTR));
248         opacityBox->setSuffix(QApplication::translate("KbAnimWidget", "%", Q_NULLPTR));
249         blendBox->clear();
250         blendBox->insertItems(0, QStringList()
251          << QApplication::translate("KbAnimWidget", "Normal", Q_NULLPTR)
252          << QApplication::translate("KbAnimWidget", "Add", Q_NULLPTR)
253          << QApplication::translate("KbAnimWidget", "Subtract", Q_NULLPTR)
254          << QApplication::translate("KbAnimWidget", "Multiply", Q_NULLPTR)
255          << QApplication::translate("KbAnimWidget", "Divide", Q_NULLPTR)
256         );
257         label_6->setText(QApplication::translate("KbAnimWidget", "Blend mode:", Q_NULLPTR));
258         label_5->setText(QApplication::translate("KbAnimWidget", "Opacity:", Q_NULLPTR));
259         label_7->setText(QApplication::translate("KbAnimWidget", "Name:", Q_NULLPTR));
260         keyButton->setText(QApplication::translate("KbAnimWidget", "Set keys", Q_NULLPTR));
261         deleteButton->setText(QApplication::translate("KbAnimWidget", "Delete animation",
    Q_NULLPTR));
262         propertyButton->setText(QApplication::translate("KbAnimWidget", "Edit properties",
    Q_NULLPTR));
263         aMissingLabel->setText(QApplication::translate("KbAnimWidget", "Mising animation",
    Q_NULLPTR));
264     } // retranslateUi
```

Here is the caller graph for this function:



12.23.2.2 void Ui_KbAnimWidget::setupUi (QWidget * *KbAnimWidget*) [inline],[inherited]

Definition at line 69 of file ui_kbanimwidget.h.

References Ui_KbAnimWidget::aCopyLabel, Ui_KbAnimWidget::aMissingLabel, Ui_KbAnimWidget::aMissingPage, Ui_KbAnimWidget::aNameLabel, Ui_KbAnimWidget::animList, Ui_KbAnimWidget::animPage, Ui_KbAnimWidget::aVerLabel, Ui_KbAnimWidget::blendBox, Ui_KbAnimWidget::deleteButton, Ui_KbAnimWidget::gridLayout, Ui_KbAnimWidget::gridLayout_2, Ui_KbAnimWidget::gridLayout_3, Ui_KbAnimWidget::gridLayout_4, Ui_KbAnimWidget::horizontalLayout, Ui_KbAnimWidget::horizontalSpacer, Ui_KbAnimWidget::horizontalSpacer_2, Ui_KbAnimWidget::horizontalSpacer_3, Ui_KbAnimWidget::horizontalSpacer_4, Ui_KbAnimWidget::horizontalSpacer_5, Ui_KbAnimWidget::keyButton, Ui_KbAnimWidget::label_5, Ui_KbAnimWidget::label_6, Ui_KbAnimWidget::label_7, Ui_KbAnimWidget::nameBox, Ui_KbAnimWidget::noAnimLabel, Ui_KbAnimWidget::noSelPage, Ui_KbAnimWidget::opacityBox, Ui_KbAnimWidget::propertyButton, Ui_KbAnimWidget::retranslateUi(), Ui_KbAnimWidget::selectionStack, Ui_KbAnimWidget::verticalLayout, Ui_KbAnimWidget::verticalSpacer, Ui_KbAnimWidget::verticalSpacer_2, and Ui_KbAnimWidget::verticalSpacer_3.

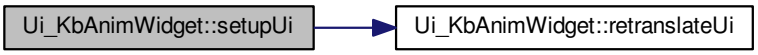
Referenced by KbAnimWidget::KbAnimWidget().

```
70     {
71         if (KbAnimWidget->objectName().isEmpty())
72             KbAnimWidget->setObjectName(QStringLiteral("KbAnimWidget"));
73         KbAnimWidget->resize(613, 300);
74         gridLayout = new QGridLayout(KbAnimWidget);
75         gridLayout->setObjectName(QStringLiteral("gridLayout"));
76         gridLayout->setContentsMargins(0, 0, 0, 0);
77         selectionStack = new QStackedWidget(KbAnimWidget);
78         selectionStack->setObjectName(QStringLiteral("selectionStack"));
79         noSelPage = new QWidget();
80         noSelPage->setObjectName(QStringLiteral("noSelPage"));
81         gridLayout_2 = new QGridLayout(noSelPage);
82         gridLayout_2->setObjectName(QStringLiteral("gridLayout_2"));
83         gridLayout_2->setContentsMargins(6, 6, 6, 6);
84         noAnimLabel = new QLabel(noSelPage);
```

```
85          noAnimLabel->setObjectName(QStringLiteral("noAnimLabel"));
86
87          gridLayout_2->addWidget(noAnimLabel, 0, 0, 1, 1);
88
89          verticalSpacer = new QSpacerItem(20, 40, QSizePolicy::Minimum, QSizePolicy::Expanding
    );
90
91          gridLayout_2->addItem(verticalSpacer, 1, 0, 1, 1);
92
93          selectionStack->addWidget(noSelPage);
94          animPage = new QWidget();
95          animPage->setObjectName(QStringLiteral("animPage"));
96          verticalLayout = new QVBoxLayout(animPage);
97          verticalLayout->setObjectName(QStringLiteral("verticalLayout"));
98          verticalLayout->setContentsMargins(6, 0, 0, 0);
99          horizontalLayout = new QHBoxLayout();
100          horizontalLayout->setObjectName(QStringLiteral("horizontalLayout"));
101          aNameLabel = new QLabel(animPage);
102          aNameLabel->setObjectName(QStringLiteral("aNameLabel"));
103          QFont font;
104          font.setBold(true);
105          font.setWeight(75);
106          aNameLabel->setFont(font);
107
108          horizontalLayout->addWidget(aNameLabel);
109
110          aVerLabel = new QLabel(animPage);
111          aVerLabel->setObjectName(QStringLiteral("aVerLabel"));
112
113          horizontalLayout->addWidget(aVerLabel);
114
115          horizontalSpacer = new QSpacerItem(40, 20, QSizePolicy::Expanding,
    QSizePolicy::Minimum);
116
117          horizontalLayout->addItem(horizontalSpacer);
118
119          aCopyLabel = new QLabel(animPage);
120          aCopyLabel->setObjectName(QStringLiteral("aCopyLabel"));
121
122          horizontalLayout->addWidget(aCopyLabel);
123
124
125          verticalLayout->addLayout(horizontalLayout);
126
127          verticalSpacer_3 = new QSpacerItem(20, 40, QSizePolicy::Minimum,
    QSizePolicy::Expanding);
128
129          verticalLayout->addItem(verticalSpacer_3);
130
131          gridLayout_3 = new QGridLayout();
132          gridLayout_3->setObjectName(QStringLiteral("gridLayout_3"));
133          nameBox = new QLineEdit(animPage);
134          nameBox->setObjectName(QStringLiteral("nameBox"));
135          QSizePolicy sizePolicy(QSizePolicy::Fixed, QSizePolicy::Fixed);
136          sizePolicy.setHorizontalStretch(0);
137          sizePolicy.setVerticalStretch(0);
138          sizePolicy.setHeightForWidth(nameBox->sizePolicy().hasHeightForWidth());
139          nameBox->setSizePolicy(sizePolicy);
140          nameBox->setMinimumSize(QSize(150, 0));
141          nameBox->setMaximumSize(QSize(150, 16777215));
142
143          gridLayout_3->addWidget(nameBox, 0, 3, 1, 2);
144
145          opacityBox = new QDoubleSpinBox(animPage);
146          opacityBox->setObjectName(QStringLiteral("opacityBox"));
147          opacityBox->setDecimals(1);
148          opacityBox->setValue(100);
149
150          gridLayout_3->addWidget(opacityBox, 1, 3, 1, 1);
151
152          blendBox = new QComboBox(animPage);
153          blendBox->setObjectName(QStringLiteral("blendBox"));
154
155          gridLayout_3->addWidget(blendBox, 2, 3, 1, 2);
156
157          label_6 = new QLabel(animPage);
158          label_6->setObjectName(QStringLiteral("label_6"));
159
160          gridLayout_3->addWidget(label_6, 2, 1, 1, 1);
161
162          horizontalSpacer_4 = new QSpacerItem(0, 0, QSizePolicy::Fixed,
    QSizePolicy::Minimum);
163
164          gridLayout_3->addItem(horizontalSpacer_4, 0, 0, 1, 1);
165
166          horizontalSpacer_2 = new QSpacerItem(30, 20, QSizePolicy::Fixed,
    QSizePolicy::Minimum);
```

```
167
168         gridLayout_3->addItem(horizontalSpacer_2, 0, 2, 1, 1);
169
170         label_5 = new QLabel(animPage);
171         label_5->setObjectName(QStringLiteral("label_5"));
172
173         gridLayout_3->addWidget(label_5, 1, 1, 1, 1);
174
175         horizontalSpacer_3 = new QSpacerItem(40, 20, QSizePolicy::Expanding,
    QSizePolicy::Minimum);
176
177         gridLayout_3->addItem(horizontalSpacer_3, 0, 5, 1, 1);
178
179         label_7 = new QLabel(animPage);
180         label_7->setObjectName(QStringLiteral("label_7"));
181
182         gridLayout_3->addWidget(label_7, 0, 1, 1, 1);
183
184         keyButton = new QPushButton(animPage);
185         keyButton->setObjectName(QStringLiteral("keyButton"));
186
187         gridLayout_3->addWidget(keyButton, 1, 6, 1, 1);
188
189         deleteButton = new QPushButton(animPage);
190         deleteButton->setObjectName(QStringLiteral("deleteButton"));
191
192         gridLayout_3->addWidget(deleteButton, 2, 6, 1, 1);
193
194         propertyButton = new QPushButton(animPage);
195         propertyButton->setObjectName(QStringLiteral("propertyButton"));
196
197         gridLayout_3->addWidget(propertyButton, 0, 6, 1, 1);
198
199         horizontalSpacer_5 = new QSpacerItem(40, 20, QSizePolicy::Expanding,
    QSizePolicy::Minimum);
200
201         gridLayout_3->addItem(horizontalSpacer_5, 1, 4, 1, 1);
202
203
204         verticalLayout->addLayout(gridLayout_3);
205
206         verticalSpacer_2 = new QSpacerItem(20, 40, QSizePolicy::Minimum,
    QSizePolicy::Expanding);
207
208         verticalLayout->addItem(verticalSpacer_2);
209
210         selectionStack->addWidget(animPage);
211         aMissingPage = new QWidget();
212         aMissingPage->setObjectName(QStringLiteral("aMissingPage"));
213         gridLayout_4 = new QGridLayout(aMissingPage);
214         gridLayout_4->setObjectName(QStringLiteral("gridLayout_4"));
215         aMissingLabel = new QLabel(aMissingPage);
216         aMissingLabel->setObjectName(QStringLiteral("aMissingLabel"));
217         aMissingLabel->setFont(font);
218
219         gridLayout_4->addWidget(aMissingLabel, 0, 0, 1, 1);
220
221         selectionStack->addWidget(aMissingPage);
222
223         gridLayout->addWidget(selectionStack, 0, 2, 2, 1);
224
225         animList = new RListWidget(KbAnimWidget);
226         animList->setObjectName(QStringLiteral("animList"));
227         animList->setMaximumSize(QSize(130, 16777215));
228         animList->setContextMenuPolicy(Qt::CustomContextMenu);
229         animList->setFrameShadow(QFrame::Plain);
230         animList->setDragDropMode(QAbstractItemView::InternalMove);
231         animList->setMovement(QListView::Snap);
232
233         gridLayout->addWidget(animList, 0, 0, 2, 1);
234
235
236         retranslateUi(KbAnimWidget);
237
238         QMetaObject::connectSlotsByName(KbAnimWidget);
239     } // setupUi
```

Here is the call graph for this function:



Here is the caller graph for this function:

12.23.3 Field Documentation

12.23.3.1 QLabel* Ui_KbAnimWidget::aCopyLabel [inherited]

Definition at line 47 of file ui_kbanimwidget.h.

Referenced by Ui_KbAnimWidget::retranslateUi(), KbAnimWidget::setCurrent(), and Ui_KbAnimWidget::setupUi().

12.23.3.2 QLabel* Ui_KbAnimWidget::aMissingLabel [inherited]

Definition at line 66 of file ui_kbanimwidget.h.

Referenced by Ui_KbAnimWidget::retranslateUi(), KbAnimWidget::setCurrent(), and Ui_KbAnimWidget::setupUi().

12.23.3.3 QWidget* Ui_KbAnimWidget::aMissingPage [inherited]

Definition at line 64 of file ui_kbanimwidget.h.

Referenced by Ui_KbAnimWidget::setupUi().

12.23.3.4 QLabel* Ui_KbAnimWidget::aNameLabel [inherited]

Definition at line 44 of file ui_kbanimwidget.h.

Referenced by Ui_KbAnimWidget::retranslateUi(), KbAnimWidget::setCurrent(), and Ui_KbAnimWidget::setupUi().

12.23.3.5 RListWidget* Ui_KbAnimWidget::animList [inherited]

Definition at line 67 of file ui_kbanimwidget.h.

Referenced by KbAnimWidget::addAnim(), KbAnimWidget::clearSelection(), KbAnimWidget::duplicateAnim(), KbAnimWidget::KbAnimWidget(), KbAnimWidget::on_animList_customContextMenuRequested(), KbAnimWidget::on_deleteButton_clicked(), KbAnimWidget::on_nameBox_textEdited(), KbAnimWidget::refreshList(), KbAnimWidget::reorderAnims(), and Ui_KbAnimWidget::setupUi().

12.23.3.6 QWidget* Ui_KbAnimWidget::animPage [inherited]

Definition at line 41 of file ui_kbanimwidget.h.

Referenced by Ui_KbAnimWidget::setupUi().

12.23.3.7 QLabel* Ui_KbAnimWidget::aVerLabel [inherited]

Definition at line 45 of file ui_kbanimwidget.h.

Referenced by Ui_KbAnimWidget::retranslateUi(), KbAnimWidget::setCurrent(), and Ui_KbAnimWidget::setupUi().

12.23.3.8 QComboBox* Ui_KbAnimWidget::blendBox [inherited]

Definition at line 52 of file ui_kbanimwidget.h.

Referenced by Ui_KbAnimWidget::retranslateUi(), KbAnimWidget::setCurrent(), and Ui_KbAnimWidget::setupUi().

12.23.3.9 QPushButton* Ui_KbAnimWidget::deleteButton [inherited]

Definition at line 60 of file ui_kbanimwidget.h.

Referenced by Ui_KbAnimWidget::retranslateUi(), and Ui_KbAnimWidget::setupUi().

12.23.3.10 QGridLayout* Ui_KbAnimWidget::gridLayout [inherited]

Definition at line 35 of file ui_kbanimwidget.h.

Referenced by Ui_KbAnimWidget::setupUi().

12.23.3.11 QGridLayout* Ui_KbAnimWidget::gridLayout_2 [inherited]

Definition at line 38 of file ui_kbanimwidget.h.

Referenced by Ui_KbAnimWidget::setupUi().

12.23.3.12 QGridLayout* Ui_KbAnimWidget::gridLayout_3 [inherited]

Definition at line 49 of file ui_kbanimwidget.h.

Referenced by Ui_KbAnimWidget::setupUi().

12.23.3.13 QGridLayout* Ui_KbAnimWidget::gridLayout_4 [inherited]

Definition at line 65 of file ui_kbanimwidget.h.

Referenced by Ui_KbAnimWidget::setupUi().

12.23.3.14 QHBoxLayout* Ui_KbAnimWidget::horizontalLayout [inherited]

Definition at line 43 of file ui_kbanimwidget.h.

Referenced by Ui_KbAnimWidget::setupUi().

12.23.3.15 QSpacerItem* Ui_KbAnimWidget::horizontalSpacer [inherited]

Definition at line 46 of file ui_kbanimwidget.h.

Referenced by Ui_KbAnimWidget::setupUi().

12.23.3.16 QSpacerItem* Ui_KbAnimWidget::horizontalSpacer_2 [inherited]

Definition at line 55 of file ui_kbanimwidget.h.

Referenced by Ui_KbAnimWidget::setupUi().

12.23.3.17 QSpacerItem* Ui_KbAnimWidget::horizontalSpacer_3 [inherited]

Definition at line 57 of file ui_kbanimwidget.h.

Referenced by Ui_KbAnimWidget::setupUi().

12.23.3.18 QSpacerItem* Ui_KbAnimWidget::horizontalSpacer_4 [inherited]

Definition at line 54 of file ui_kbanimwidget.h.

Referenced by Ui_KbAnimWidget::setupUi().

12.23.3.19 QSpacerItem* Ui_KbAnimWidget::horizontalSpacer_5 [inherited]

Definition at line 62 of file ui_kbanimwidget.h.

Referenced by Ui_KbAnimWidget::setupUi().

12.23.3.20 QPushButton* Ui_KbAnimWidget::keyButton [inherited]

Definition at line 59 of file ui_kbanimwidget.h.

Referenced by Ui_KbAnimWidget::retranslateUi(), KbAnimWidget::setSelectedKeys(), and Ui_KbAnimWidget::setupUi().

12.23.3.21 QLabel* Ui_KbAnimWidget::label_5 [inherited]

Definition at line 56 of file ui_kbanimwidget.h.

Referenced by Ui_KbAnimWidget::retranslateUi(), and Ui_KbAnimWidget::setupUi().

12.23.3.22 QLabel* Ui_KbAnimWidget::label_6 [inherited]

Definition at line 53 of file ui_kbanimwidget.h.

Referenced by Ui_KbAnimWidget::retranslateUi(), and Ui_KbAnimWidget::setupUi().

12.23.3.23 QLabel* Ui_KbAnimWidget::label_7 [inherited]

Definition at line 58 of file ui_kbanimwidget.h.

Referenced by Ui_KbAnimWidget::retranslateUi(), and Ui_KbAnimWidget::setupUi().

12.23.3.24 QLineEdit* Ui_KbAnimWidget::nameBox [inherited]

Definition at line 50 of file ui_kbanimwidget.h.

Referenced by KbAnimWidget::on_animList_itemChanged(), KbAnimWidget::on_propertyButton_clicked(), KbAnimWidget::setCurrent(), and Ui_KbAnimWidget::setupUi().

12.23.3.25 QLabel* Ui_KbAnimWidget::noAnimLabel [inherited]

Definition at line 39 of file ui_kbanimwidget.h.

Referenced by KbAnimWidget::addAnim(), KbAnimWidget::duplicateAnim(), KbAnimWidget::on_deleteButton_clicked(), KbAnimWidget::refreshList(), Ui_KbAnimWidget::retranslateUi(), and Ui_KbAnimWidget::setupUi().

12.23.3.26 QWidget* Ui_KbAnimWidget::noSelPage [inherited]

Definition at line 37 of file ui_kbanimwidget.h.

Referenced by Ui_KbAnimWidget::setupUi().

12.23.3.27 QDoubleSpinBox* Ui_KbAnimWidget::opacityBox [inherited]

Definition at line 51 of file ui_kbanimwidget.h.

Referenced by Ui_KbAnimWidget::retranslateUi(), KbAnimWidget::setCurrent(), and Ui_KbAnimWidget::setupUi().

12.23.3.28 QPushButton* Ui_KbAnimWidget::propertyButton [inherited]

Definition at line 61 of file ui_kbanimwidget.h.

Referenced by Ui_KbAnimWidget::retranslateUi(), and Ui_KbAnimWidget::setupUi().

12.23.3.29 QStackedWidget* Ui_KbAnimWidget::selectionStack [inherited]

Definition at line 36 of file ui_kbanimwidget.h.

Referenced by KbAnimWidget::setCurrent(), and Ui_KbAnimWidget::setupUi().

12.23.3.30 QVBoxLayout* Ui_KbAnimWidget::verticalLayout [inherited]

Definition at line 42 of file ui_kbanimwidget.h.

Referenced by Ui_KbAnimWidget::setupUi().

12.23.3.31 QSpacerItem* Ui_KbAnimWidget::verticalSpacer [inherited]

Definition at line 40 of file ui_kbanimwidget.h.

Referenced by Ui_KbAnimWidget::setupUi().

12.23.3.32 QSpacerItem* Ui_KbAnimWidget::verticalSpacer_2 [inherited]

Definition at line 63 of file ui_kbanimwidget.h.

Referenced by Ui_KbAnimWidget::setupUi().

12.23.3.33 QSpacerItem* Ui_KbAnimWidget::verticalSpacer_3 [inherited]

Definition at line 48 of file ui_kbanimwidget.h.

Referenced by Ui_KbAnimWidget::setupUi().

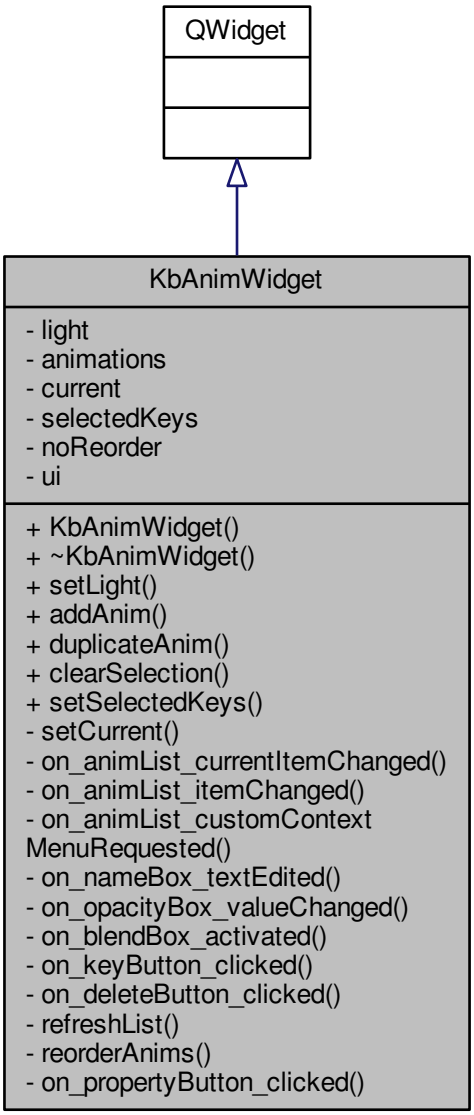
The documentation for this class was generated from the following file:

- src/ckb/ui_kbanimwidget.h

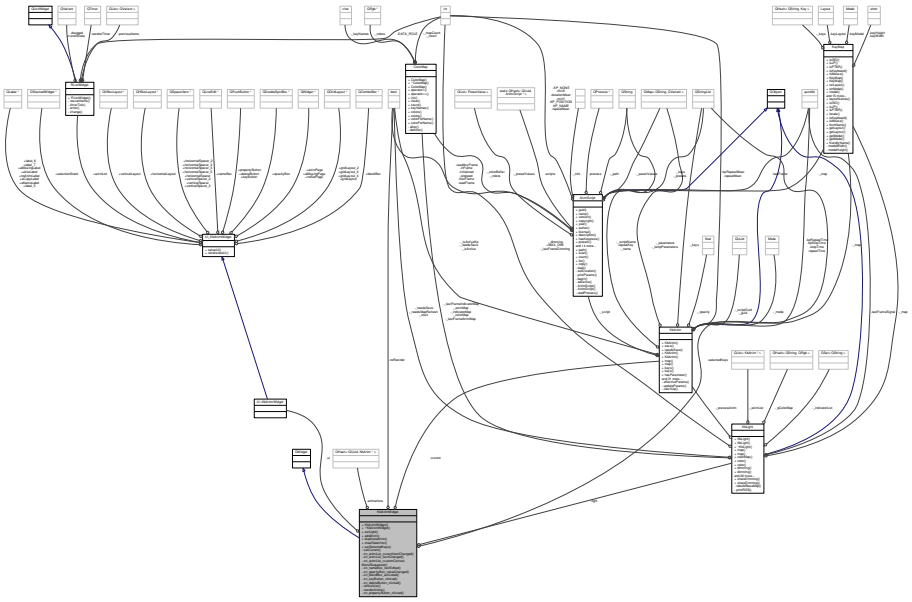
12.24 KbAnimWidget Class Reference

`#include <src/ckb/kbanimwidget.h>`

Inheritance diagram for KbAnimWidget:



Collaboration diagram for KbAnimWidget:



Signals

- void animChanged (KbAnim ∗selection)
- void didUpdateSelection (QStringList keys)

Public Member Functions

- KbAnimWidget (QWidget ∗parent=0)
- ∼KbAnimWidget ()
- void setLight (KbLight ∗newLight)
- void addAnim (const AnimScript ∗base, const QStringList &keyList, const QString &name, const QMap< QString, QVariant > &preset)
- void duplicateAnim (KbAnim ∗old)
- void clearSelection ()
- void setSelectedKeys (const QStringList &keys)

Private Slots

- void on_animList_currentItemChanged (QListWidgetItem ∗current, QListWidgetItem ∗previous)
- void on_animList_itemChanged (QListWidgetItem ∗item)
- void on_animList_customContextMenuRequested (const QPoint &pos)
- void on_nameBox_textEdited (const QString &arg1)
- void on_opacityBox_valueChanged (double arg1)
- void on_blendBox_activated (int index)
- void on_keyButton_clicked ()
- void on_deleteButton_clicked ()
- void refreshList ()
- void reorderAnims ()
- void on_propertyButton_clicked ()

Private Member Functions

- void setCurrent (KbAnim *newCurrent)

Private Attributes

- KbLight * light
- QHash< QUuid, KbAnim * > animations
- KbAnim * current
- QStringList selectedKeys
- bool noReorder
- Ui::KbAnimWidget * ui

12.24.1 Detailed Description

Definition at line 15 of file kbanimwidget.h.

12.24.2 Constructor & Destructor Documentation

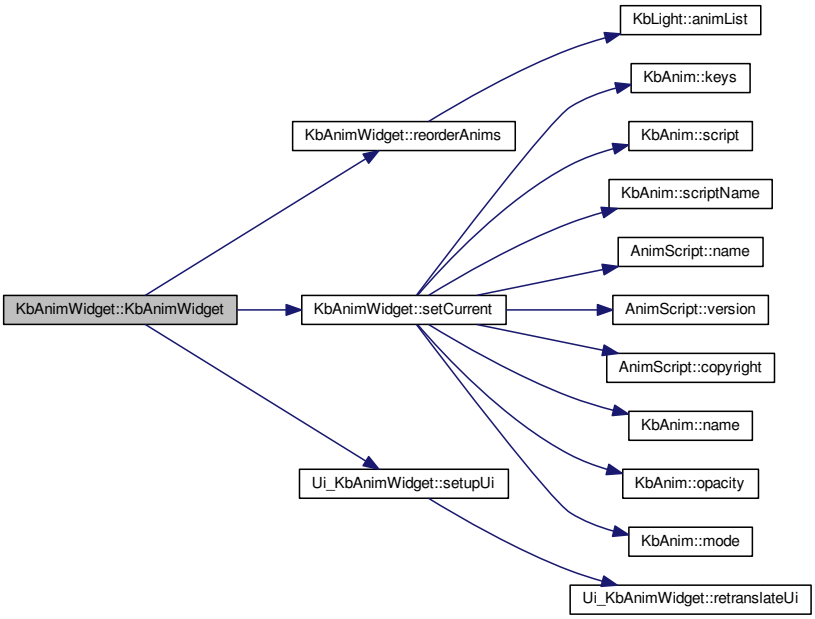
12.24.2.1 KbAnimWidget::KbAnimWidget (QWidget * *parent* = 0) [explicit]

Definition at line 7 of file kbanimwidget.cpp.

References Ui_KbAnimWidget::animList, reorderAnims(), setCurrent(), Ui_KbAnimWidget::setupUi(), and ui.

```
7                                              :
8     QWidget(parent), light(0), current(0), noReorder(false),
9     ui(new Ui::KbAnimWidget)
10 {
11     ui->setupUi(this);
12     ui->animList->setVisible(false);
13     setCurrent(0);
14     connect(ui->animList, SIGNAL(orderChanged()), this, SLOT(
      reorderAnims()));
15 }
```

Here is the call graph for this function:



12.24.2.2 KbAnimWidget::~KbAnimWidget ( )

Definition at line 17 of file kbanimwidget.cpp.

References ui.

```
17                              {
18     delete ui;
19 }
```

12.24.3 Member Function Documentation

12.24.3.1 void KbAnimWidget::addAnim (const AnimScript * *base*, const QStringList & *keyList*, const QString & *name*, const QMap< QString, QVariant > & *preset*)

Definition at line 82 of file kbanimwidget.cpp.

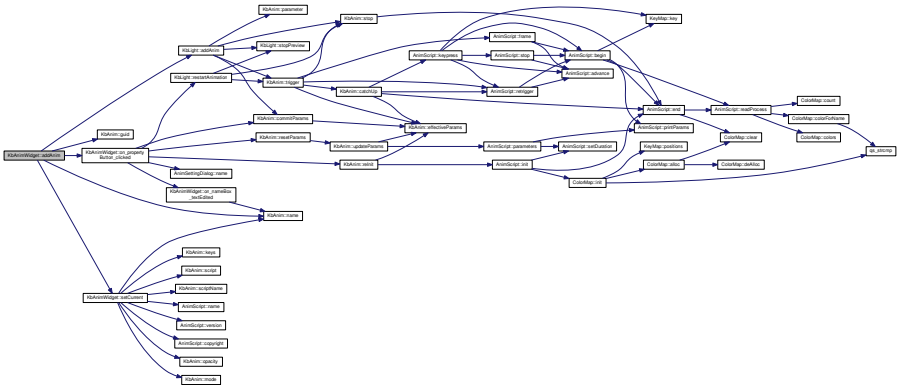
References KbLight::addAnim(), animations, Ui_KbAnimWidget::animList, KbAnim::guid(), light, KbAnim::name(), Ui_KbAnimWidget::noAnimLabel, noReorder, on_propertyButton_clicked(), setCurrent(), and ui.

Referenced by KbLightWidget::on_animButton_clicked().

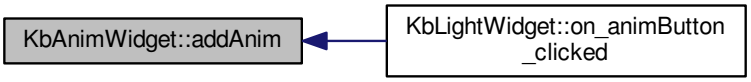
```
82
                                    {
83     if(!light)
84         return;
85     noReorder = true;
86     KbAnim* animation = light->addAnim(base, keyList, name, preset);
87     QListWidgetItem* item = new QListWidgetItem(animation->name(), ui->
      animList);
88     item->setData(Qt::UserRole, animation->guid());
89     item->setFlags(item->flags() | Qt::ItemIsEditable);
90     animations[animation->guid()] = animation;
91     ui->animList->addItem(item);
92     ui->animList->setCurrentItem(item);
```

```
93      ui->animList->setVisible(true);
94      ui->noAnimLabel->setVisible(false);
95
96      setCurrent(animation);
97      noReorder = false;
98
99      // Activate settings dialog
100      on_propertyButton_clicked();
101 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



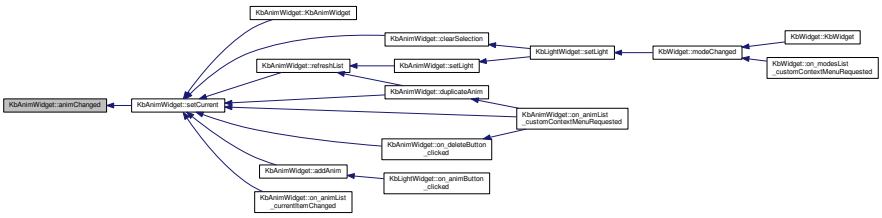
12.24.3.2 void KbAnimWidget::animChanged (KbAnim * *selection*) [signal]

Definition at line 213 of file moc_kbanimwidget.cpp.

Referenced by setCurrent().

```
214 {
215     void *_a[] = { Q_NULLPTR, const_cast<void*>(reinterpret_cast<const void*>(&_t1)) };
216     QMetaObject::activate(this, &staticMetaObject, 0, _a);
217 }
```

Here is the caller graph for this function:



12.24.3.3 void KbAnimWidget::clearSelection ( )

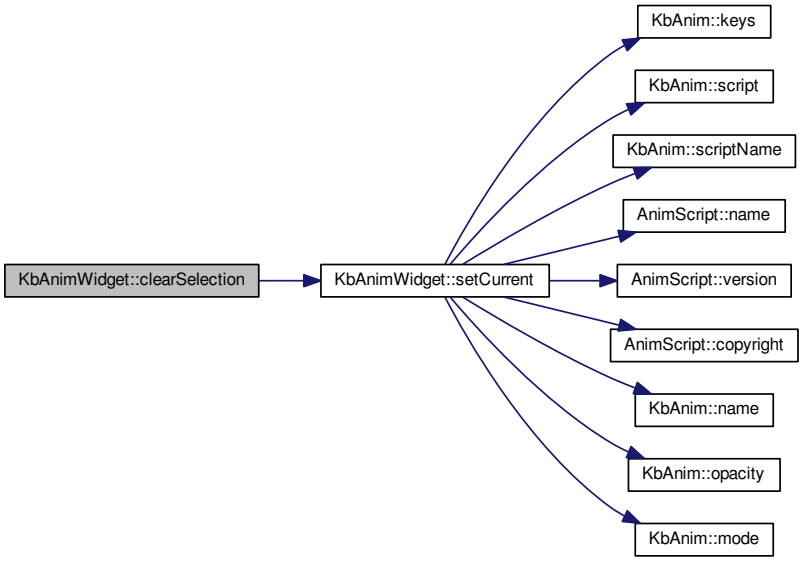
Definition at line 77 of file kbanimwidget.cpp.

References Ui_KbAnimWidget::animList, setCurrent(), and ui.

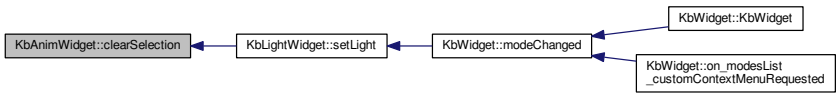
Referenced by KbLightWidget::setLight().

```
77                                  {
78     ui->animList->setCurrentItem(0);
79     setCurrent(0);
80 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



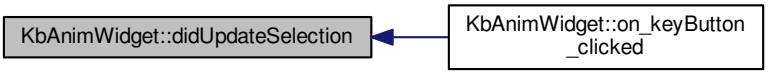
12.24.3.4 void KbAnimWidget::didUpdateSelection (QStringList *keys*) [signal]

Definition at line 220 of file moc_kbanimwidget.cpp.

Referenced by on_keyButton_clicked().

```
221 {
222     void *_a[] = { Q_NULLPTR, const_cast<void*>(reinterpret_cast<const void*>(&_t1)) };
223     QMetaObject::activate(this, &staticMetaObject, 1, _a);
224 }
```

Here is the caller graph for this function:



12.24.3.5 void KbAnimWidget::duplicateAnim (KbAnim * *old*)

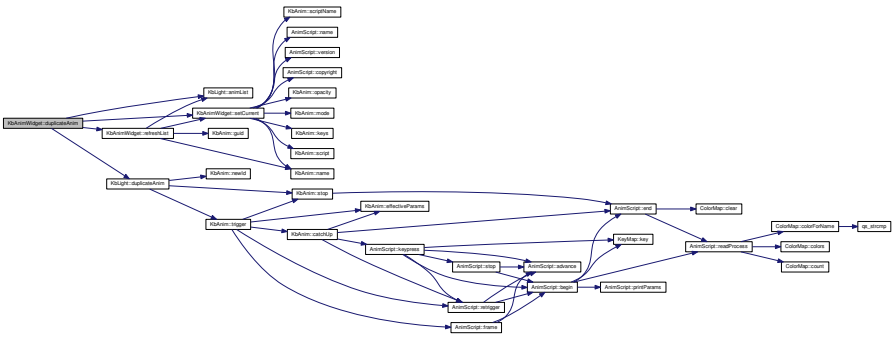
Definition at line 103 of file kbanimwidget.cpp.

References KbLight::animList(), Ui_KbAnimWidget::animList, KbLight::duplicateAnim(), light, Ui_KbAnimWidget::noAnimLabel, noReorder, refreshList(), setCurrent(), and ui.

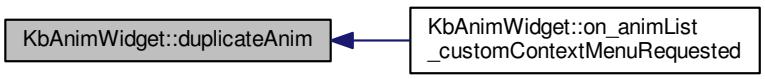
Referenced by on_animList_customContextMenuRequested().

```
103                                          {
104     if(!light)
105         return;
106     noReorder = true;
107     KbAnim* animation = light->duplicateAnim(old);
108     // Refresh the list. insertItem doesn't seem to place the item in the correct position on its own...
109     refreshList();
110     ui->animList->setCurrentRow(light->animList().indexOf(animation));
111     ui->animList->setVisible(true);
112     ui->noAnimLabel->setVisible(false);
113 
114     setCurrent(animation);
115     noReorder = false;
116 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



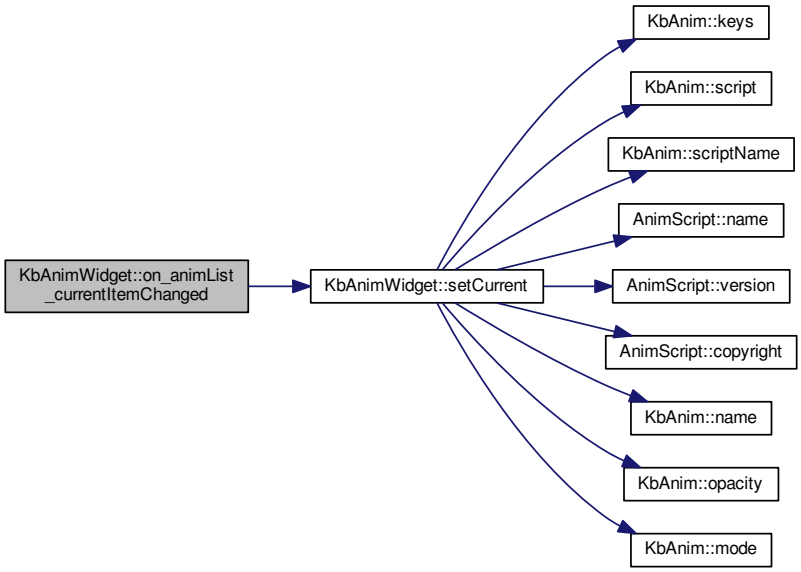
12.24.3.6 void KbAnimWidget::on_animList_currentItemChanged (QListWidgetItem * *current*, QListWidgetItem * *previous*) [private], [slot]

Definition at line 151 of file kbanimwidget.cpp.

References animations, and setCurrent().

```
151                                                                                                                 {
152     if(!current)
153         setCurrent(0);
154     else
155         setCurrent(animations[current->data(Qt::UserRole).toUuid()]);
156 }
```

Here is the call graph for this function:



12.24.3.7 void KbAnimWidget::on_animList_customContextMenuRequested (const QPoint & *pos*) [private], [slot]

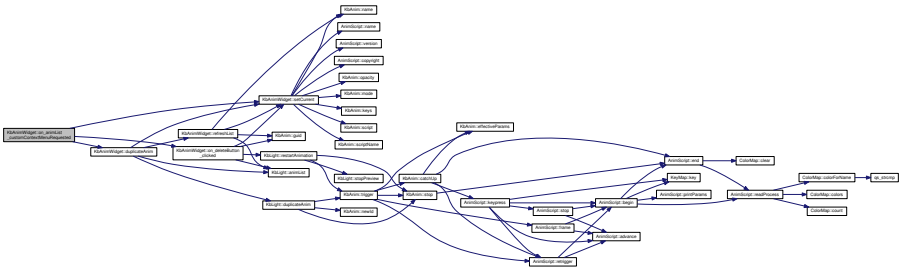
Definition at line 169 of file kbanimwidget.cpp.

References animations, Ui_KbAnimWidget::animList, current, duplicateAnim(), on_deleteButton_clicked(), setCurrent(), and ui.

```
169                                                                                 {
170     QListWidgetItem* item = ui->animList->itemAt(pos);
171     if(!item)
172         return;
173     setCurrent(animations[item->data(Qt::UserRole).toUuid()]);
174
175     QMenu menu(this);
176     QAction* rename = new QAction("Rename...", this);
177     QAction* duplicate = new QAction("Duplicate", this);
178     QAction* del = new QAction("Delete", this);
179     menu.addAction(rename);
180     menu.addAction(duplicate);
181     menu.addAction(del);
182     QAction* result = menu.exec(QCursor::pos());
183     if(result == rename)
184         ui->animList->editItem(item);
```

```
185     else if(result == duplicate)
186         duplicateAnim(current);
187     else if(result == del)
188         on_deleteButton_clicked();
189 }
```

Here is the call graph for this function:



12.24.3.8 void KbAnimWidget::on_animList_itemChanged (QListWidgetItem * *item*) [private], [slot]

Definition at line 158 of file kbanimwidget.cpp.

References animations, current, KbAnim::name(), Ui_KbAnimWidget::nameBox, noReorder, and ui.

```
158                                                                 {
159     if(item){
160         KbAnim* anim = animations[item->data(Qt::UserRole).toUuid()];
161         if(anim){
162             anim->name(item->text().trimmed());
163             if(anim == current && !noReorder)
164                 ui->nameBox->setText(anim->name());
165         }
166     }
167 }
```

Here is the call graph for this function:

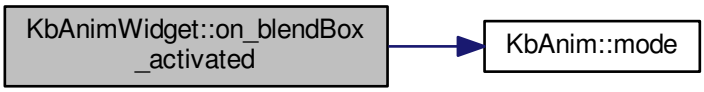
12.24.3.9 void KbAnimWidget::on_blendBox_activated (int *index*) [private], [slot]

Definition at line 205 of file kbanimwidget.cpp.

References current, and KbAnim::mode().

```
205                                                 {
206     if(current)
207         current->mode((KbAnim::Mode)index);
208 }
```

Here is the call graph for this function:



12.24.3.10 void KbAnimWidget::on_deleteButton_clicked ( ) [private], [slot]

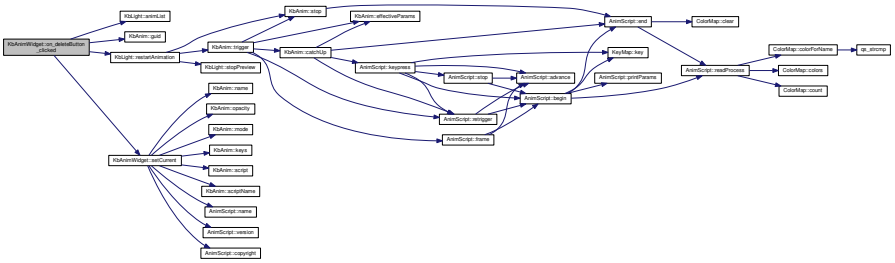
Definition at line 226 of file kbanimwidget.cpp.

References animations, KbLight::animList(), Ui_KbAnimWidget::animList, current, KbAnim::guid(), light, Ui_KbAnimWidget::noAnimLabel, KbLight::restartAnimation(), setCurrent(), and ui.

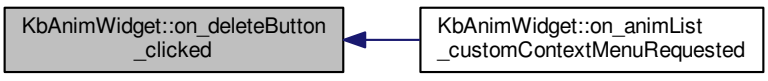
Referenced by on_animList_customContextMenuRequested().

```
226                                          {
227     if(current){
228         animations.remove(current->guid());
229         QList<KbAnim*> animList = light->animList();
230         animList.removeAll(current);
231         light->animList(animList);
232         current->deleteLater();
233         setCurrent(0);
234         delete ui->animList->currentItem();
235         if(animations.count() == 0){
236             ui->animList->setVisible(false);
237             ui->noAnimLabel->setVisible(true);
238         }
239     }
240     light->restartAnimation();
241 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



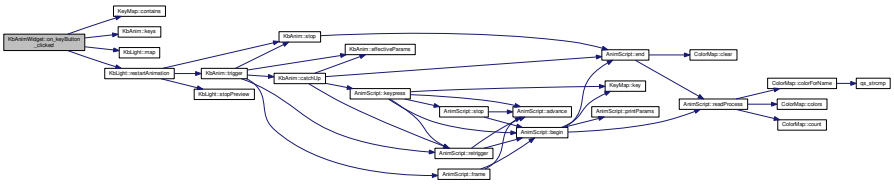
12.24.3.11 void KbAnimWidget::on_keyButton_clicked ( ) [private], [slot]

Definition at line 210 of file kbanimwidget.cpp.

References KeyMap::contains(), current, didUpdateSelection(), KbAnim::keys(), light, KbLight::map(), KbLight::restartAnimation(), and selectedKeys.

```
210                                         {
211     if(current){
212         QStringList keys = selectedKeys;
213         // If any keys were selected previously that aren't in the keymap now, leave them in
214         // This is important for layout compatibility - e.g. being able to select both bslash_iso and
        bslash, because no single layout contains both
215         const KeyMap& map = light->map();
216         foreach(const QString& key, current->keys()){
217             if(!map.contains(key))
218                 keys << key;
219         }
220         current->keys(keys);
221         emit didUpdateSelection(selectedKeys);
222     }
223     light->restartAnimation();
224 }
```

Here is the call graph for this function:



12.24.3.12 void KbAnimWidget::on_nameBox_textEdited (const QString & *arg1*) [private], [slot]

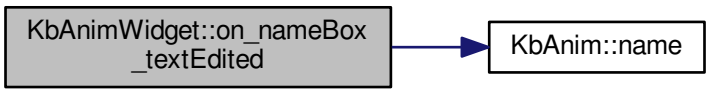
Definition at line 191 of file kbanimwidget.cpp.

References Ui_KbAnimWidget::animList, current, KbAnim::name(), noReorder, and ui.

Referenced by on_propertyButton_clicked().

```
191                                                     {
192     if(current){
193         noReorder = true;
194         current->name(arg1.trimmed());
195         ui->animList->currentItem()->setText(current->name());
196         noReorder = false;
197     }
198 }
```

Here is the call graph for this function:



Here is the caller graph for this function:

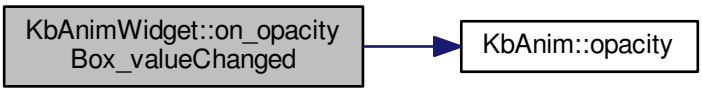
12.24.3.13 void KbAnimWidget::on_opacityBox_valueChanged (double *arg1*) [private], [slot]

Definition at line 200 of file kbanimwidget.cpp.

References current, and KbAnim::opacity().

```
200                                                     {
201     if(current)
202         current->opacity(arg1 / 100.);
203 }
```

Here is the call graph for this function:



12.24.3.14 void KbAnimWidget::on_propertyButton_clicked ( ) [private], [slot]

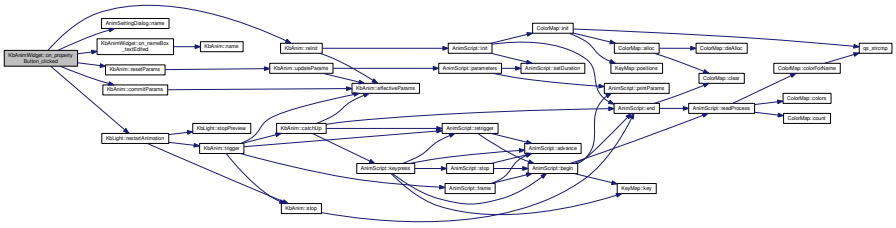
Definition at line 243 of file kbanimwidget.cpp.

References KbAnim::commitParams(), current, light, AnimSettingDialog::name(), Ui_KbAnimWidget::nameBox, on_nameBox_textEdited(), KbAnim::reInit(), KbAnim::resetParams(), KbLight::restartAnimation(), and ui.

Referenced by addAnim().

```
243                                         {
244     if(!current)
245         return;
246     // Present animation property popup
247     AnimSettingDialog dialog(this, current);
248     dialog.exec();
249     if(dialog.result() != QDialog::Accepted){
250         current->resetParams();
251         return;
252     }
253     // Apply settings and restart all animations
254     current->commitParams();
255     current->reInit();
256     light->restartAnimation();
257     // Update name
258     ui->nameBox->setText(dialog.name());
259     on_nameBox_textEdited(dialog.name());
260 }
```

Here is the call graph for this function:



Here is the caller graph for this function:

12.24.3.15 void KbAnimWidget::refreshList ( ) [private], [slot]

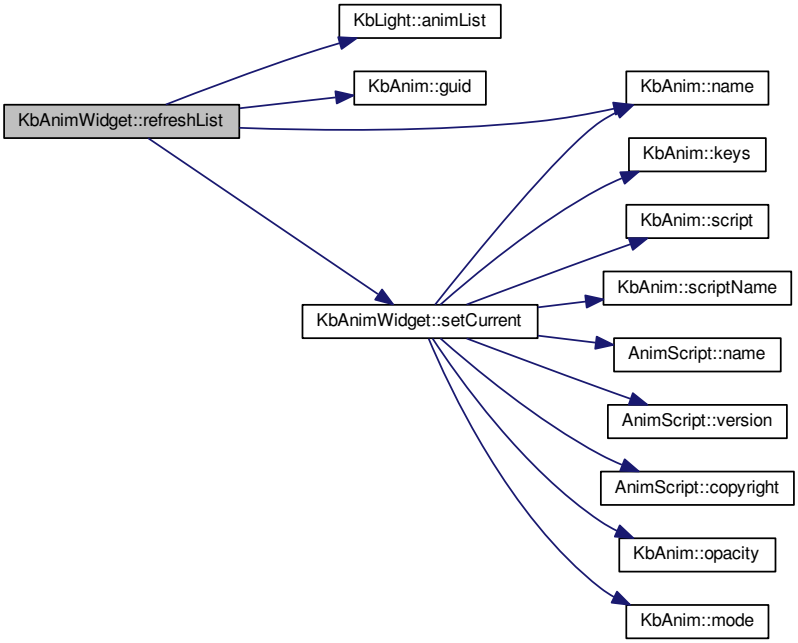
Definition at line 32 of file kbanimwidget.cpp.

References animations, KbLight::animList(), Ui_KbAnimWidget::animList, KbAnim::guid(), light, KbAnim::name(), Ui_KbAnimWidget::noAnimLabel, noReorder, setCurrent(), and ui.

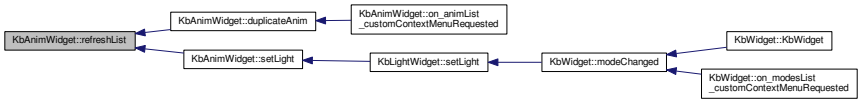
Referenced by duplicateAnim(), and setLight().

```
32                                {
33     noReorder = true;
34     setCurrent(0);
35     ui->animList->clear();
36     animations.clear();
37     // Add the animations from the new lighting mode
38     if(!light){
39         ui->animList->setVisible(false);
40         ui->noAnimLabel->setVisible(true);
41         return;
42     }
43     QList<KbAnim*> newAnimations = light->animList();
44     if(newAnimations.count() == 0){
45         ui->animList->setVisible(false);
46         ui->noAnimLabel->setVisible(true);
47         return;
48     }
49     ui->animList->setVisible(true);
50     foreach(KbAnim* anim, newAnimations){
51         QListWidgetItem* item = new QListWidgetItem(anim->name(), ui->
       animList);
52         item->setData(Qt::UserRole, anim->guid());
53         item->setFlags(item->flags() | Qt::ItemIsEditable);
54         animations[anim->guid()] = anim;
55         ui->animList->addItem(item);
56     }
57     ui->noAnimLabel->setVisible(false);
58     noReorder = false;
59 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.24.3.16 void KbAnimWidget::reorderAnims ( ) [private], [slot]

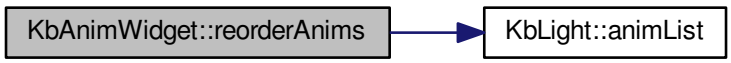
Definition at line 61 of file kbanimwidget.cpp.

References animations, KbLight::animList(), Ui_KbAnimWidget::animList, light, noReorder, and ui.

Referenced by KbAnimWidget().

```
61                                {
62     if(light && !noReorder){
63         // Clear and rebuild the list of animations in case the animation moved
64         int count = ui->animList->count();
65         QList<KbAnim*> animList;
66         for(int i = 0; i < count; i++){
67             QListWidgetItem* item = ui->animList->item(i);
68             KbAnim* anim = animations[item->data(Qt::UserRole).toUuid()];
69             if(anim && !animList.contains(anim))
70                 animList.append(anim);
71             item->setFlags(item->flags() | Qt::ItemIsEditable);
72         }
73         light->animList(animList);
74     }
75 }
```

Here is the call graph for this function:



Here is the caller graph for this function:

12.24.3.17 void KbAnimWidget::setCurrent (KbAnim * *newCurrent*) [private]

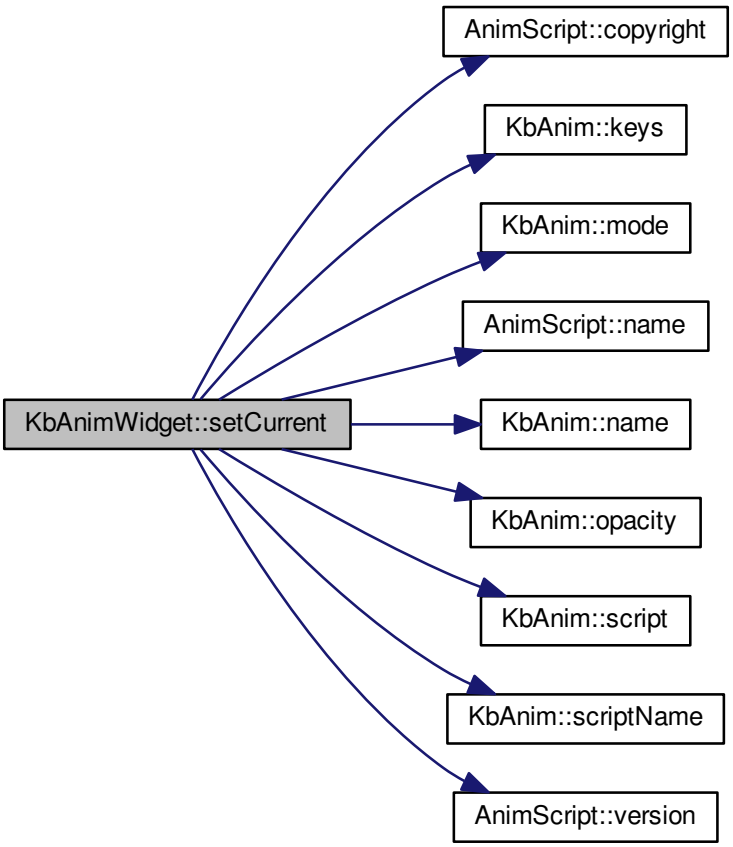
Definition at line 118 of file kbanimwidget.cpp.

References Ui_KbAnimWidget::aCopyLabel, Ui_KbAnimWidget::aMissingLabel, Ui_KbAnimWidget::aNameLabel, animChanged(), Ui_KbAnimWidget::aVerLabel, Ui_KbAnimWidget::blendBox, AnimScript::copyright(), current, KbAnim::keys(), KbAnim::mode(), AnimScript::name(), KbAnim::name(), Ui_KbAnimWidget::nameBox, KbAnim::opacity(), Ui_KbAnimWidget::opacityBox, KbAnim::script(), KbAnim::scriptName(), selectedKeys, Ui_KbAnimWidget::selectionStack, ui, and AnimScript::version().

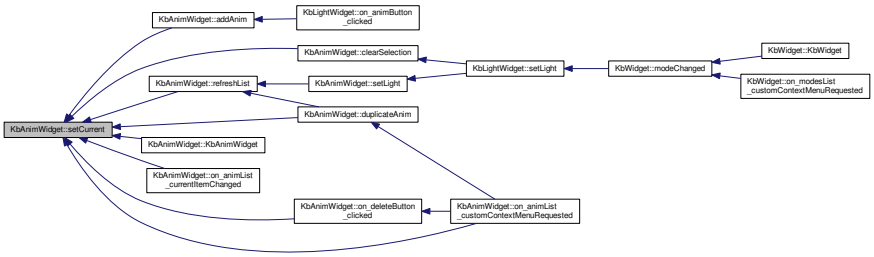
Referenced by addAnim(), clearSelection(), duplicateAnim(), KbAnimWidget(), on_animList_currentItemChanged(), on_animList_customContextMenuRequested(), on_deleteButton_clicked(), and refreshList().

```
118                                                 {
119     if(newCurrent != current)
120         emit animChanged(current = newCurrent);
121     if(!current){
122         selectedKeys.clear();
123         ui->selectionStack->setCurrentIndex(0);
124         return;
125     }
126     selectedKeys = current->keys();
127     const AnimScript* script = current->script();
128     if(!script){
129         ui->selectionStack->setCurrentIndex(2);
130         ui->aMissingLabel->setText("The \"" + current->
      scriptName() + "\" script could not be loaded. Please check your animation directory.");
131         return;
132     }
133     ui->selectionStack->setCurrentIndex(1);
134     ui->aNameLabel->setText(script->name());
135     ui->aVerLabel->setText("v" + script->version());
136     ui->aCopyLabel->setText(script->copyright());
137
138     ui->nameBox->setText(current->name());
139     ui->opacityBox->setValue(current->opacity() * 100.);
140     ui->blendBox->setCurrentIndex((int)current->mode());
141 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.24.3.18 void KbAnimWidget::setLight (KbLight * *newLight*)

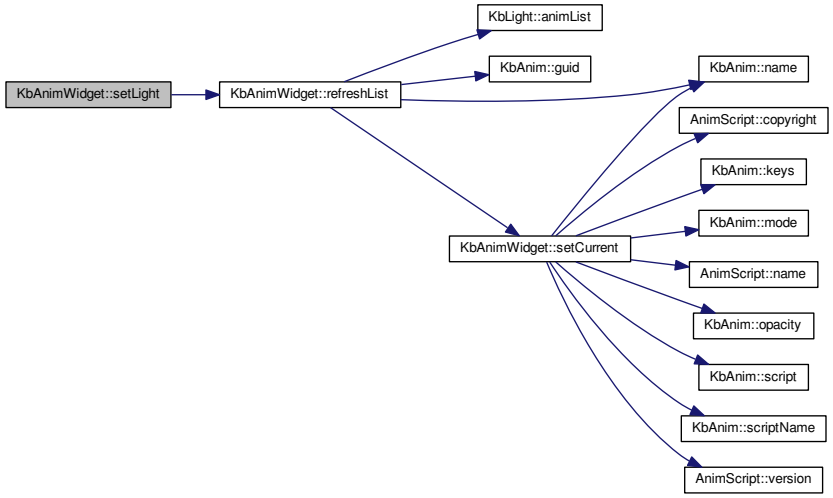
Definition at line 21 of file kbanimwidget.cpp.

References light, and refreshList().

Referenced by KbLightWidget::setLight().

```
21                                          {
22     if(light != newLight){
23         if(light)
24             disconnect(light, SIGNAL(didLoad()), this, SLOT(refreshList()));
25         if(newLight)
26             connect(newLight, SIGNAL(didLoad()), this, SLOT(refreshList()));
27         light = newLight;
28     }
29     refreshList();
30 }
```

Here is the call graph for this function:



Here is the caller graph for this function:

12.24.3.19 void KbAnimWidget::setSelectedKeys (const QStringList & *keys*)

Definition at line 143 of file kbanimwidget.cpp.

References Ui_KbAnimWidget::keyButton, selectedKeys, and ui.

Referenced by KbLightWidget::newSelection().

```
143                                                     {
144     selectedKeys = keys;
145     if(keys.count() == 0)
146         ui->keyButton->setVisible(false);
147     else
148         ui->keyButton->setVisible(true);
149 }
```

Here is the caller graph for this function:

12.24.4 Field Documentation

12.24.4.1 QHash<QUuid, KbAnim*> KbAnimWidget::animations [private]

Definition at line 53 of file kbanimwidget.h.

Referenced by addAnim(), on_animList_currentItemChanged(), on_animList_customContextMenuRequested(), on_animList_itemChanged(), on_deleteButton_clicked(), refreshList(), and reorderAnims().

12.24.4.2 KbAnim* KbAnimWidget::current [private]

Definition at line 55 of file kbanimwidget.h.

Referenced by on_animList_customContextMenuRequested(), on_animList_itemChanged(), on_blendBox_activated(), on_deleteButton_clicked(), on_keyButton_clicked(), on_nameBox_textEdited(), on_opacityBox_valueChanged(), on_propertyButton_clicked(), and setCurrent().

12.24.4.3 KbLight* KbAnimWidget::light [private]

Definition at line 52 of file kbanimwidget.h.

Referenced by addAnim(), duplicateAnim(), on_deleteButton_clicked(), on_keyButton_clicked(), on_propertyButton_clicked(), refreshList(), reorderAnims(), and setLight().

12.24.4.4 bool KbAnimWidget::noReorder [private]

Definition at line 58 of file kbanimwidget.h.

Referenced by addAnim(), duplicateAnim(), on_animList_itemChanged(), on_nameBox_textEdited(), refreshList(), and reorderAnims().

12.24.4.5 QStringList KbAnimWidget::selectedKeys [private]

Definition at line 57 of file kbanimwidget.h.

Referenced by on_keyButton_clicked(), setCurrent(), and setSelectedKeys().

12.24.4.6 Ui::KbAnimWidget* KbAnimWidget::ui [private]

Definition at line 60 of file kbanimwidget.h.

Referenced by addAnim(), clearSelection(), duplicateAnim(), KbAnimWidget(), on_animList_customContextMenuRequested(), on_animList_itemChanged(), on_deleteButton_clicked(), on_nameBox_textEdited(), on_propertyButton_clicked(), refreshList(), reorderAnims(), setCurrent(), setSelectedKeys(), and ~KbAnimWidget().

The documentation for this class was generated from the following files:

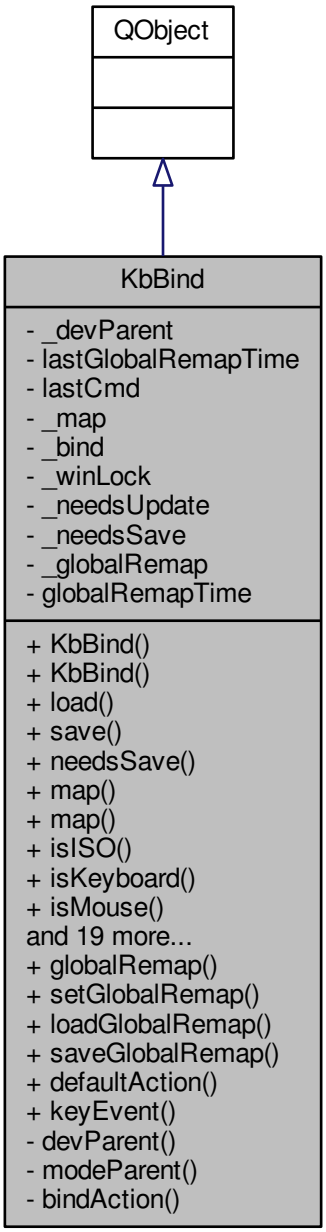
- src/ckb/kbanimwidget.h

- src/ckb/kbanimwidget.cpp
- src/ckb/release/moc_kbanimwidget.cpp

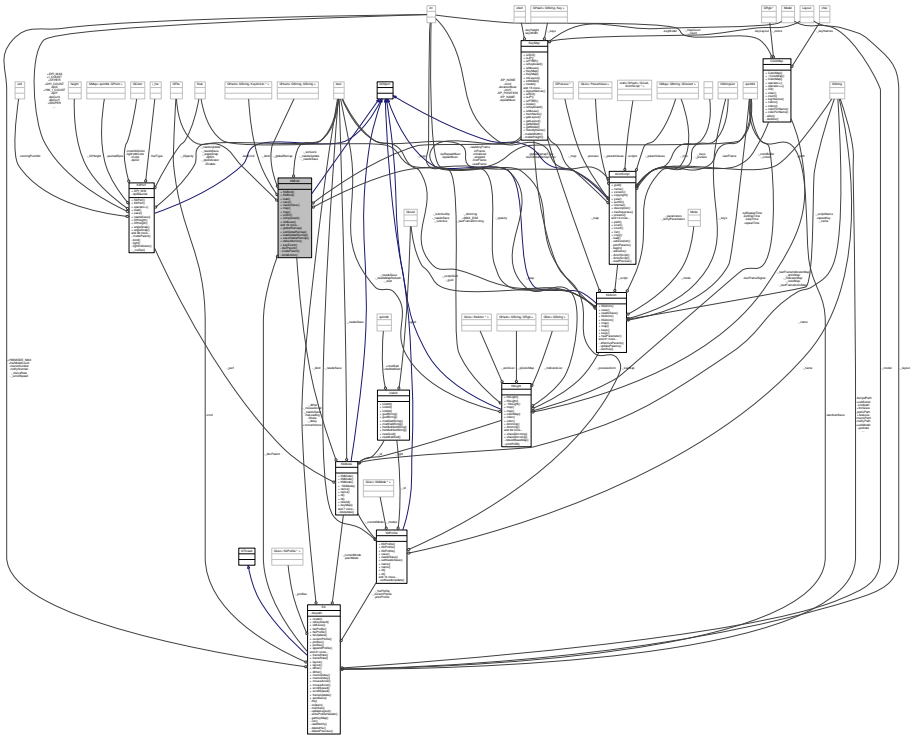
12.25 KbBind Class Reference

`#include <src/ckb/kbbind.h>`

Inheritance diagram for KbBind:



Collaboration diagram for KbBind:



Public Slots

- void keyEvent (const QString &key, bool down)

Signals

- void didLoad ()
- void layoutChanged ()
- void updated ()

Public Member Functions

- KbBind (KbMode *modeParent, Kb *devParent, const KeyMap &keyMap)
- KbBind (KbMode *modeParent, Kb *devParent, const KeyMap &keyMap, const KbBind &other)

  KbBind.
- void load (CkbSettings &settings)
- void save (CkbSettings &settings)
- bool needsSave () const
- const KeyMap & map ()
- void map (const KeyMap &map)
- bool isISO () const
- bool isKeyboard () const
- bool isMouse () const
- KbPerf * perf ()
- KbLight * light ()
- QString action (const QString &key)
- QString defaultAction (const QString &key)

- QString friendlyName (const QString &key)
- QString friendlyActionName (const QString &key)
- void setAction (const QString &key, const QString &action)
- void setAction (const QStringList &keys, const QString &action)
- void resetAction (const QString &key)
- void resetAction (const QStringList &keys)
- void noAction (const QString &key)
- void noAction (const QStringList &keys)
- bool winLock ()
- void winLock (bool newWinLock)
- void update (QFile &cmd, bool force=false)
- void setNeedsUpdate ()
- int getMacroNumber ()

  KbBind::getMacroNumber.
- QString getMacroPath ()

  KbBind::getMacroPath.
- void handleNotificationChannel (bool start)

  handleNotificationChannel sends commands to ckb-daemon for (de-) activating the notify channel. Send a notify cmd to the keyboard to set or clear notification for reading macro definition. The file handle for the cmd pipe is stored in lastCmd.

Static Public Member Functions

- static QString globalRemap (const QString &key)
- static void setGlobalRemap (const QHash< QString, QString > keyToActual)
- static void loadGlobalRemap ()
- static void saveGlobalRemap ()
- static QString defaultAction (const QString &key, KeyMap::Model model)

Private Member Functions

- Kb * devParent () const
- KbMode * modeParent () const
- KeyAction * bindAction (const QString &key)

Private Attributes

- Kb * _devParent
- quint64 lastGlobalRemapTime
- QFile * lastCmd

  lastCmd is a cache-hack. Because the QFile ist opened in Kb, and we need it in the macro processing functions, we cache the value her in lastCmd.
- KeyMap _map
- QHash< QString, KeyAction * > _bind
- bool _winLock
- bool _needsUpdate
- bool _needsSave

Static Private Attributes

- static QHash< QString, QString > _globalRemap
- static quint64 globalRemapTime = 0

Friends

- class KeyAction

12.25.1 Detailed Description

Definition at line 20 of file kbbind.h.

12.25.2 Constructor & Destructor Documentation

12.25.2.1 KbBind::KbBind (KbMode * *modeParent*, Kb * *devParent*, const KeyMap & *keyMap*) [explicit]

Definition at line 12 of file kbbind.cpp.

```
12                                                                          :
13     QObject(modeParent), _devParent(parentBoard),
       lastGlobalRemapTime(globalRemapTime), _map(keyMap),
14     _winLock(false), _needsUpdate(true), _needsSave(true) {
15 }
```

12.25.2.2 KbBind::KbBind (KbMode * *modeParent*, Kb * *parentBoard*, const KeyMap & *keyMap*, const KbBind & *other*)

KbBind::KbBind // copy all existing Key bindings.

Parameters

modeParent	
devParent	
keyMap	
other	This is the KbBind object to copy from Use this constructor to copy an existing Binding
modeParent	
parentBoard	
keyMap	
other	

Create a new Hash table and copy all entries

clear the destination list (there are the original KeyActions as references, so do not delete them)

and move the KeyActions we just created

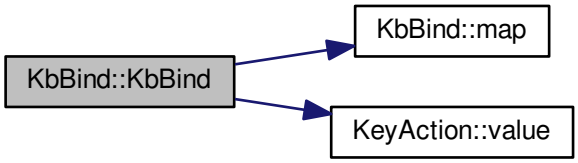
Definition at line 24 of file kbbind.cpp.

References _bind, KeyAction, map(), and KeyAction::value().

```
24                                                                                  :
25     QObject(modeParent), _devParent(parentBoard),
       lastGlobalRemapTime(globalRemapTime), _bind(other.
       _bind),
26     _winLock(false), _needsUpdate(true), _needsSave(true) {
27     map(keyMap);
28
30     QHash<QString, KeyAction*> newBind;
31     foreach(QString key, _bind.keys()) {
32         KeyAction* act = _bind.value(key);
33         if(act) {
34             newBind[key] = new KeyAction(act->value(), this);
35         }
36     }
37
39     _bind.clear();
40     foreach(QString key, newBind.keys()) {
41         KeyAction* act = newBind.value(key);
42         if(act) {
44             _bind[key] = new KeyAction(act->value(), this);
45         }
46     }
```

```
47      newBind.clear();        // here we *must not* delete the KeyActions, because they are referenced by _bind
        now
48 }
```

Here is the call graph for this function:



12.25.3 Member Function Documentation

12.25.3.1 QString KbBind::action (const QString & *key*)

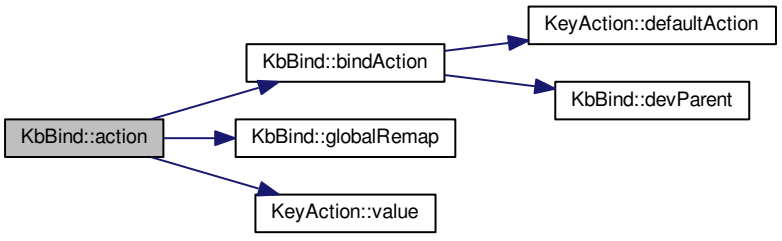
Definition at line 145 of file kbbind.cpp.

References bindAction(), globalRemap(), and KeyAction::value().

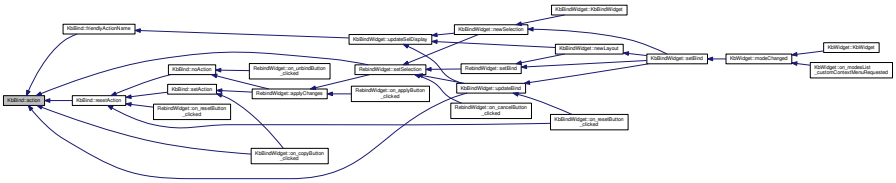
Referenced by friendlyActionName(), KbBindWidget::on_copyButton_clicked(), resetAction(), RebindWidget::setSelection(), and KbBindWidget::updateBind().

```
145                                         {
146      QString rKey = globalRemap(key);
147      return bindAction(rKey)->value();
148 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.25.3.2 KeyAction * KbBind::bindAction (const QString & *key*) [private]

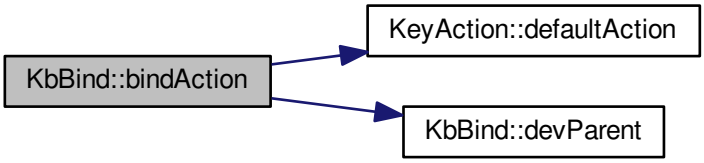
Definition at line 140 of file kbbind.cpp.

References _bind, KeyAction::defaultAction(), devParent(), and KeyAction.

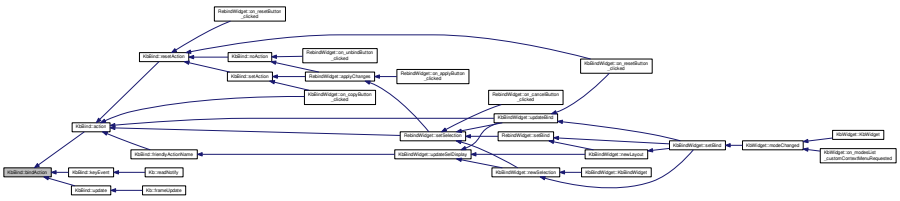
Referenced by action(), keyEvent(), and update().

```
140                                                 {
141     if(!_bind.contains(key)) return _bind[key] = new KeyAction(
        KeyAction::defaultAction(key, devParent()->model()), this);
142     return _bind[key];
143 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.25.3.3 QString KbBind::defaultAction (const QString & *key*)

Definition at line 155 of file kbbind.cpp.

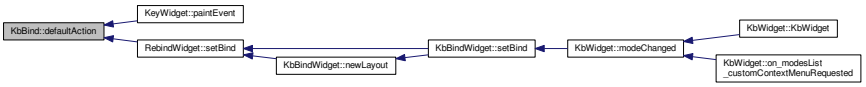
References devParent().

Referenced by KeyWidget::paintEvent(), and RebindWidget::setBind().

```
155                                                 {
156     return defaultAction(key, devParent()->model());
157 }
```

Here is the call graph for this function:

Here is the caller graph for this function:



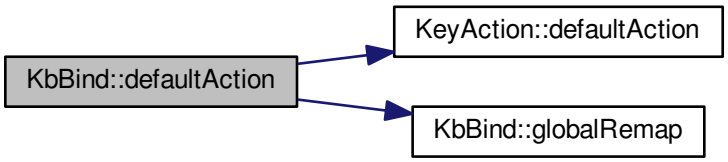
12.25.3.4 QString KbBind::defaultAction (const QString & *key*, KeyMap::Model *model*) [static]

Definition at line 150 of file kbbind.cpp.

References KeyAction::defaultAction(), and globalRemap().

```
150                                                                 {
151     QString rKey = globalRemap(key);
152     return KeyAction::defaultAction(rKey, model);
153 }
```

Here is the call graph for this function:



12.25.3.5 Kb* KbBind::devParent ( ) const [inline], [private]

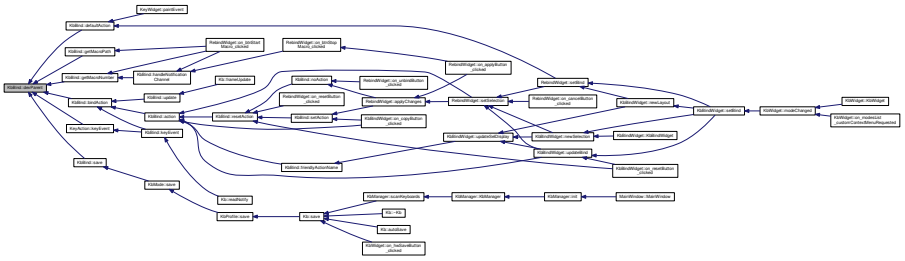
Definition at line 115 of file kbbind.h.

References _devParent.

Referenced by bindAction(), defaultAction(), getMacroNumber(), getMacroPath(), KeyAction::keyEvent(), and save().

```
115 { return _devParent; }
```

Here is the caller graph for this function:



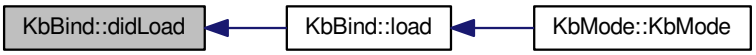
12.25.3.6 void KbBind::didLoad ( ) [signal]

Definition at line 155 of file moc_kbbind.cpp.

Referenced by load().

```
156 {
157     QMetaObject::activate(this, &staticMetaObject, 0, Q_NULLPTR);
158 }
```

Here is the caller graph for this function:



12.25.3.7 QString KbBind::friendlyActionName (const QString & *key*)

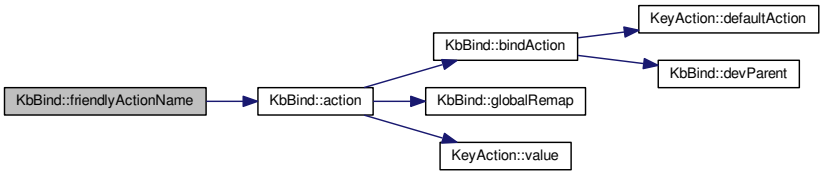
Definition at line 166 of file kbbind.cpp.

References _map, action(), and KeyAction.

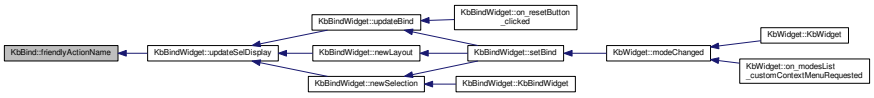
Referenced by KbBindWidget::updateSelDisplay().

```
166                                                 {
167     QString act = action(key);
168     return KeyAction(act).friendlyName(_map);
169 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



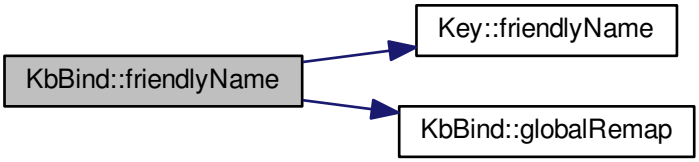
12.25.3.8 QString KbBind::friendlyName (const QString & *key*)

Definition at line 159 of file kbbind.cpp.

References _map, Key::friendlyName(), and globalRemap().

```
159                                                 {
160     const Key& pos = _map[globalRemap(key)];
161     if(!pos)
162         return "(Unknown)";
163     return pos.friendlyName();
164 }
```

Here is the call graph for this function:



12.25.3.9 int KbBind::getMacroNumber ( )

Returns

number of notification channel. Use it in combination with notifyon/off-Statement

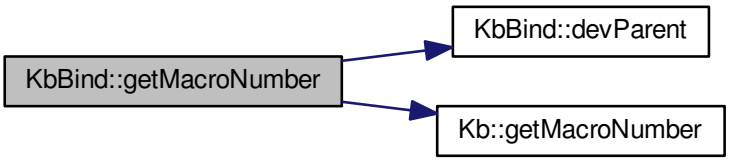
Definition at line 266 of file kbbind.cpp.

References devParent(), and Kb::getMacroNumber().

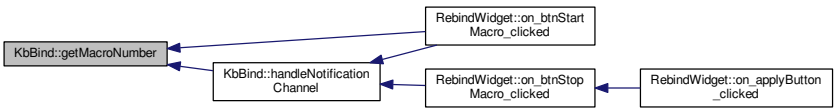
Referenced by handleNotificationChannel(), and RebindWidget::on_btnStartMacro_clicked().

```
266                             {
267     return devParent()->getMacroNumber();
268 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.25.3.10 QString KbBind::getMacroPath ( )

Returns

Filepath of macro notification pipe. If not set, returns initial value ""

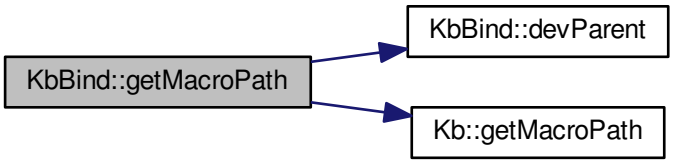
Definition at line 274 of file kbbind.cpp.

References devParent(), and Kb::getMacroPath().

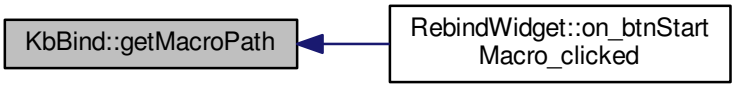
Referenced by RebindWidget::on_btnStartMacro_clicked().

```
274                             {
275     return devParent()->getMacroPath();
276 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.25.3.11 QString KbBind::globalRemap (const QString & *key*) [static]

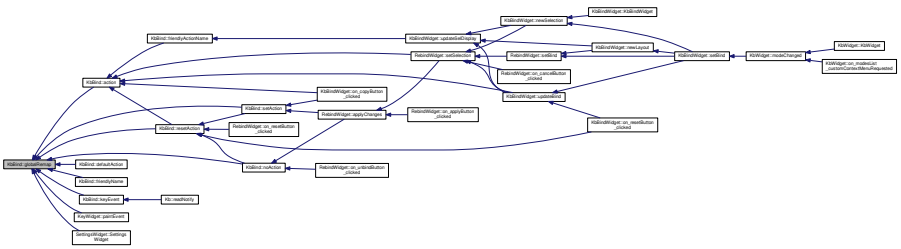
Definition at line 98 of file kbbind.cpp.

References _globalRemap.

Referenced by action(), defaultAction(), friendlyName(), keyEvent(), noAction(), KeyWidget::paintEvent(), resetAction(), setAction(), and SettingsWidget::SettingsWidget().

```
98                                                 {
99     if(!_globalRemap.contains(key))
100         return key;
101     return _globalRemap.value(key);
102 }
```

Here is the caller graph for this function:



12.25.3.12 void KbBind::handleNotificationChannel (bool *start*)

Parameters

start	is boolean. If true, notification channel is opened for all keys, otherwise channel ist closed.
start	If true, notification channel is opened for all keys, otherwise channel ist closed.

Definition at line 284 of file kbbind.cpp.

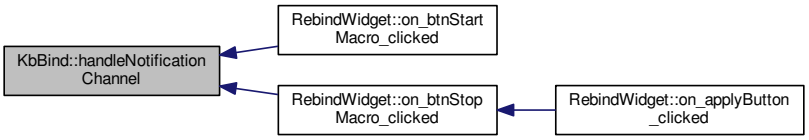
References getMacroNumber(), and lastCmd.

Referenced by RebindWidget::on_btnStartMacro_clicked(), and RebindWidget::on_btnStopMacro_clicked().

```
284                                                 {
285     if (getMacroNumber() > 0 && lastCmd) {
286         if (start) {
287             lastCmd->write (QString("\nnotifyon %1\n@%1 notify all:on\n").arg(
    getMacroNumber()).toLatin1());
288         } else {
289             lastCmd->write (QString("\n@%1 notify all:off\nnotifyoff %1\n").arg(
    getMacroNumber()).toLatin1());
290         }
291         lastCmd->flush();
292     } else qDebug() << QString("No cmd or valid handle for notification found, macroNumber = %1, lastCmd =
     %2")
293                        .arg(getMacroNumber()).arg(lastCmd? "set" : "unset");
294 }
```

Here is the call graph for this function:

Here is the caller graph for this function:



12.25.3.13 bool KbBind::isISO ( ) const [inline]

Definition at line 43 of file kbbind.h.

References _map, and KeyMap::isISO().

```
43 { return _map.isISO(); }
```

Here is the call graph for this function:

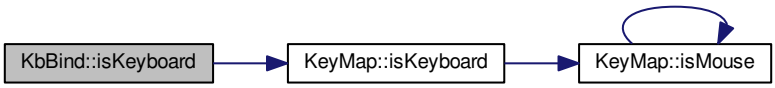
12.25.3.14 bool KbBind::isKeyboard ( ) const [inline]

Definition at line 44 of file kbbind.h.

References _map, and KeyMap::isKeyboard().

```
44 { return _map.isKeyboard(); }
```

Here is the call graph for this function:



12.25.3.15 bool KbBind::isMouse ( ) const [inline]

Definition at line 45 of file kbbind.h.

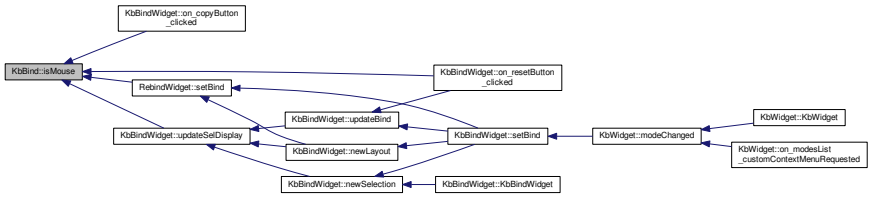
References _map, and KeyMap::isMouse().

Referenced by KbBindWidget::on_copyButton_clicked(), KbBindWidget::on_resetButton_clicked(), RebindWidget::setBind(), and KbBindWidget::updateSelDisplay().

```
45 { return _map.isMouse(); }
```

Here is the call graph for this function:

Here is the caller graph for this function:



12.25.3.16 void KbBind::keyEvent (const QString & *key*, bool *down*) [slot]

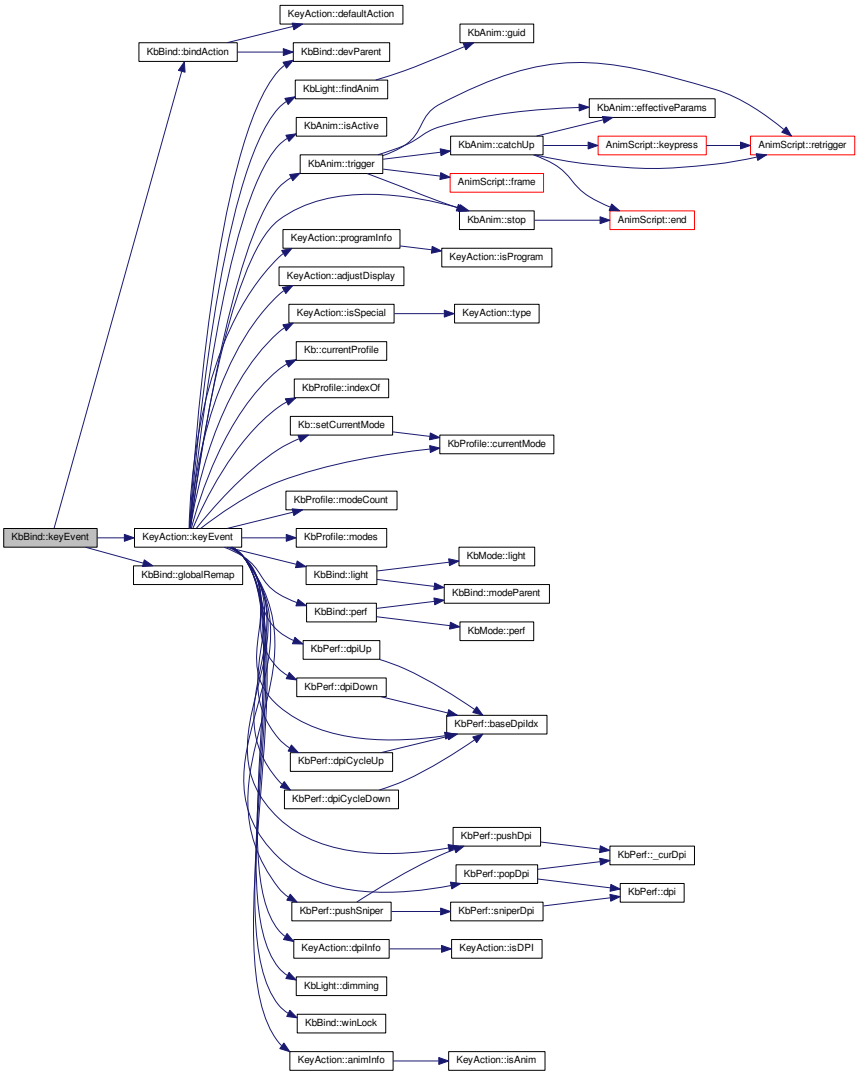
Definition at line 296 of file kbbind.cpp.

References bindAction(), globalRemap(), and KeyAction::keyEvent().

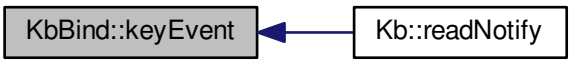
Referenced by Kb::readNotify().

```
296                                                 {
297     QString rKey = globalRemap(key);
298     KeyAction* act = bindAction(rKey);
299     if(act)
300         act->keyEvent(this, down);
301 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



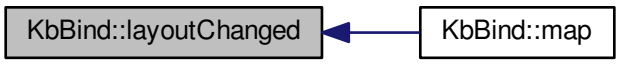
12.25.3.17 void KbBind::layoutChanged ( ) [signal]

Definition at line 161 of file moc_kbbind.cpp.

Referenced by map().

```
162 {
163     QMetaObject::activate(this, &staticMetaObject, 1, Q_NULLPTR);
164 }
```

Here is the caller graph for this function:



12.25.3.18 KbLight * KbBind::light ( )

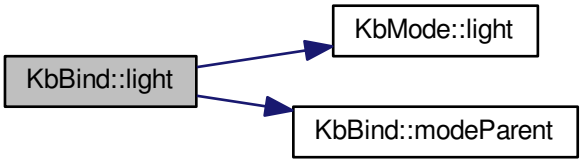
Definition at line 54 of file kbbind.cpp.

References KbMode::light(), and modeParent().

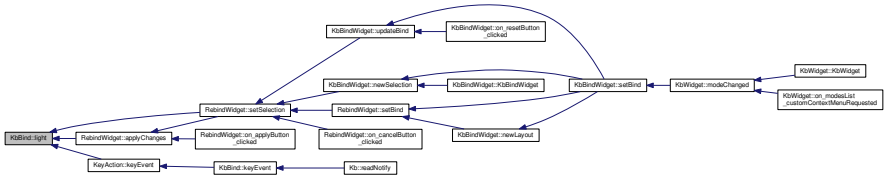
Referenced by RebindWidget::applyChanges(), KeyAction::keyEvent(), and RebindWidget::setSelection().

```
54                       {
55     return modeParent()->light();
56 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.25.3.19 void KbBind::load (CkbSettings & *settings*)

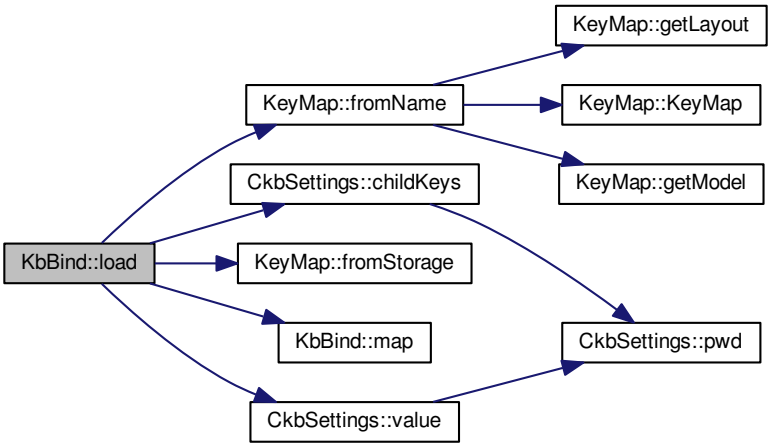
Definition at line 58 of file kbbind.cpp.

References _bind, _map, _needsSave, _winLock, CkbSettings::childKeys(), didLoad(), KeyMap::fromName(), KeyMap::fromStorage(), KeyAction, map(), and CkbSettings::value().

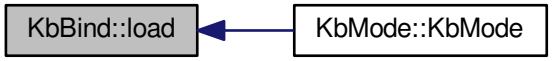
Referenced by KbMode::KbMode().

```
58                                      {
59     _needsSave = false;
60     SGroup group(settings, "Binding");
61     KeyMap currentMap = _map;
62     _map = KeyMap::fromName(settings.value("KeyMap").toString());
63     // Load key settings
64     bool useReal = settings.value("UseRealNames").toBool();
65     _bind.clear();
66     {
67         SGroup group(settings, "Keys");
68         foreach(QString key, settings.childKeys()){
69             QString name = key.toLower();
70             if(!useReal)
71                 name = _map.fromStorage(name);
72             QString bind = settings.value(key).toString();
73             _bind[name] = new KeyAction(bind, this);
74         }
75     }
76     _winLock = settings.value("WinLock").toBool();
77     emit didLoad();
78     map(currentMap);
79 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.25.3.20 void KbBind::loadGlobalRemap ( ) [static]

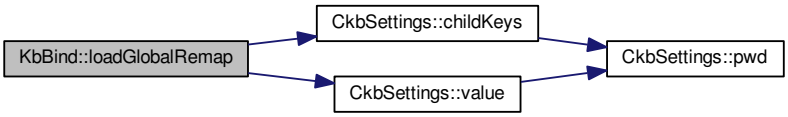
Definition at line 116 of file kbbind.cpp.

References _globalRemap, CkbSettings::childKeys(), globalRemapTime, and CkbSettings::value().

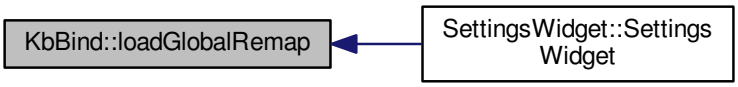
Referenced by SettingsWidget::SettingsWidget().

```
116                             {
117     _globalRemap.clear();
118     CkbSettings settings("Program/GlobalRemap");
119     foreach(const QString& key, settings.childKeys())
120         _globalRemap[key] = settings.value(key).toString();
121     globalRemapTime = QDateTime::currentMSecsSinceEpoch();
122 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.25.3.21 const KeyMap& KbBind::map ( ) [inline]

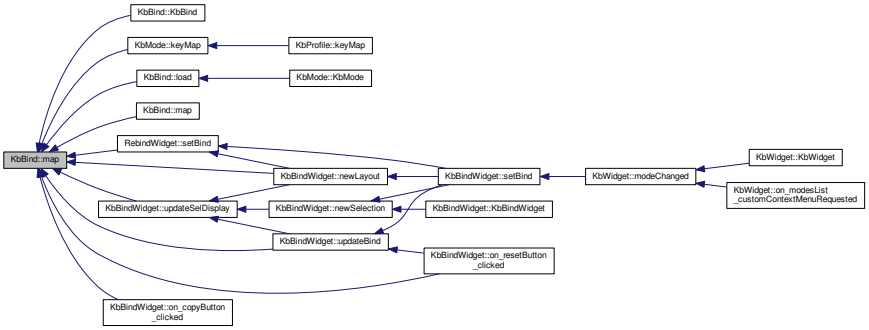
Definition at line 41 of file kbbind.h.

References _map.

Referenced by KbBind(), KbMode::keyMap(), load(), map(), KbBindWidget::newLayout(), KbBindWidget::on_copyButton_clicked(), KbBindWidget::on_resetButton_clicked(), RebindWidget::setBind(), KbBindWidget::updateBind(), and KbBindWidget::updateSelDisplay().

```
41 { return _map; }
```

Here is the caller graph for this function:



12.25.3.22 void KbBind::map (const KeyMap & *map*)

Definition at line 133 of file kbbind.cpp.

References _map, _needsSave, _needsUpdate, layoutChanged(), and map().

```
133                                 {
134     _map = map;
135     _needsUpdate = true;
136     _needsSave = true;
137     emit layoutChanged();
138 }
```

Here is the call graph for this function:

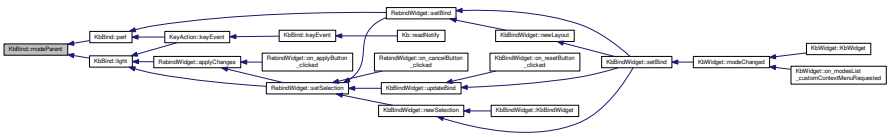
12.25.3.23 KbMode* KbBind::modeParent ( ) const [inline], [private]

Definition at line 116 of file kbbind.h.

Referenced by light(), and perf().

```
116 { return (KbMode*)parent(); }
```

Here is the caller graph for this function:



12.25.3.24 bool KbBind::needsSave ( ) const [inline]

Definition at line 38 of file kbbind.h.

References _needsSave.

Referenced by KbMode::needsSave().

```
38 { return _needsSave; }
```

Here is the caller graph for this function:

12.25.3.25 void KbBind::noAction (const QString & *key*)

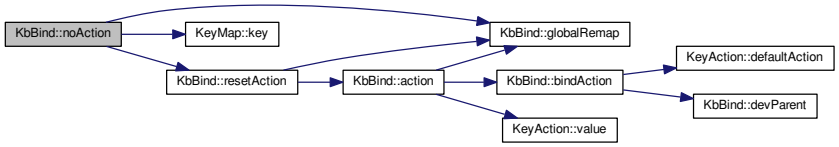
Definition at line 181 of file kbbind.cpp.

References _bind, _map, globalRemap(), KeyMap::key(), KeyAction, and resetAction().

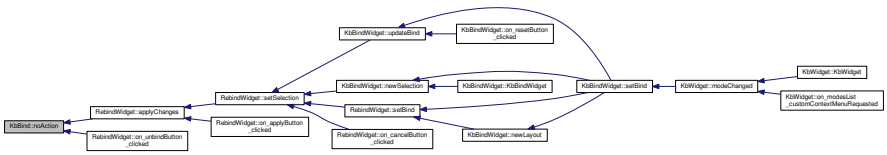
Referenced by RebindWidget::applyChanges(), and RebindWidget::on_unbindButton_clicked().

```
181                                         {
182     resetAction(key);
183     QString rKey = globalRemap(key);
184     if(!_map.key(rKey))
185         return;
186     _bind[rKey] = new KeyAction(this);
187 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.25.3.26 void KbBind::noAction (const QStringList & *keys*) [inline]

Definition at line 76 of file kbbind.h.

References noAction().

Referenced by noAction().

```
76 { foreach(const QString& key, keys) noAction(key); }
```

Here is the call graph for this function:

Here is the caller graph for this function:

12.25.3.27 KbPerf * KbBind::perf ( )

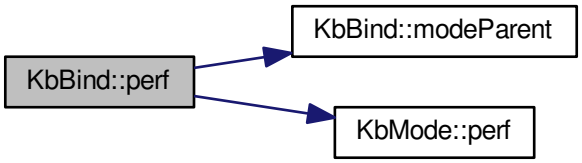
Definition at line 50 of file kbbind.cpp.

References modeParent(), and KbMode::perf().

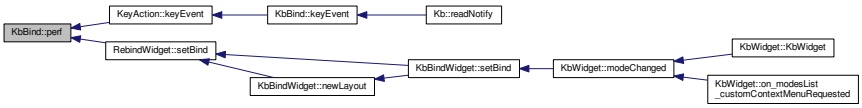
Referenced by KeyAction::keyEvent(), and RebindWidget::setBind().

```
50                     {
51     return modeParent()->perf();
52 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.25.3.28 void KbBind::resetAction (const QString & *key*)

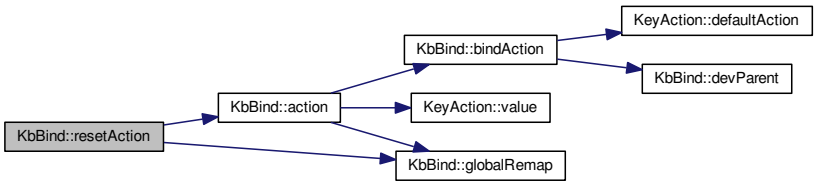
Definition at line 171 of file kbbind.cpp.

References _bind, _needsSave, _needsUpdate, action(), and globalRemap().

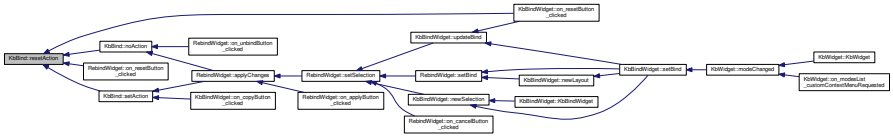
Referenced by noAction(), KbBindWidget::on_resetButton_clicked(), RebindWidget::on_resetButton_clicked(), and setAction().

```
171                                          {
172     QString rKey = globalRemap(key);
173     // Clean up existing action (if any)
174     KeyAction* action = _bind.value(rKey);
175     delete action;
176     _bind.remove(rKey);
177     _needsUpdate = true;
178     _needsSave = true;
179 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.25.3.29 void KbBind::resetAction (const QStringList & *keys*) [inline]

Definition at line 73 of file kbbind.h.

References resetAction().

Referenced by resetAction().

```
73 { foreach(const QString& key, keys) resetAction(key); }
```

Here is the call graph for this function:

Here is the caller graph for this function:

12.25.3.30 void KbBind::save (CkbSettings & *settings*)

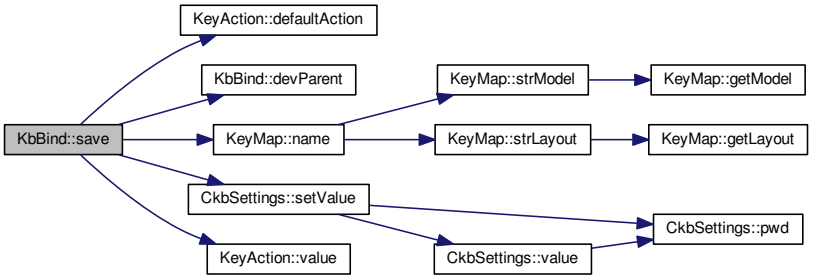
Definition at line 81 of file kbbind.cpp.

References _bind, _map, _needsSave, _winLock, KeyAction::defaultAction(), devParent(), KeyMap::name(), CkbSettings::setValue(), and KeyAction::value().

Referenced by KbMode::save().

```
81                                      {
82     _needsSave = false;
83     SGroup group(settings, "Binding");
84     settings.setValue("KeyMap", _map.name());
85     // Save key settings
86     settings.setValue("UseRealNames", true);
87     {
88         SGroup group(settings, "Keys");
89         foreach(QString key, _bind.keys()){
90             KeyAction* act = _bind.value(key);
91             if(act && act->value() != KeyAction::defaultAction(key,
     devParent()->model()))
92                 settings.setValue(key, act->value());
93         }
94     }
95     settings.setValue("WinLock", _winLock);
96 }
```

Here is the call graph for this function:



Here is the caller graph for this function:

12.25.3.31 void KbBind::saveGlobalRemap ( ) [static]

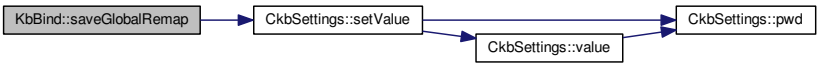
Definition at line 124 of file kbbind.cpp.

References _globalRemap, and CkbSettings::setValue().

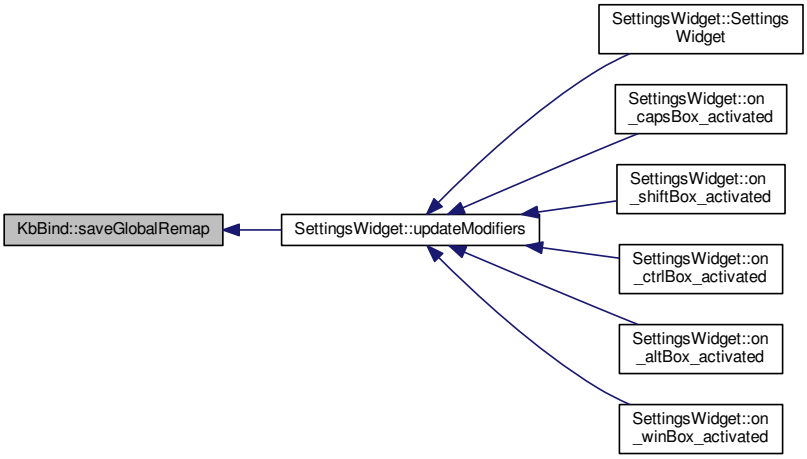
Referenced by SettingsWidget::updateModifiers().

```
124                             {
125     CkbSettings settings("Program/GlobalRemap", true);
126     QHashIterator<QString, QString> i(_globalRemap);
127     while(i.hasNext()){
128         i.next();
129         settings.setValue(i.key(), i.value());
130     }
131 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.25.3.32 void KbBind::setAction (const QString & *key,* const QString & *action*)

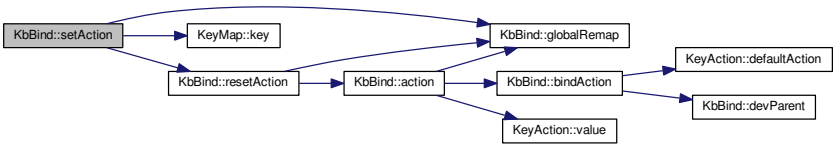
Definition at line 189 of file kbbind.cpp.

References _bind, _map, globalRemap(), KeyMap::key(), KeyAction, and resetAction().

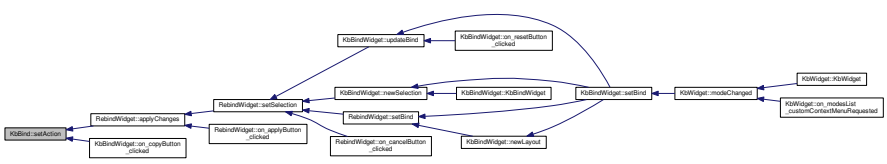
Referenced by RebindWidget::applyChanges(), and KbBindWidget::on_copyButton_clicked().

```
189                                                                 {
190     resetAction(key);
191     QString rKey = globalRemap(key);
192     if(!_map.key(rKey))
193         return;
194     _bind[rKey] = new KeyAction(action, this);
195 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.25.3.33 void KbBind::setAction (const QStringList & *keys*, const QString & *action*) [inline]

Definition at line 70 of file kbbind.h.

References setAction().

Referenced by setAction().

```
70 { foreach(const QString& key, keys) setAction(key, action); }
```

Here is the call graph for this function:

Here is the caller graph for this function:

12.25.3.34 void KbBind::setGlobalRemap (const QHash< QString, QString > *keyToActual*) [static]

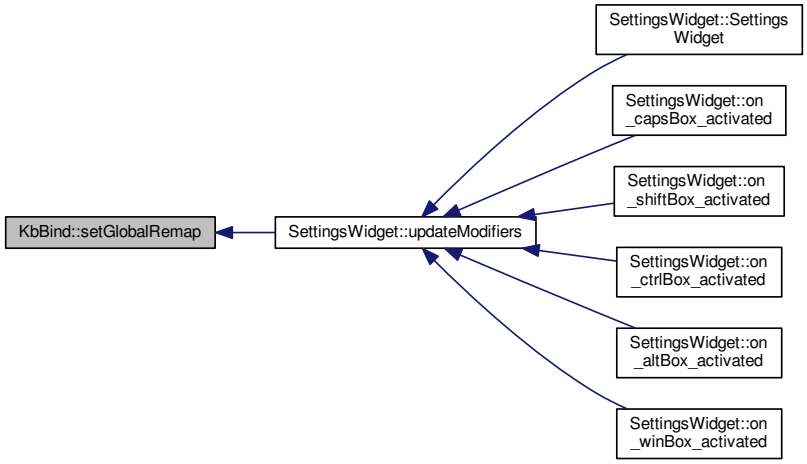
Definition at line 104 of file kbbind.cpp.

References _globalRemap, and globalRemapTime.

Referenced by SettingsWidget::updateModifiers().

```
104                                                                 {
105     _globalRemap.clear();
106     // Ignore any keys with the standard binding
107     QHashIterator<QString, QString> i(keyToActual);
108     while(i.hasNext()){
109         i.next();
110         if(i.key() != i.value())
111             _globalRemap[i.key()] = i.value();
112     }
113     globalRemapTime = QDateTime::currentMSecsSinceEpoch();
114 }
```

Here is the caller graph for this function:



12.25.3.35 void KbBind::setNeedsUpdate ( ) [inline]

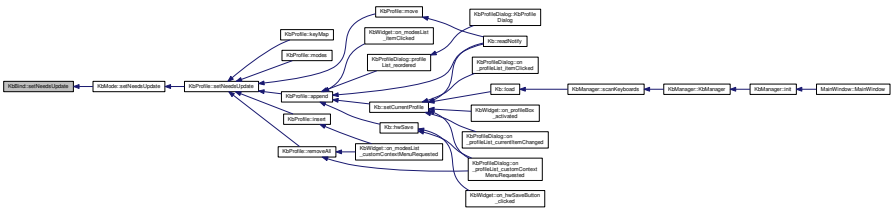
Definition at line 85 of file kbbind.h.

References _needsUpdate.

Referenced by KbMode::setNeedsUpdate().

```
85 { _needsUpdate = true; }
```

Here is the caller graph for this function:



12.25.3.36 void KbBind::update (QFile & *cmd,* bool *force* = false)

Definition at line 197 of file kbbind.cpp.

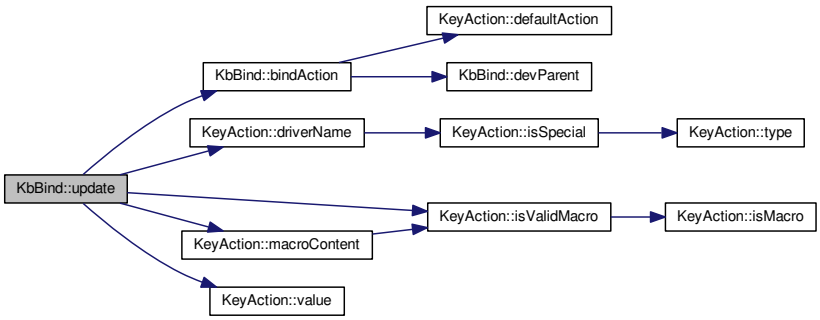
References _bind, _globalRemap, _needsUpdate, _winLock, bindAction(), KeyAction::driverName(), globalRemapTime, KeyAction::isValidMacro(), lastCmd, lastGlobalRemapTime, KeyAction::macroContent(), updated(), and KeyAction::value().

Referenced by Kb::frameUpdate().

```
197                                        {
198     if(!force && !_needsUpdate && lastGlobalRemapTime ==
      globalRemapTime)
199         return;
```

```
200     lastGlobalRemapTime = globalRemapTime;
201     emit updated();
202     _needsUpdate = false;
203     // Reset all keys and enable notifications for all
204     cmd.write("rebind all notify all");
205     // Make sure modifier keys are included as they may be remapped globally
206     QHash<QString, KeyAction*> bind(_bind);
207     if(!_bind.contains("caps")) bind["caps"] = 0;
208     if(!_bind.contains("lshift")) bind["lshift"] = 0;
209     if(!_bind.contains("rshift")) bind["rshift"] = 0;
210     if(!_bind.contains("lctrl")) bind["lctrl"] = 0;
211     if(!_bind.contains("rctrl")) bind["rctrl"] = 0;
212     if(!_bind.contains("lwin")) bind["lwin"] = 0;
213     if(!_bind.contains("rwin")) bind["rwin"] = 0;
214     if(!_bind.contains("lalt")) bind["lalt"] = 0;
215     if(!_bind.contains("ralt")) bind["ralt"] = 0;
216     if(!_bind.contains("fn")) bind["fn"] = 0;
217     QHashIterator<QString, KeyAction*> i(bind);
218
219     // Initialize String buffer for macro Key definitions (G-keys)
220     // "macro clear" is neccessary, if an older definition is unbound.
221     QString macros = "\nmacro clear\n";
222
223     // Write out rebound keys and collect infos for macro definitions
224     while(i.hasNext()){
225         i.next();
226         QString key = i.key();
227         KeyAction* act = i.value();
228         if(_globalRemap.contains(key))
229             act = bindAction(_globalRemap.value(key));
230         if(!act)
231             continue;
232         QString value = act->driverName();
233         if(value.isEmpty()){
234             // If the key is unbound or is a special action, unbind it
235             cmd.write(" unbind ");
236             cmd.write(key.toLatin1());
237             // if a macro definiton for the key is given,
238             // add the converted string to key-buffer "macro"
239             if (act->isValidMacro()) {
240                 if (act->macroContent().length() > 0) {
241                     macros.append("macro " + key.toLatin1() + ":" + act->
    macroContent().toLatin1() + "\n");
242                 }
243             }
244         } else {
245             // Otherwise, write the binding
246             cmd.write(" bind ");
247             cmd.write(key.toLatin1());
248             cmd.write(":");
249             cmd.write(value.toLatin1());
250         }
251     }
252     // If win lock is enabled, unbind windows keys
253     if(_winLock)
254         cmd.write(" unbind lwin rwin");
255
256     // At last, send Macro definitions if available.
257     // If no definitions are made, clear macro will be sent only to reset all macros,
258     cmd.write(macros.toLatin1());
259     lastCmd = &cmd;
260 }
```

Here is the call graph for this function:



Here is the caller graph for this function:

12.25.3.37 void KbBind::updated ( ) [signal]

Definition at line 167 of file moc_kbbind.cpp.

Referenced by update().

```
168 {
169     QMetaObject::activate(this, &staticMetaObject, 2, Q_NULLPTR);
170 }
```

Here is the caller graph for this function:

12.25.3.38 bool KbBind::winLock ( ) [inline]

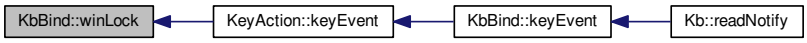
Definition at line 79 of file kbbind.h.

References _winLock.

Referenced by KeyAction::keyEvent().

```
79 { return _winLock; }
```

Here is the caller graph for this function:



12.25.3.39 void KbBind::winLock (bool *newWinLock*) [inline]

Definition at line 80 of file kbbind.h.

References _needsUpdate, and _winLock.

```
80 { _winLock = newWinLock; _needsUpdate = true; }
```

12.25.4 Friends And Related Function Documentation

12.25.4.1 friend class KeyAction [friend]

Definition at line 137 of file kbbind.h.

Referenced by bindAction(), friendlyActionName(), KbBind(), load(), noAction(), and setAction().

12.25.5 Field Documentation

12.25.5.1 QHash<QString, KeyAction*> KbBind::_bind [private]

Definition at line 132 of file kbbind.h.

Referenced by bindAction(), KbBind(), load(), noAction(), resetAction(), save(), setAction(), and update().

12.25.5.2 Kb* KbBind::_devParent [private]

Definition at line 114 of file kbbind.h.

Referenced by devParent().

12.25.5.3 QHash< QString, QString > KbBind::_globalRemap [static], [private]

Definition at line 120 of file kbbind.h.

Referenced by globalRemap(), loadGlobalRemap(), saveGlobalRemap(), setGlobalRemap(), and update().

12.25.5.4 KeyMap KbBind::_map [private]

Definition at line 130 of file kbbind.h.

Referenced by friendlyActionName(), friendlyName(), isISO(), isKeyboard(), isMouse(), load(), map(), noAction(), save(), and setAction().

12.25.5.5 bool KbBind::_needsSave [private]

Definition at line 136 of file kbbind.h.

Referenced by load(), map(), needsSave(), resetAction(), and save().

12.25.5.6 bool KbBind::_needsUpdate [private]

Definition at line 135 of file kbbind.h.

Referenced by map(), resetAction(), setNeedsUpdate(), update(), and winLock().

12.25.5.7 bool KbBind::_winLock [private]

Definition at line 134 of file kbbind.h.

Referenced by load(), save(), update(), and winLock().

12.25.5.8 quint64 KbBind::globalRemapTime = 0 [static], [private]

Definition at line 121 of file kbbind.h.

Referenced by loadGlobalRemap(), setGlobalRemap(), and update().

12.25.5.9 QFile* KbBind::lastCmd [private]

Definition at line 128 of file kbbind.h.

Referenced by handleNotificationChannel(), and update().

12.25.5.10 quint64 KbBind::lastGlobalRemapTime [private]

Definition at line 122 of file kbbind.h.

Referenced by update().

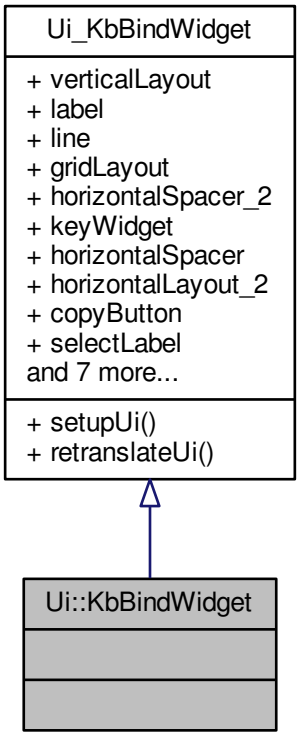
The documentation for this class was generated from the following files:

- src/ckb/kbbind.h
- src/ckb/kbbind.cpp
- src/ckb/release/moc_kbbind.cpp

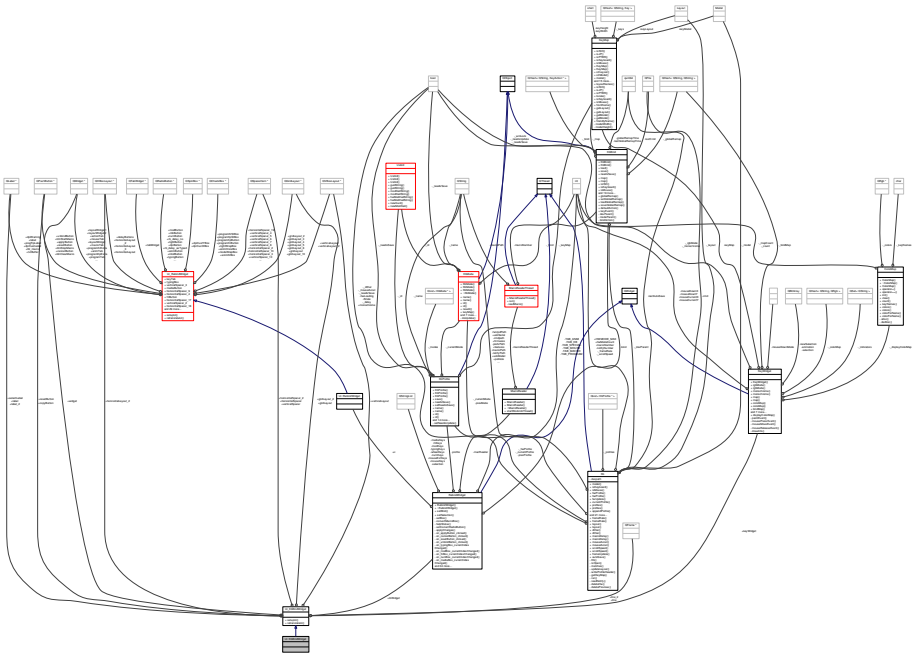
12.26 Ui::KbBindWidget Class Reference

`#include <src/ckb/ui_kbbindwidget.h>`

Inheritance diagram for Ui::KbBindWidget:



Collaboration diagram for Ui::KbBindWidget:



Public Member Functions

- void setupUi (QWidget *KbBindWidget)
- void retranslateUi (QWidget *KbBindWidget)

Data Fields

- QVBoxLayout * verticalLayout
- QLabel * label
- QFrame * line
- QGridLayout * gridLayout
- QSpacerItem * horizontalSpacer_2
- KeyWidget * keyWidget
- QSpacerItem * horizontalSpacer
- QHBoxLayout * horizontalLayout_2
- QPushButton * copyButton
- QLabel * selectLabel
- QPushButton * resetButton
- QSpacerItem * verticalSpacer
- QLabel * label_2
- QFrame * line_2
- QWidget * widget
- QGridLayout * gridLayout_2
- RebindWidget * rbWidget

12.26.1 Detailed Description

Definition at line 202 of file ui_kbbindwidget.h.

12.26.2 Member Function Documentation

12.26.2.1 void Ui_KbBindWidget::retranslateUi (QWidget * *KbBindWidget*) [inline], [inherited]

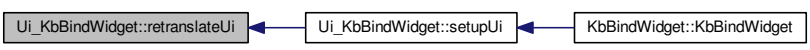
Definition at line 189 of file ui_kbbindwidget.h.

References Ui_KbBindWidget::copyButton, Ui_KbBindWidget::label, Ui_KbBindWidget::label_2, Ui_KbBindWidget::resetButton, and Ui_KbBindWidget::selectLabel.

Referenced by Ui_KbBindWidget::setupUi().

```
190     {
191         KbBindWidget->setWindowTitle(QApplication::translate("KbBindWidget", "Form", Q_NULLPTR));
192         label->setText(QApplication::translate("KbBindWidget", "Keys", Q_NULLPTR));
193         copyButton->setText(QApplication::translate("KbBindWidget", "Copy to mode...", Q_NULLPTR)
    );
194         selectLabel->setText(QApplication::translate("KbBindWidget", "Click to select keys",
    Q_NULLPTR));
195         resetButton->setText(QApplication::translate("KbBindWidget", "Reset", Q_NULLPTR));
196         label_2->setText(QApplication::translate("KbBindWidget", "Binding", Q_NULLPTR));
197     } // retranslateUi
```

Here is the caller graph for this function:



12.26.2.2 void Ui_KbBindWidget::setupUi (QWidget * *KbBindWidget*) [inline],[inherited]

Definition at line 51 of file ui_kbbindwidget.h.

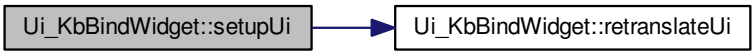
References Ui_KbBindWidget::copyButton, Ui_KbBindWidget::gridLayout, Ui_KbBindWidget::gridLayout_2, Ui_KbBindWidget::horizontalLayout_2, Ui_KbBindWidget::horizontalSpacer, Ui_KbBindWidget::horizontalSpacer_2, Ui_KbBindWidget::keyWidget, Ui_KbBindWidget::label, Ui_KbBindWidget::label_2, Ui_KbBindWidget::line, Ui_KbBindWidget::line_2, Ui_KbBindWidget::rbWidget, Ui_KbBindWidget::resetButton, Ui_KbBindWidget::retranslateUi(), Ui_KbBindWidget::selectLabel, Ui_KbBindWidget::verticalLayout, Ui_KbBindWidget::verticalSpacer, and Ui_KbBindWidget::widget.

Referenced by KbBindWidget::KbBindWidget().

```
52     {
53         if (KbBindWidget->objectName().isEmpty())
54             KbBindWidget->setObjectName(QStringLiteral("KbBindWidget"));
55         KbBindWidget->resize(581, 473);
56         verticalLayout = new QVBoxLayout(KbBindWidget);
57         verticalLayout->setObjectName(QStringLiteral("verticalLayout"));
58         verticalLayout->setContentsMargins(0, 0, 0, -1);
59         label = new QLabel(KbBindWidget);
60         label->setObjectName(QStringLiteral("label"));
61         QFont font;
62         font.setBold(true);
63         font.setWeight(75);
64         label->setFont(font);
65 
66         verticalLayout->addWidget(label);
67 
68         line = new QFrame(KbBindWidget);
69         line->setObjectName(QStringLiteral("line"));
70         line->setFrameShape(QFrame::HLine);
71         line->setFrameShadow(QFrame::Sunken);
72 
73         verticalLayout->addWidget(line);
74 
75         gridLayout = new QGridLayout();
76         gridLayout->setObjectName(QStringLiteral("gridLayout"));
77         horizontalSpacer_2 = new QSpacerItem(40, 20, QSizePolicy::Expanding,
    QSizePolicy::Minimum);
78 
79         gridLayout->addItem(horizontalSpacer_2, 0, 4, 1, 1);
80 
81         keyWidget = new KeyWidget(KbBindWidget);
82         keyWidget->setObjectName(QStringLiteral("keyWidget"));
83         QSizePolicy sizePolicy(QSizePolicy::Preferred, QSizePolicy::Preferred);
84         sizePolicy.setHorizontalStretch(0);
85         sizePolicy.setVerticalStretch(0);
86         sizePolicy.setHeightForWidth(keyWidget->sizePolicy().hasHeightForWidth());
87         keyWidget->setSizePolicy(sizePolicy);
88 
89         gridLayout->addWidget(keyWidget, 0, 1, 1, 3);
90 
91         horizontalSpacer = new QSpacerItem(40, 20, QSizePolicy::Expanding,
    QSizePolicy::Minimum);
92 
93         gridLayout->addItem(horizontalSpacer, 0, 0, 1, 1);
94 
95         horizontalLayout_2 = new QHBoxLayout();
96         horizontalLayout_2->setSpacing(0);
97         horizontalLayout_2->setObjectName(QStringLiteral("horizontalLayout_2"));
98         copyButton = new QPushButton(KbBindWidget);
99         copyButton->setObjectName(QStringLiteral("copyButton"));
100         QSizePolicy sizePolicy1(QSizePolicy::Fixed, QSizePolicy::Fixed);
101         sizePolicy1.setHorizontalStretch(0);
102         sizePolicy1.setVerticalStretch(0);
103         sizePolicy1.setHeightForWidth(copyButton->sizePolicy().hasHeightForWidth());
104         copyButton->setSizePolicy(sizePolicy1);
105         copyButton->setMinimumSize(QSize(140, 0));
106         copyButton->setMaximumSize(QSize(140, 16777215));
107 
108         horizontalLayout_2->addWidget(copyButton);
109 
110         selectLabel = new QLabel(KbBindWidget);
111         selectLabel->setObjectName(QStringLiteral("selectLabel"));
112         QSizePolicy sizePolicy2(QSizePolicy::Expanding, QSizePolicy::Preferred);
113         sizePolicy2.setHorizontalStretch(0);
114         sizePolicy2.setVerticalStretch(0);
115         sizePolicy2.setHeightForWidth(selectLabel->sizePolicy().hasHeightForWidth());
116         selectLabel->setSizePolicy(sizePolicy2);
117         selectLabel->setAlignment(Qt::AlignCenter);
118 
119         horizontalLayout_2->addWidget(selectLabel);
```

```
120
121         resetButton = new QPushButton(KbBindWidget);
122         resetButton->setObjectName(QStringLiteral("resetButton"));
123         QSizePolicy sizePolicy3(QSizePolicy::Expanding, QSizePolicy::Fixed);
124         sizePolicy3.setHorizontalStretch(0);
125         sizePolicy3.setVerticalStretch(0);
126         sizePolicy3.setHeightForWidth(resetButton->sizePolicy().hasHeightForWidth());
127         resetButton->setSizePolicy(sizePolicy3);
128         resetButton->setMinimumSize(QSize(140, 0));
129         resetButton->setMaximumSize(QSize(140, 16777215));
130
131         horizontalLayout_2->addWidget(resetButton);
132
133
134         gridLayout->addLayout(horizontalLayout_2, 1, 1, 1, 3);
135
136
137         verticalLayout->addLayout(gridLayout);
138
139         verticalSpacer = new QSpacerItem(20, 40, QSizePolicy::Minimum, QSizePolicy::Expanding
    );
140
141         verticalLayout->addItem(verticalSpacer);
142
143         label_2 = new QLabel(KbBindWidget);
144         label_2->setObjectName(QStringLiteral("label_2"));
145         label_2->setFont(font);
146
147         verticalLayout->addWidget(label_2);
148
149         line_2 = new QFrame(KbBindWidget);
150         line_2->setObjectName(QStringLiteral("line_2"));
151         line_2->setFrameShape(QFrame::HLine);
152         line_2->setFrameShadow(QFrame::Sunken);
153
154         verticalLayout->addWidget(line_2);
155
156         widget = new QWidget(KbBindWidget);
157         widget->setObjectName(QStringLiteral("widget"));
158         QSizePolicy sizePolicy4(QSizePolicy::Preferred, QSizePolicy::Fixed);
159         sizePolicy4.setHorizontalStretch(0);
160         sizePolicy4.setVerticalStretch(0);
161         sizePolicy4.setHeightForWidth(widget->sizePolicy().hasHeightForWidth());
162         widget->setSizePolicy(sizePolicy4);
163         widget->setMinimumSize(QSize(0, 280));
164         widget->setMaximumSize(QSize(16777215, 280));
165         gridLayout_2 = new QGridLayout(widget);
166         gridLayout_2->setSpacing(0);
167         gridLayout_2->setObjectName(QStringLiteral("gridLayout_2"));
168         gridLayout_2->setContentsMargins(0, 0, 0, 0);
169         rbWidget = new RebindWidget(widget);
170         rbWidget->setObjectName(QStringLiteral("rbWidget"));
171         QSizePolicy sizePolicy5(QSizePolicy::Expanding, QSizePolicy::Expanding);
172         sizePolicy5.setHorizontalStretch(0);
173         sizePolicy5.setVerticalStretch(0);
174         sizePolicy5.setHeightForWidth(rbWidget->sizePolicy().hasHeightForWidth());
175         rbWidget->setSizePolicy(sizePolicy5);
176
177         gridLayout_2->addWidget(rbWidget, 0, 0, 1, 1);
178
179
180         verticalLayout->addWidget(widget);
181
182         QWidget::setTabOrder(copyButton, resetButton);
183
184         retranslateUi(KbBindWidget);
185
186         QMetaObject::connectSlotsByName(KbBindWidget);
187     } // setupUi
```

Here is the call graph for this function:



Here is the caller graph for this function:

12.26.3 Field Documentation

12.26.3.1 QPushButton* Ui_KbBindWidget::copyButton [inherited]

Definition at line 41 of file ui_kbbindwidget.h.

Referenced by Ui_KbBindWidget::retranslateUi(), and Ui_KbBindWidget::setupUi().

12.26.3.2 QGridLayout* Ui_KbBindWidget::gridLayout [inherited]

Definition at line 36 of file ui_kbbindwidget.h.

Referenced by Ui_KbBindWidget::setupUi().

12.26.3.3 QGridLayout* Ui_KbBindWidget::gridLayout_2 [inherited]

Definition at line 48 of file ui_kbbindwidget.h.

Referenced by Ui_KbBindWidget::setupUi().

12.26.3.4 QHBoxLayout* Ui_KbBindWidget::horizontalLayout_2 [inherited]

Definition at line 40 of file ui_kbbindwidget.h.

Referenced by Ui_KbBindWidget::setupUi().

12.26.3.5 QSpacerItem* Ui_KbBindWidget::horizontalSpacer [inherited]

Definition at line 39 of file ui_kbbindwidget.h.

Referenced by Ui_KbBindWidget::setupUi().

12.26.3.6 QSpacerItem* Ui_KbBindWidget::horizontalSpacer_2 [inherited]

Definition at line 37 of file ui_kbbindwidget.h.

Referenced by Ui_KbBindWidget::setupUi().

12.26.3.7 KeyWidget* Ui_KbBindWidget::keyWidget [inherited]

Definition at line 38 of file ui_kbbindwidget.h.

Referenced by KbBindWidget::KbBindWidget(), KbBindWidget::newLayout(), KbBindWidget::setBind(), Ui_KbBindWidget::setupUi(), and KbBindWidget::updateBind().

12.26.3.8 QLabel* Ui_KbBindWidget::label [inherited]

Definition at line 34 of file ui_kbbindwidget.h.

Referenced by Ui_KbBindWidget::retranslateUi(), and Ui_KbBindWidget::setupUi().

12.26.3.9 QLabel* Ui_KbBindWidget::label_2 [inherited]

Definition at line 45 of file ui_kbbindwidget.h.

Referenced by Ui_KbBindWidget::retranslateUi(), and Ui_KbBindWidget::setupUi().

12.26.3.10 QFrame* Ui_KbBindWidget::line [inherited]

Definition at line 35 of file ui_kbbindwidget.h.

Referenced by Ui_KbBindWidget::setupUi().

12.26.3.11 QFrame* Ui_KbBindWidget::line_2 [inherited]

Definition at line 46 of file ui_kbbindwidget.h.

Referenced by Ui_KbBindWidget::setupUi().

12.26.3.12 RebindWidget* Ui_KbBindWidget::rbWidget [inherited]

Definition at line 49 of file ui_kbbindwidget.h.

Referenced by KbBindWidget::newLayout(), KbBindWidget::newSelection(), KbBindWidget::setBind(), Ui_KbBindWidget::setupUi(), and KbBindWidget::updateBind().

12.26.3.13 QPushButton* Ui_KbBindWidget::resetButton [inherited]

Definition at line 43 of file ui_kbbindwidget.h.

Referenced by Ui_KbBindWidget::retranslateUi(), and Ui_KbBindWidget::setupUi().

12.26.3.14 QLabel* Ui_KbBindWidget::selectLabel [inherited]

Definition at line 42 of file ui_kbbindwidget.h.

Referenced by Ui_KbBindWidget::retranslateUi(), Ui_KbBindWidget::setupUi(), and KbBindWidget::updateSelDisplay().

12.26.3.15 QVBoxLayout* Ui_KbBindWidget::verticalLayout [inherited]

Definition at line 33 of file ui_kbbindwidget.h.

Referenced by Ui_KbBindWidget::setupUi().

12.26.3.16 QSpacerItem* Ui_KbBindWidget::verticalSpacer [inherited]

Definition at line 44 of file ui_kbbindwidget.h.

Referenced by Ui_KbBindWidget::setupUi().

12.26.3.17 QWidget* Ui_KbBindWidget::widget [inherited]

Definition at line 47 of file ui_kbbindwidget.h.

Referenced by Ui_KbBindWidget::setupUi().

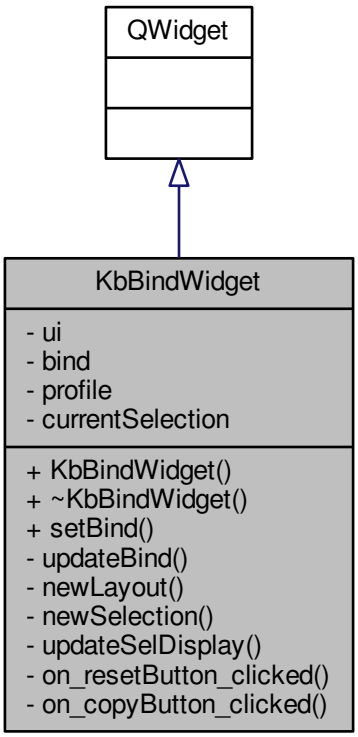
The documentation for this class was generated from the following file:

- src/ckb/ui_kbbindwidget.h

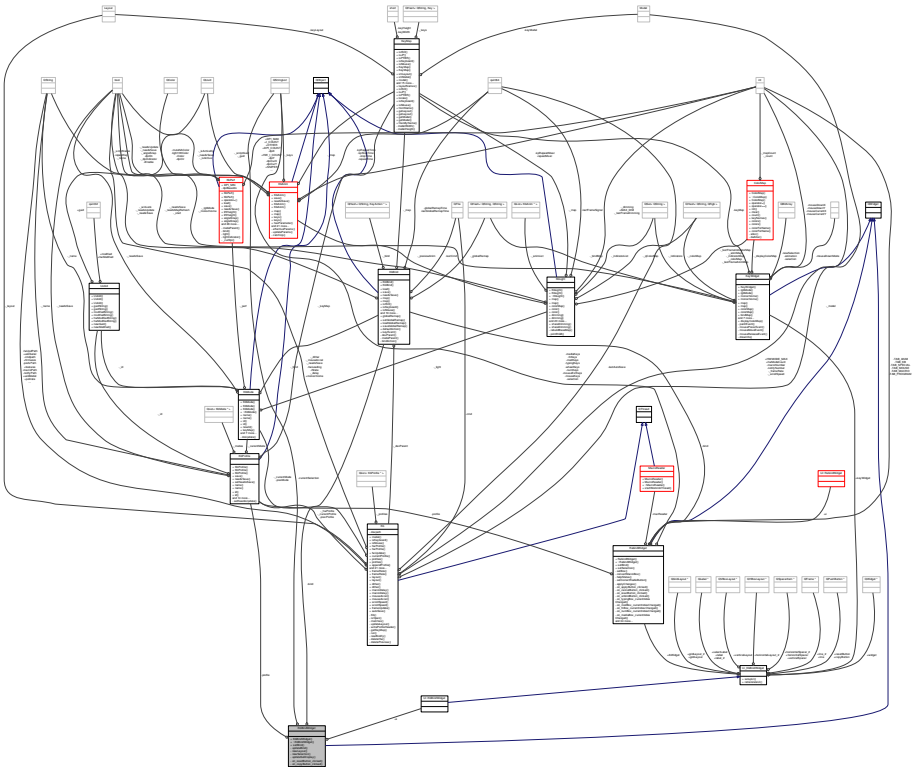
12.27 KbBindWidget Class Reference

`#include <src/ckb/kbbindwidget.h>`

Inheritance diagram for KbBindWidget:



Collaboration diagram for KbBindWidget:



Public Member Functions

- KbBindWidget (QWidget ∗parent=0)
- ∼KbBindWidget ()
- void setBind (KbBind ∗newBind, KbProfile ∗newProfile)

Private Slots

- void updateBind ()
- void newLayout ()
- void newSelection (QStringList selection)
- void updateSelDisplay ()
- void on_resetButton_clicked ()
- void on_copyButton_clicked ()

Private Attributes

- Ui::KbBindWidget ∗ ui
- KbBind ∗ bind
- KbProfile ∗ profile
- QStringList currentSelection

12.27.1 Detailed Description

Definition at line 12 of file kbbindwidget.h.

12.27.2 Constructor & Destructor Documentation

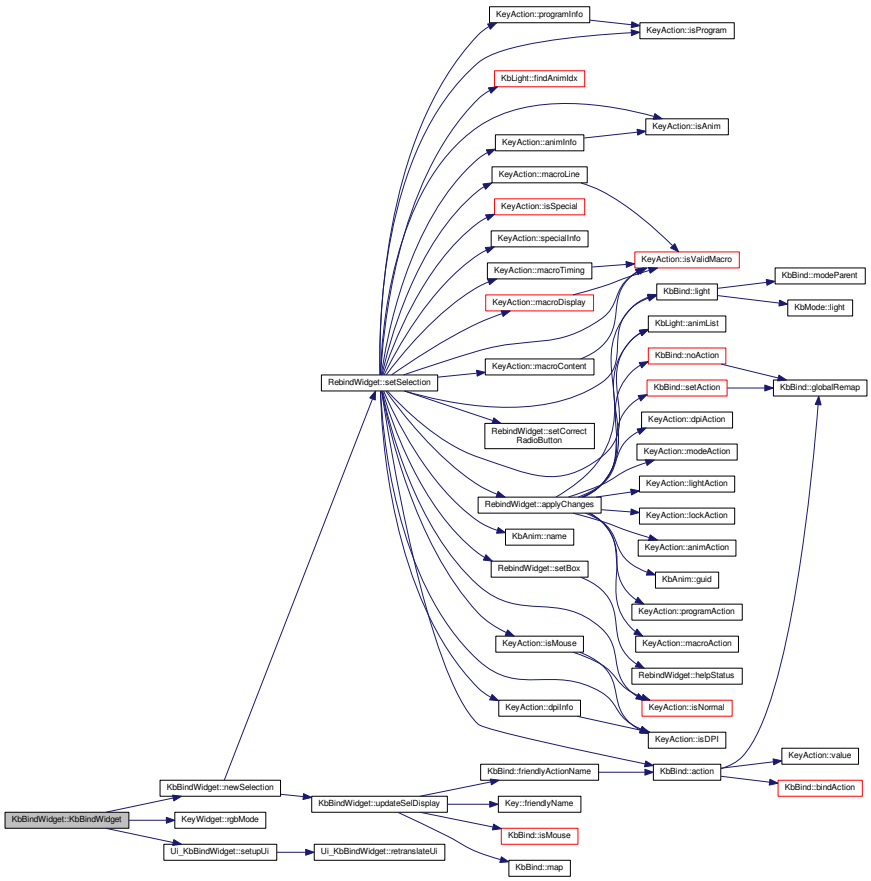
12.27.2.1 KbBindWidget::KbBindWidget (QWidget * *parent* = 0) [explicit]

Definition at line 7 of file kbbindwidget.cpp.

References Ui_KbBindWidget::keyWidget, newSelection(), KeyWidget::rgbMode(), Ui_KbBindWidget::setupUi(), and ui.

```
7                                                :
8     QWidget(parent),
9     ui(new Ui::KbBindWidget), bind(0), profile(0)
10 {
11     ui->setupUi(this);
12     ui->keyWidget->rgbMode(false);
13     connect(ui->keyWidget, SIGNAL(selectionChanged(QStringList)), this, SLOT(
       newSelection(QStringList)));
14 }
```

Here is the call graph for this function:



12.27.2.2 KbBindWidget::~KbBindWidget ( )

Definition at line 16 of file kbbindwidget.cpp.

References ui.

```
16                                {
17     delete ui;
18 }
```

12.27.3 Member Function Documentation

12.27.3.1 void KbBindWidget::newLayout ( ) [private], [slot]

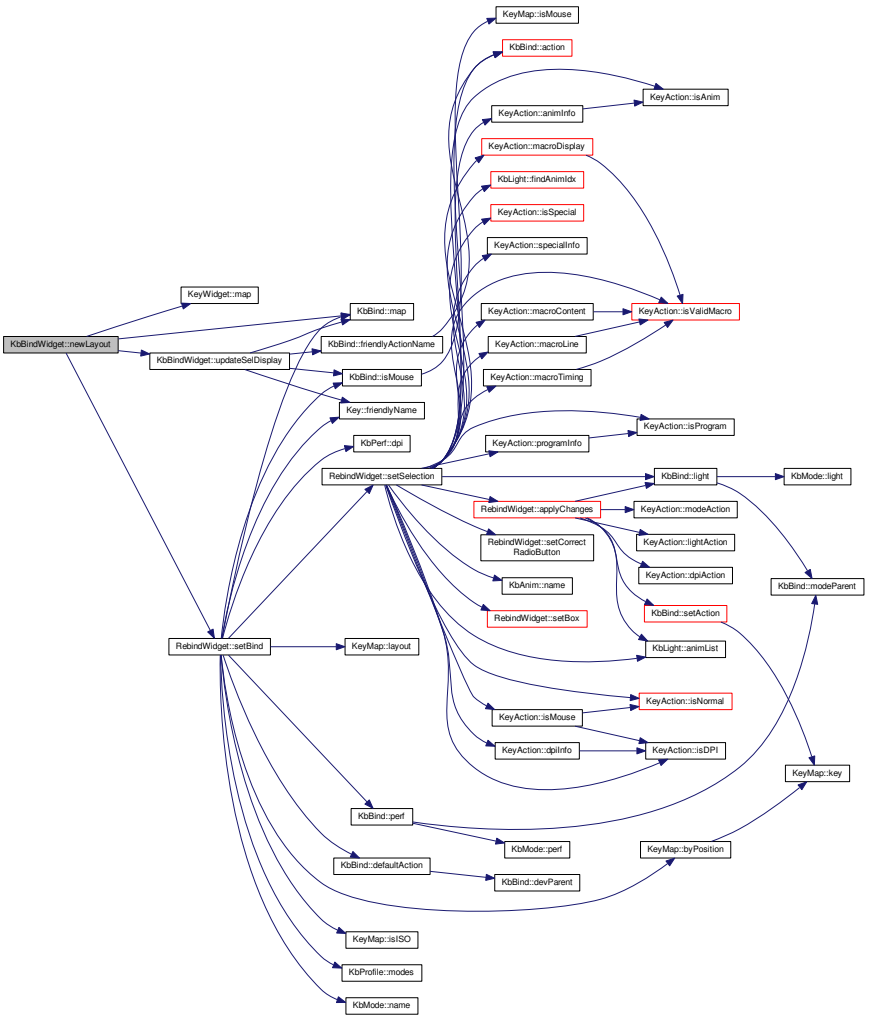
Definition at line 50 of file kbbindwidget.cpp.

References bind, Ui_KbBindWidget::keyWidget, KeyWidget::map(), KbBind::map(), profile, Ui_KbBindWidget::rbWidget, RebindWidget::setBind(), ui, and updateSelDisplay().

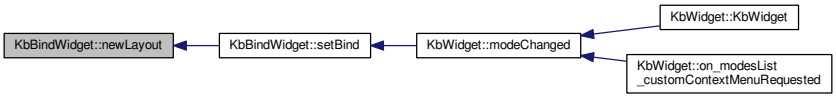
Referenced by setBind().

```
50                                {
51     ui->keyWidget->map(bind->map());
52     ui->rbWidget->setBind(bind, profile);
53     updateSelDisplay();
54 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.27.3.2 void KbBindWidget::newSelection (QStringList *selection*) [private], [slot]

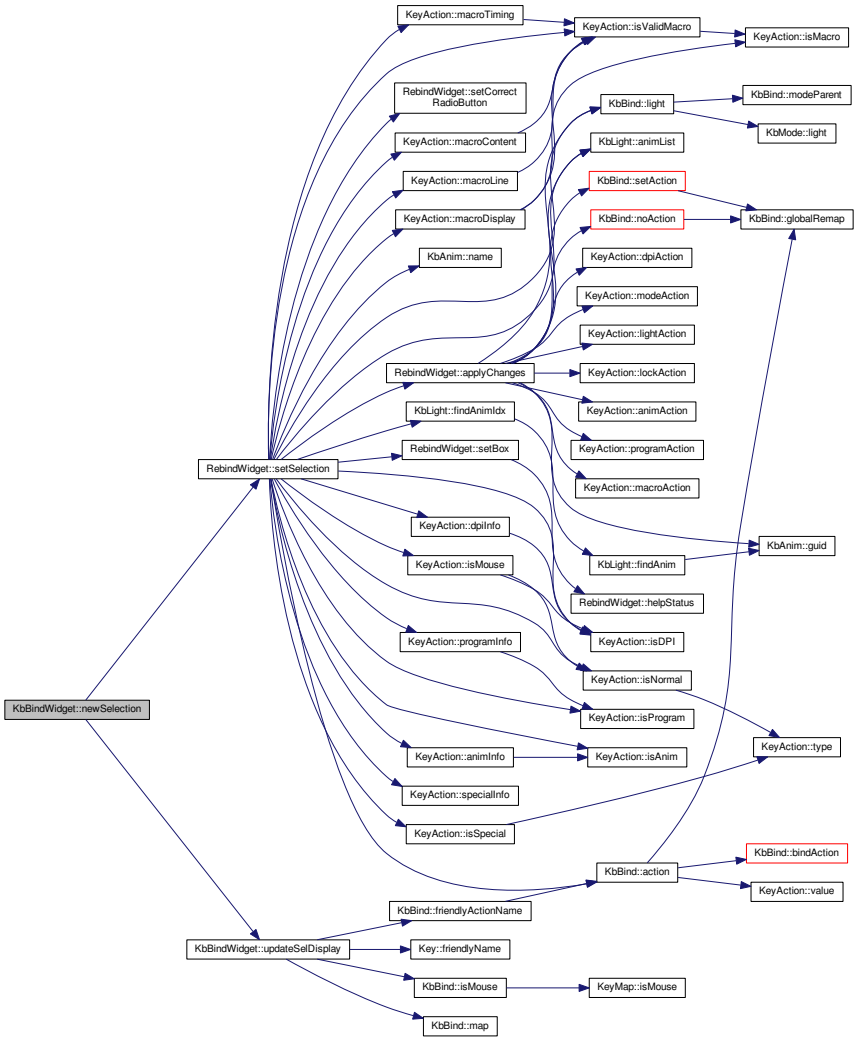
Definition at line 56 of file kbbindwidget.cpp.

References currentSelection, Ui_KbBindWidget::rbWidget, RebindWidget::setSelection(), ui, and updateSelDisplay().

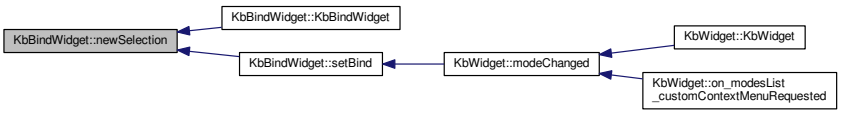
Referenced by KbBindWidget(), and setBind().

```
56                                                  {
57     currentSelection = selection;
58     ui->rbWidget->setSelection(selection, true);
59     updateSelDisplay();
60 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



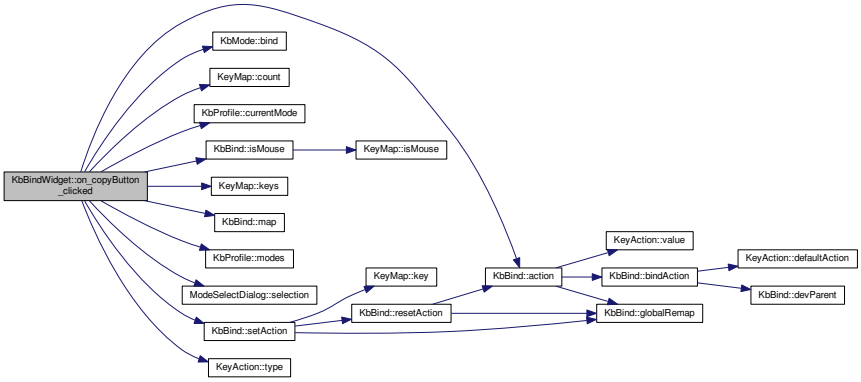
12.27.3.3 void KbBindWidget::on_copyButton_clicked ( ) [private], [slot]

Definition at line 103 of file kbbindwidget.cpp.

References KbBind::action(), bind, KbMode::bind(), KeyMap::count(), KbProfile::currentMode(), currentSelection, KbBind::isMouse(), KeyMap::keys(), KbBind::map(), KbProfile::modes(), profile, ModeSelectDialog::selection(), KbBind::setAction(), and KeyAction::type().

```
103                                     {
104     QStringList selection = currentSelection;
105     const KeyMap& map = bind->map();
106     int count = selection.count();
107     QString type = bind->isMouse() ? "button" : "key";
108     QString text = tr("%1 %2").arg(count).arg(type) + (count == 1 ? "" : "s");
109     if(count == 0){
110         // Copy all keys if none selected
111         selection = map.keys();
112         text = tr("all %1s").arg(type);
113     }
114     text = "Copy binding for " + text + " to:";
115     // Display popup
116     ModeSelectDialog dialog(this, profile->currentMode(),
    profile->modes(), text);
117     if(dialog.exec() != QDialog::Accepted)
118         return;
119     // Copy selected keys to selected modes
120     QList<KbMode*> selectedModes = dialog.selection();
121     foreach(KbMode* mode, selectedModes){
122         KbBind* modeBind = mode->bind();
123         foreach(const QString& key, selection){
124             modeBind->setAction(key, bind->action(key));
125         }
126     }
127 }
```

Here is the call graph for this function:



12.27.3.4 void KbBindWidget::on_resetButton_clicked ( ) [private],[slot]

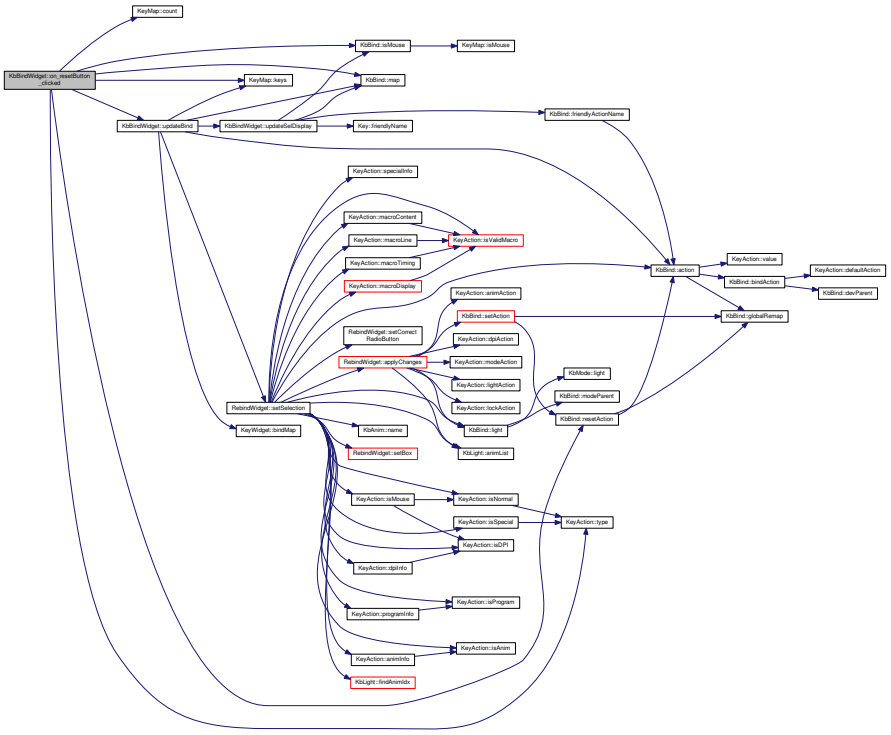
Definition at line 82 of file kbbindwidget.cpp.

References bind, KeyMap::count(), currentSelection, KbBind::isMouse(), KeyMap::keys(), KbBind::map(), KbBind::resetAction(), KeyAction::type(), and updateBind().

```
82                                     {
83     QStringList selection = currentSelection;
84     const KeyMap& map = bind->map();
85     if(selection.isEmpty())
86         // Reset all keys if none selected
87         selection = map.keys();
88     uint count = selection.count();
89     QString type = bind->isMouse() ? "button" : "key";
90     QString text;
91     if(count == map.count())
92         text = "<center>Reset all %1s to default?</center>";
93     else if(count == 1)
94         text = "<center>Reset this %1 to default?</center>";
95     else
96         text = tr("<center>Reset %1 %2s to default?</center>").arg(count);
97     if(QMessageBox(QMessageBox::NoIcon, "Confirm action", text.arg(type), QMessageBox::Yes |
    QMessageBox::No, this).exec() != QMessageBox::Yes)
98         return;
99     bind->resetAction(selection);
```

```
100     updateBind();
101 }
```

Here is the call graph for this function:



12.27.3.5 void KbBindWidget::setBind (KbBind * *newBind*, KbProfile * *newProfile*)

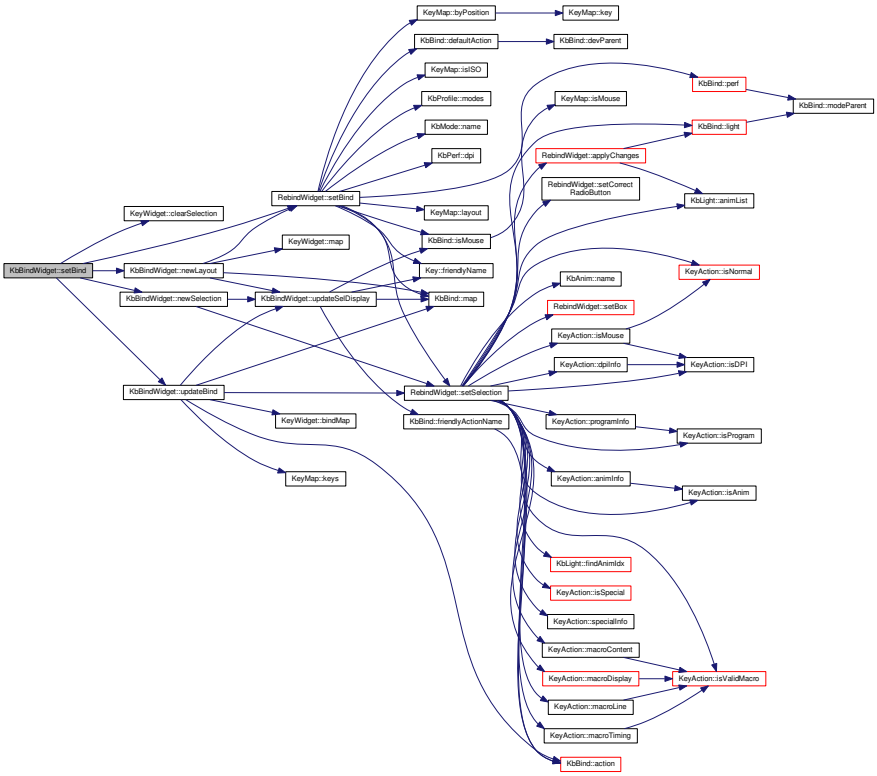
Definition at line 20 of file kbbindwidget.cpp.

References bind, KeyWidget::clearSelection(), Ui_KbBindWidget::keyWidget, newLayout(), newSelection(), profile, Ui_KbBindWidget::rbWidget, RebindWidget::setBind(), ui, and updateBind().

Referenced by KbWidget::modeChanged().

```
20                                                      {
21     ui->keyWidget->clearSelection();
22     ui->rbWidget->setBind(newBind, newProfile);
23     newSelection(QStringList());
24     if(bind == newBind)
25         return;
26     if(bind){
27         disconnect(bind, SIGNAL(layoutChanged()), this, SLOT(newLayout()));
28         disconnect(bind, SIGNAL(updated()), this, SLOT(updateBind()));
29     }
30     connect(newBind, SIGNAL(layoutChanged()), this, SLOT(newLayout()));
31     connect(newBind, SIGNAL(updated()), this, SLOT(updateBind()));
32     bind = newBind;
33     profile = newProfile;
34     newLayout();
35     ui->rbWidget->setBind(bind, profile);
36     updateBind();
37 }
```

Here is the call graph for this function:



Here is the caller graph for this function:

12.27.3.6 void KbBindWidget::updateBind ( ) [private], [slot]

Definition at line 39 of file kbbindwidget.cpp.

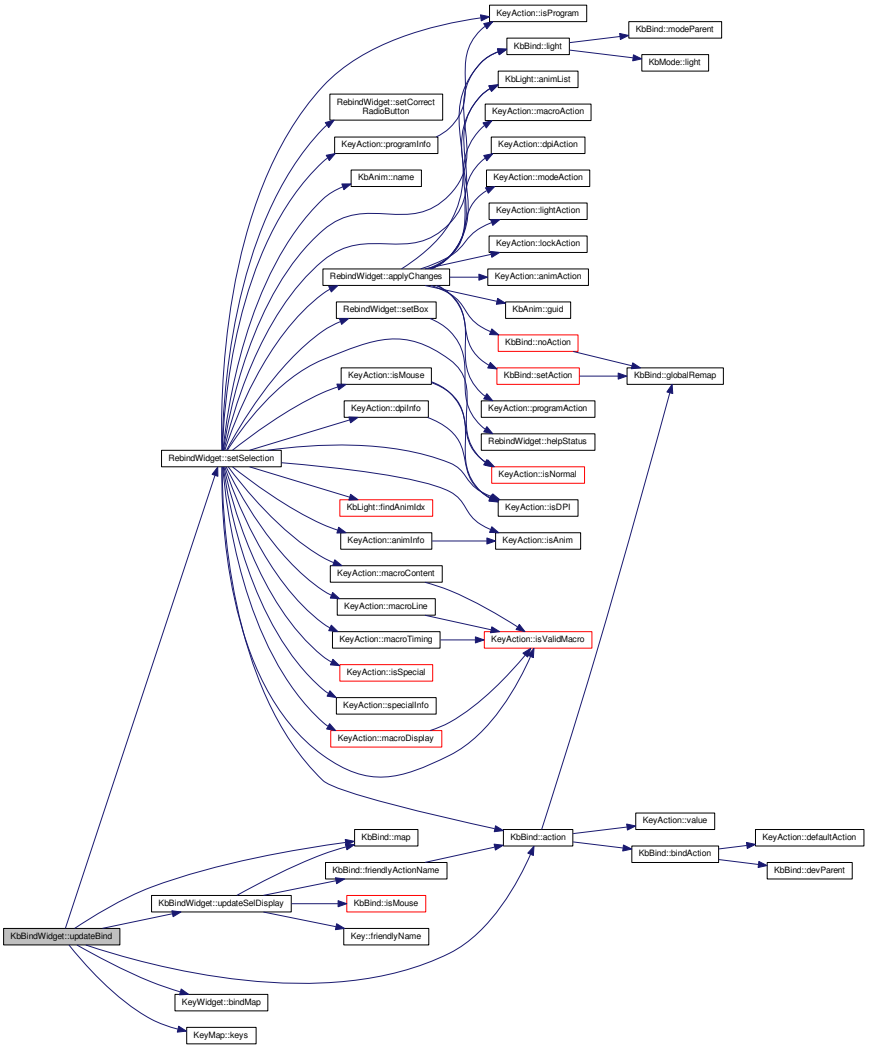
References KbBind::action(), bind, KeyWidget::bindMap(), currentSelection, KeyMap::keys(), Ui_KbBindWidget::keyWidget, KbBind::map(), Ui_KbBindWidget::rbWidget, RebindWidget::setSelection(), ui, and updateSelDisplay().

Referenced by on_resetButton_clicked(), and setBind().

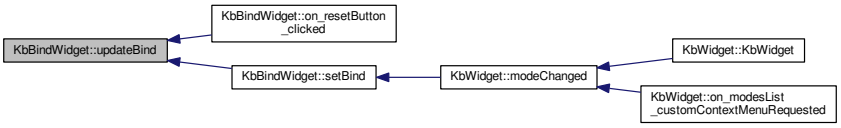
```
39                                {
40     const KeyMap& map = bind->map();
41     // Build the action list based on the keymap
42     QHash<QString, QString> actions;
43     foreach(const QString& key, map.keys())
44         actions[key] = bind->action(key);
45     ui->keyWidget->bindMap(actions);
46     ui->rbWidget->setSelection(currentSelection);
```

```
47     updateSelDisplay();
48 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.27.3.7 void KbBindWidget::updateSelDisplay ( ) [private], [slot]

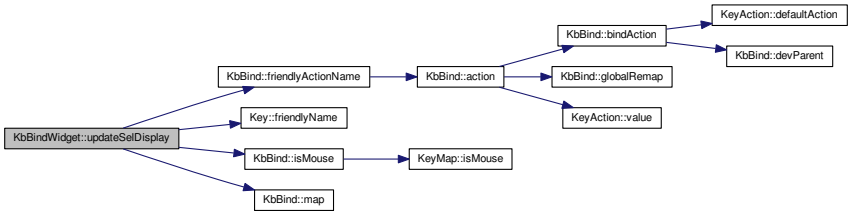
Definition at line 62 of file kbbindwidget.cpp.

References bind, currentSelection, KbBind::friendlyActionName(), Key::friendlyName(), KbBind::isMouse(), KbBind::map(), Ui_KbBindWidget::selectLabel, and ui.

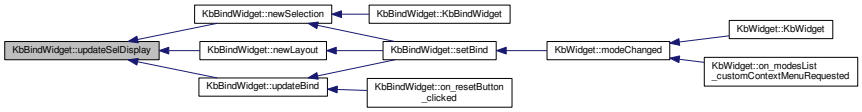
Referenced by newLayout(), newSelection(), and updateBind().

```
62                                      {
63     int count = currentSelection.count();
64     if(count == 0){
65         // No keys selected
66         ui->selectLabel->setText("Click to select");
67         return;
68     }
69     if(count == 1){
70         // Single key selected: show key name and binding
71         QString key = currentSelection[0];
72         const Key& pos = bind->map()[key];
73         if(!pos)
74             ui->selectLabel->setText("(Unknown)");
75         else
76             ui->selectLabel->setText(pos.friendlyName(false).split("\n")[0] + " →
      " + bind->friendlyActionName(key).split("\n")[0]);
77         return;
78     }
79     ui->selectLabel->setText(QString("%1 %2 selected").arg(count).arg(
      bind->isMouse() ? "buttons" : "keys"));
80 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.27.4 Field Documentation

12.27.4.1 KbBind* KbBindWidget::bind [private]

Definition at line 34 of file kbbindwidget.h.

Referenced by newLayout(), on_copyButton_clicked(), on_resetButton_clicked(), setBind(), updateBind(), and updateSelDisplay().

12.27.4.2 QStringList KbBindWidget::currentSelection [private]

Definition at line 36 of file kbbindwidget.h.

Referenced by newSelection(), on_copyButton_clicked(), on_resetButton_clicked(), updateBind(), and updateSelDisplay().

12.27.4.3 KbProfile* KbBindWidget::profile [private]

Definition at line 35 of file kbbindwidget.h.

Referenced by newLayout(), on_copyButton_clicked(), and setBind().

12.27.4.4 Ui::KbBindWidget* KbBindWidget::ui [private]

Definition at line 32 of file kbbindwidget.h.

Referenced by KbBindWidget(), newLayout(), newSelection(), setBind(), updateBind(), updateSelDisplay(), and ~KbBindWidget().

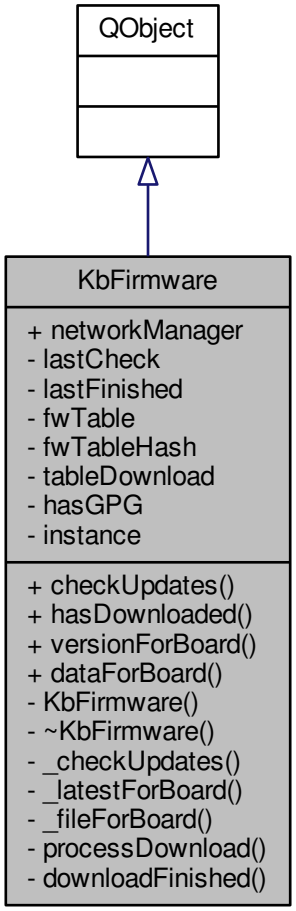
The documentation for this class was generated from the following files:

- src/ckb/kbbindwidget.h
- src/ckb/kbbindwidget.cpp

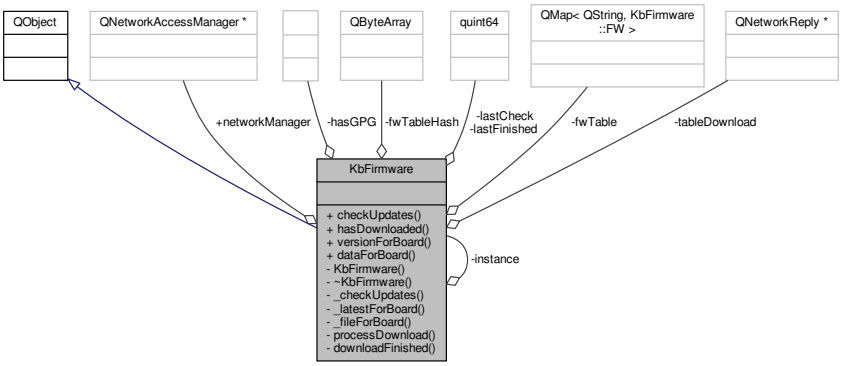
12.28 KbFirmware Class Reference

`#include <src/ckb/kbfirmware.h>`

Inheritance diagram for KbFirmware:



Collaboration diagram for KbFirmware:



Data Structures

- struct FW

Signals

- void downloaded ()

Static Public Member Functions

- static bool checkUpdates ()
- static bool hasDownloaded ()
- static float versionForBoard (const QString &features, bool waitForComplete=false)
- static QByteArray dataForBoard (const QString &features)

Data Fields

- QNetworkAccessManager * networkManager

Private Types

- enum { UNKNOWN = -1, NO, YES }

Private Slots

- void processDownload (QNetworkReply *reply)
- void downloadFinished ()

Private Member Functions

- KbFirmware ()
- ~KbFirmware ()
- bool _checkUpdates ()
- float _latestForBoard (const QString &features, bool waitForComplete)
- QByteArray _fileForBoard (const QString &features)

Private Attributes

- quint64 lastCheck
- quint64 lastFinished
- QMap< QString, FW > fwTable
- QByteArray fwTableHash
- QNetworkReply * tableDownload
- enum KbFirmware:: { ... } hasGPG

Static Private Attributes

- static KbFirmware instance

12.28.1 Detailed Description

Definition at line 10 of file kbfirmware.h.

12.28.2 Member Enumeration Documentation

12.28.2.1 anonymous enum [private]

Enumerator

UNKNOWN

NO

YES

Definition at line 52 of file kbfirmware.h.

```
52 { UNKNOWN = -1, NO, YES } hasGPG :2;
```

12.28.3 Constructor & Destructor Documentation

12.28.3.1 KbFirmware::KbFirmware ( ) [private]

Definition at line 14 of file kbfirmware.cpp.

References networkManager.

```
14                        :
15     lastCheck(0), lastFinished(0), networkManager(0),
       tableDownload(0), hasGPG(UNKNOWN)
16 {
17     // Disable bearer polling. This corrects an issue with latency spikes when
18     // using WiFi. The problem and workaround are described here:
19     // https://lostdomain.org/2017/06/17/qt-qnetworkaccessmanager-causing-latency-spikes-on-wifi/
20     qputenv("QT_BEARER_POLL_TIMEOUT", QByteArray::number(-1));
21     networkManager = new QNetworkAccessManager();
22 }
```

12.28.3.2 KbFirmware::~KbFirmware ( ) [private]

Definition at line 24 of file kbfirmware.cpp.

References networkManager.

```
24                        {
25     delete networkManager;
26 }
```

12.28.4 Member Function Documentation

12.28.4.1 bool KbFirmware::_checkUpdates ( ) [private]

Definition at line 28 of file kbfirmware.cpp.

References AUTO_CHECK_TIME, downloadFinished(), lastCheck, networkManager, and tableDownload.

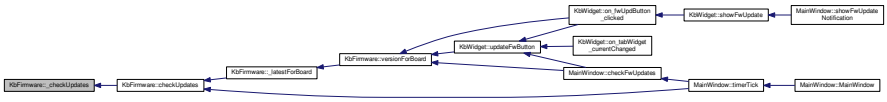
Referenced by checkUpdates().

```
28                                {
29     quint64 now = QDateTime::currentMSecsSinceEpoch();
30     if(now < lastCheck + AUTO_CHECK_TIME)
31         return false;
```

```
32     // First location is for debugging only.
33     // tableDownload =
       networkManager->get(QNetworkRequest(QUrl("https://raw.githubusercontent.com/frickler24/ckb-next/issues-26-Firmware-Inc
34     // This one is the production one.
35     tableDownload = networkManager->get(QNetworkRequest(QUrl("
      https://raw.githubusercontent.com/mattanger/ckb-next/master/FIRMWARE")));
36     connect(tableDownload, SIGNAL(finished()), this, SLOT(
      downloadFinished()));
37     lastCheck = now;
38     return true;
39 }
```

Here is the call graph for this function:

Here is the caller graph for this function:



12.28.4.2 QByteArray KbFirmware::_fileForBoard (const QString & *features*) [private]

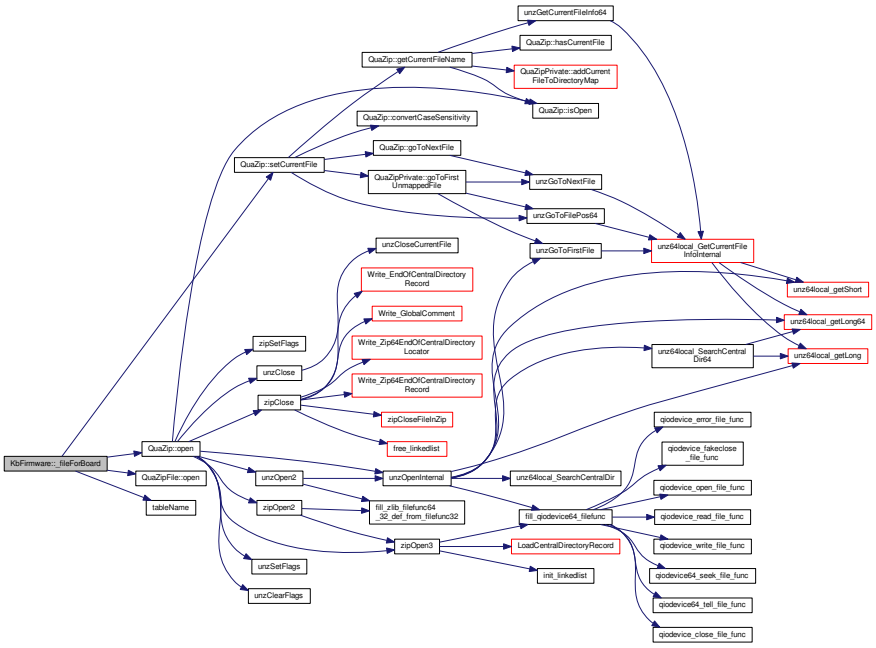
Definition at line 180 of file kbfirmware.cpp.

References QuaZip::csInsensitive, KbFirmware::FW::fileName, fwTable, KbFirmware::FW::hash, QuaZip::mdUnzip, networkManager, QuaZip::open(), QuaZipFile::open(), QuaZip::setCurrentFile(), tableName(), and KbFirmware::FW::url.

Referenced by dataForBoard().

```
180                                                     {
181     QString name = tableName(features);
182     FW info = fwTable.value(name);
183     if(info.hash.isEmpty())
184         return "";
185     // Download zip from URL. Wait for it to finish.
186     QNetworkReply* reply = networkManager->get(QNetworkRequest(QUrl(info.url)));
187     QEventLoop loop(this);
188     connect(reply, SIGNAL(finished()), &loop, SLOT(quit()));
189     loop.exec();
190     // Download finished, process data
191     if(reply->error() != QNetworkReply::NoError)
192         return "";
193     QByteArray zipData = reply->readAll();
194     QBuffer buffer(&zipData);
195     // Open zip archive
196     QuaZip zip(&buffer);
197     if(!zip.open(QuaZip::mdUnzip))
198         return "";
199     // Find the desired file
200     if(!zip.setCurrentFile(info.fileName, QuaZip::csInsensitive))
201         return "";
202     QuaZipFile binFile(&zip);
203     if(!binFile.open(QIODevice::ReadOnly))
204         return "";
205     QByteArray binary = binFile.readAll();
206     // Check the hash
207     if(QCryptographicHash::hash(binary, QCryptographicHash::Sha256) != info.hash)
208         return "";
209     return binary;
210 }
```

Here is the call graph for this function:



Here is the caller graph for this function:

12.28.4.3 float KbFirmware::_latestForBoard (const QString & *features,* bool *waitForComplete*) [private]

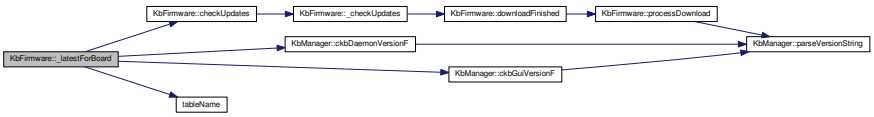
Definition at line 162 of file kbfirmware.cpp.

References checkUpdates(), KbManager::ckbDaemonVersionF(), KbManager::ckbGuiVersionF(), KbFirmware::FW::ckbVersion, downloaded(), fwTable, KbFirmware::FW::fwVersion, KbFirmware::FW::hash, tableDownload, and tableName().

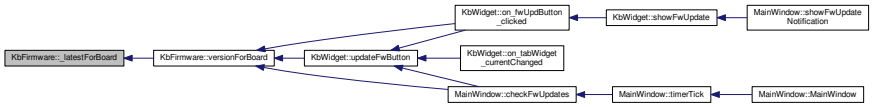
Referenced by versionForBoard().

```
162                                                                                 {
163     if((tableDownload || checkUpdates()) && waitForComplete){
164         // If waiting is desired, enter an event loop and stay here until the download is finished
165         QEventLoop loop(this);
166         connect(this, SIGNAL(downloaded()), &loop, SLOT(quit()));
167         loop.exec();
168     }
169     // Find this board
170     QString name = tableName(features);
171     FW info = fwTable.value(name);
172     if(info.hash.isEmpty())
173         return 0.f;
174     // Don't return the new version if the current ckb doesn't support it
175     if(info.ckbVersion > KbManager::ckbGuiVersionF() || info.ckbVersion >
    KbManager::ckbDaemonVersionF())
176         return -1.f;
177     return info.fwVersion;
178 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.28.4.4 static bool KbFirmware::checkUpdates ( ) [inline], [static]

Definition at line 15 of file kbfirmware.h.

References _checkUpdates(), and instance.

Referenced by _latestForBoard(), and MainWindow::timerTick().

```
15 { return instance._checkUpdates(); }
```

Here is the call graph for this function:

Here is the caller graph for this function:

12.28.4.5 static QByteArray KbFirmware::dataForBoard (const QString & *features*) [inline], [static]

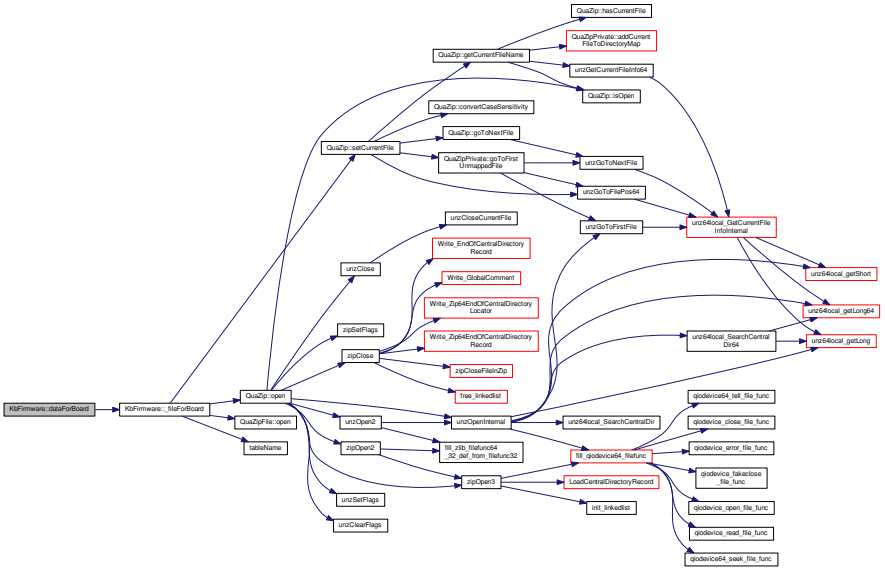
Definition at line 25 of file kbfirmware.h.

References _fileForBoard(), and instance.

Referenced by FwUpgradeDialog::exec().

```
25 { return instance._fileForBoard(features); }
```

Here is the call graph for this function:



Here is the caller graph for this function:

12.28.4.6 void KbFirmware::downloaded ( ) [signal]

Definition at line 148 of file moc_kbfirmware.cpp.

Referenced by _latestForBoard(), and downloadFinished().

```
149 {
150     QMetaObject::activate(this, &staticMetaObject, 0, Q_NULLPTR);
151 }
```

Here is the caller graph for this function:

12.28.4.7 void KbFirmware::downloadFinished ( ) [private], [slot]

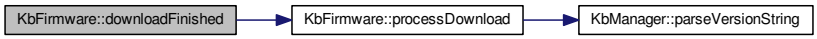
Definition at line 141 of file kbfirmware.cpp.

References downloaded(), processDownload(), and tableDownload.

Referenced by _checkUpdates().

```
141                                 {
142     if(!tableDownload)
143         return;
144     processDownload(tableDownload);
145     tableDownload->deleteLater();
146     tableDownload = 0;
147     emit downloaded();
148 }
```

Here is the call graph for this function:



Here is the caller graph for this function:

12.28.4.8 static bool KbFirmware::hasDownloaded ( ) [inline], [static]

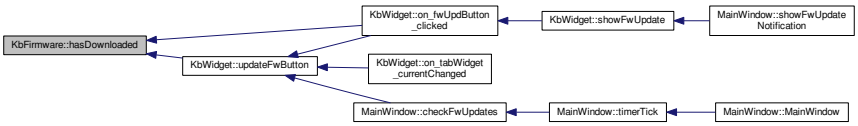
Definition at line 18 of file kbfirmware.h.

References instance, and lastFinished.

Referenced by KbWidget::on_fwUpdButton_clicked(), and KbWidget::updateFwButton().

```
18 { return instance.lastFinished != 0; }
```

Here is the caller graph for this function:



12.28.4.9 void KbFirmware::processDownload (QNetworkReply * *reply*) [private], [slot]

Definition at line 41 of file kbfirmware.cpp.

References KbFirmware::FW::ckbVersion, KbFirmware::FW::fileName, fwTable, fwTableHash, KbFirmware::FW::fwVersion, hasGPG, KbFirmware::FW::hash, lastCheck, lastFinished, NO, KbManager::parseVersionString(), UNKNOWN, KbFirmware::FW::url, and YES.

Referenced by downloadFinished().

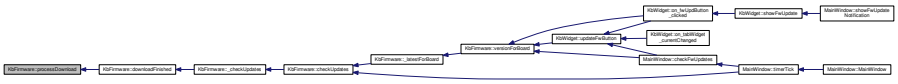
```
41                                                  {
42     if(reply->error() != QNetworkReply::NoError)
43         return;
```

```
44      // Update last check
45      lastCheck = lastFinished = QDateTime::currentMSecsSinceEpoch();
46      QByteArray data = reply->readAll();
47      // Don't do anything if this is the same as the last version downloaded
48      QByteArray hash = QCryptographicHash::hash(data, QCryptographicHash::Sha256);
49      if(hash == fwTableHash)
50          return;
51      fwTableHash = hash;
52      if(hasGPG == UNKNOWN){
53          // Check for a GPG installation
54          QProcess gpg;
55          gpg.start("gpg", QStringList("--version"));
56          gpg.waitForFinished();
57          if(gpg.error() == QProcess::FailedToStart)
58              // No GPG install
59              hasGPG = NO;
60          else {
61              QString output = QString::fromUtf8(gpg.readAll());
62              // Must support RSA keys and SHA256
63              if(output.contains("RSA", Qt::CaseInsensitive) && output.contains("SHA256", Qt::CaseInsensitive
    ))
64                  hasGPG = YES;
65              else
66                  hasGPG = NO;
67          }
68          if(!hasGPG)
69              qDebug() << "No GPG detected, signature verification disabled";
70      }
71      if(hasGPG){
72          // If GPG is available, check the signature on the file before proceeding.
73          QDir tmp = QDir::temp();
74          // Save file to a temporary path. Include PID to avoid conflicts
75          qint64 pid = QCoreApplication::applicationPid();
76          QString fwPath = tmp.absoluteFilePath(QString("ckb-%1-firmware").arg(pid));
77          QFile firmware(fwPath);
78          if(!firmware.open(QIODevice::WriteOnly)
79                  || firmware.write(data) != data.length()){
80              qDebug() << "Failed to write firmware file to temporary location, aborting firmware check";
81              return;
82          }
83          firmware.close();
84          // Write GPG key
85          QString keyPath = tmp.absoluteFilePath(QString("ckb-%1-key.gpg").arg(pid));
86          if(!QFile::copy(":/bin/ckb-next-key.gpg", keyPath)){
87              firmware.remove();
88              qDebug() << "Failed to write GPG key to temporary location, aborting firmware check";
89              return;
90          }
91          // Check signature
92          QProcess gpg;
93          gpg.start("gpg", QStringList("--no-default-keyring") << "--keyring" << keyPath << "--verify" <<
    fwPath);
94          gpg.waitForFinished();
95          // Clean up temp files
96          tmp.remove(fwPath);
97          tmp.remove(keyPath);
98          if(gpg.error() != QProcess::UnknownError || gpg.exitCode() != 0){
99              qDebug() << "GPG couldn't verify firmware signature:";
100              qDebug() << gpg.readAllStandardOutput();
101              qDebug() << gpg.readAllStandardError();
102              return;
103          }
104          // Signature good, proceed to update database
105      }
106      fwTable.clear();
107      QStringList lines = QString::fromUtf8(data).split("\n");
108      bool scan = false;
109      foreach(QString line, lines){
110          // Collapse whitespace
111          line.replace(QRegExp("\\s+"), " ").remove(QRegExp("^\\s")).remove(QRegExp("\\s$"));
112          // Skip empty or commented-out lines
113          if(line.length() == 0 || line.at(0) == '#')
114              continue;
115          // Don't read anything until the entries begin and don't read anything after they end
116          if(!scan){
117              if(line == "!BEGIN FW ENTRIES")
118                  scan = true;
119              else
120                  continue;
121          }
122          if(line == "!END FW ENTRIES")
123              break;
124          QStringList components = line.split(" ");
125          if(components.length() != 7)
126              continue;
127          // "VENDOR-PRODUCT"
128          QString device = components[0].toUpper() + "-" + components[1].toUpper();
```

```
129         FW fw;
130         fw.fwVersion = components[2].toFloat();                              // Firmware blob version
131         fw.url = QUrl::fromPercentEncoding(components[3].toLatin1());        // URL to zip file
132         fw.ckbVersion = KbManager::parseVersionString(components[4]);        //
      Minimum ckb version
133         fw.fileName = QUrl::fromPercentEncoding(components[5].toLatin1());   // Name of file inside zip
134         fw.hash = QByteArray::fromHex(components[6].toLatin1());             // SHA256 of file inside zip
135         // Update entry
136         fwTable[device] = fw;
137     }
138     qDebug() << "Downloaded new firmware list." << fwTable.count() << "entries found.";
139 }
```

Here is the call graph for this function:

Here is the caller graph for this function:



12.28.4.10 static float KbFirmware::versionForBoard (const QString & *features*, bool *waitForComplete* = `false`) [inline], [static]

Definition at line 22 of file kbfirmware.h.

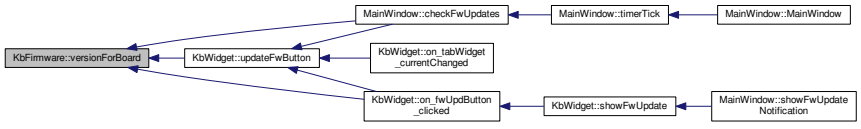
References _latestForBoard(), and instance.

Referenced by MainWindow::checkFwUpdates(), KbWidget::on_fwUpdButton_clicked(), and KbWidget::updateFwButton().

```
22 { return instance._latestForBoard(features, waitForComplete); }
```

Here is the call graph for this function:

Here is the caller graph for this function:



12.28.5 Field Documentation

12.28.5.1 QMap<QString, FW> KbFirmware::fwTable [private]

Definition at line 44 of file kbfirmware.h.

Referenced by _fileForBoard(), _latestForBoard(), and processDownload().

12.28.5.2 QByteArray KbFirmware::fwTableHash [private]

Definition at line 46 of file kbfirmware.h.

Referenced by processDownload().

12.28.5.3 enum { ... } KbFirmware::hasGPG [private]

Referenced by processDownload().

12.28.5.4 KbFirmware KbFirmware::instance [static], [private]

Definition at line 58 of file kbfirmware.h.

Referenced by checkUpdates(), dataForBoard(), hasDownloaded(), and versionForBoard().

12.28.5.5 quint64 KbFirmware::lastCheck [private]

Definition at line 35 of file kbfirmware.h.

Referenced by _checkUpdates(), and processDownload().

12.28.5.6 quint64 KbFirmware::lastFinished [private]

Definition at line 35 of file kbfirmware.h.

Referenced by hasDownloaded(), and processDownload().

12.28.5.7 QNetworkAccessManager* KbFirmware::networkManager

Definition at line 28 of file kbfirmware.h.

Referenced by _checkUpdates(), _fileForBoard(), KbFirmware(), and ~KbFirmware().

12.28.5.8 QNetworkReply* KbFirmware::tableDownload [private]

Definition at line 49 of file kbfirmware.h.

Referenced by _checkUpdates(), _latestForBoard(), and downloadFinished().

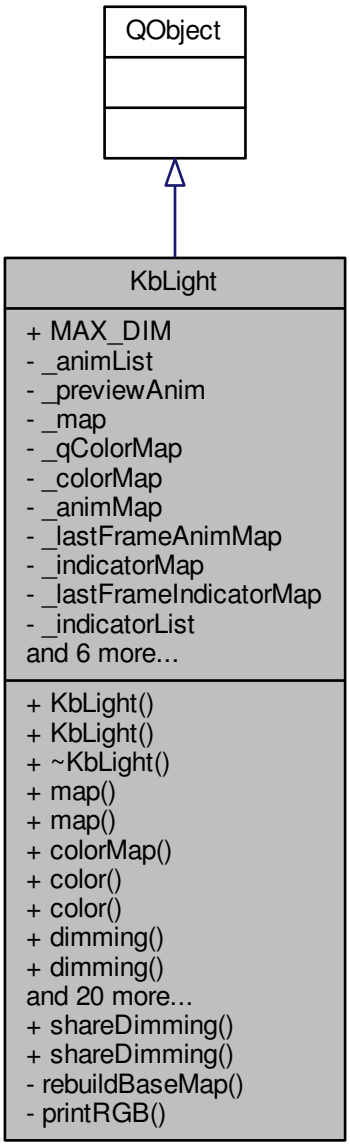
The documentation for this class was generated from the following files:

- src/ckb/kbfirmware.h
- src/ckb/kbfirmware.cpp
- src/ckb/release/moc_kbfirmware.cpp

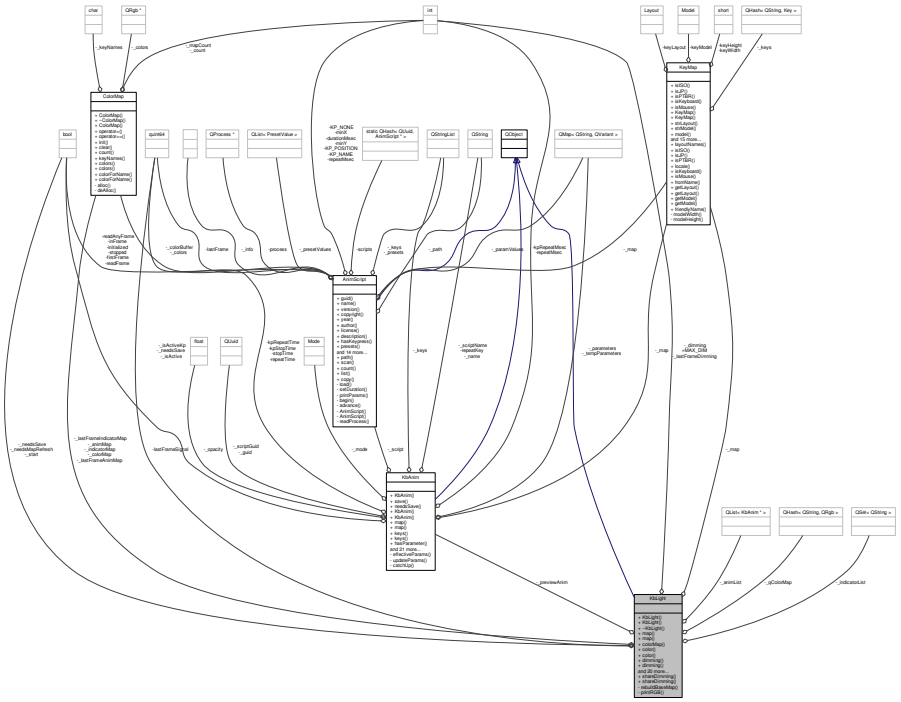
12.29 KbLight Class Reference

`#include <src/ckb/kblight.h>`

Inheritance diagram for KbLight:



Collaboration diagram for KbLight:



Public Types

- typedef QList< KbAnim * > AnimList

Signals

- void didLoad ()
- void updated ()
- void frameDisplayed (const ColorMap &animatedColors, const QSet< QString > &indicatorList)

Public Member Functions

- KbLight (KbMode *parent, const KeyMap &keyMap)
- KbLight (KbMode *parent, const KeyMap &keyMap, const KbLight &other)
- ~KbLight ()
- const KeyMap & map ()
- void map (const KeyMap &map)
- const QColorMap & colorMap ()
- void color (const QString &key, const QColor &newColor)
- void color (const QColor &newColor)
- int dimming ()
- void dimming (int newDimming)
- KbAnim * addAnim (const AnimScript *base, const QStringList &keys, const QString &name, const QMap< QString, QVariant > &preset)
- KbAnim * duplicateAnim (KbAnim *oldAnim)
- const AnimList & animList ()
- void animList (const AnimList &newAnimList)
- KbAnim * findAnim (const QUuid &guid) const

- int findAnimIdx (const QUuid &guid) const
- void previewAnim (const AnimScript *base, const QStringList &keys, const QMap< QString, QVariant > &preset)
- void stopPreview ()
- void restartAnimation ()
- void animKeypress (const QString &key, bool down)
- void open ()
- bool isStarted ()
- void close ()
- void resetIndicators ()
- void setIndicator (const char *name, QRgb argb)
- void frameUpdate (QFile &cmd, bool monochrome=false)
- void base (QFile &cmd, bool ignoreDim=false, bool monochrome=false)
- void load (CkbSettings &settings)
- void save (CkbSettings &settings)
- bool needsSave () const

Static Public Member Functions

- static int shareDimming ()
- static void shareDimming (int newShareDimming)

Static Public Attributes

- static const int MAX_DIM = 3

Private Member Functions

- void rebuildBaseMap ()
- void printRGB (QFile &cmd, const ColorMap &animMap)

Private Attributes

- AnimList _animList
- KbAnim * _previewAnim
- KeyMap _map
- QColorMap _qColorMap
- ColorMap _colorMap
- ColorMap _animMap
- ColorMap _lastFrameAnimMap
- ColorMap _indicatorMap
- ColorMap _lastFrameIndicatorMap
- QSet< QString > _indicatorList
- quint64 lastFrameSignal
- int _dimming
- int _lastFrameDimming
- bool _start
- bool _needsSave
- bool _needsMapRefresh

12.29.1 Detailed Description

Definition at line 17 of file kblight.h.

12.29.2 Member Typedef Documentation

12.29.2.1 typedef QList<KbAnim*> KbLight::AnimList

Definition at line 46 of file kblight.h.

12.29.3 Constructor & Destructor Documentation

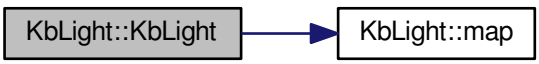
12.29.3.1 KbLight::KbLight (KbMode * *parent*, const KeyMap & *keyMap*)

Definition at line 10 of file kblight.cpp.

References map().

```
10                                                          :
11     QObject(parent), _previewAnim(0), lastFrameSignal(0),
      _dimming(0), _lastFrameDimming(0), _start(false),
      _needsSave(true), _needsMapRefresh(true)
12 {
13     map(keyMap);
14 }
```

Here is the call graph for this function:



12.29.3.2 KbLight::KbLight (KbMode * *parent*, const KeyMap & *keyMap*, const KbLight & *other*)

Definition at line 16 of file kblight.cpp.

References _animList, and map().

```
16                                                                                  :
17     QObject(parent), _previewAnim(0), _map(other._map),
      _qColorMap(other._qColorMap), lastFrameSignal(0),
      _dimming(other._dimming), _lastFrameDimming(other.
      _lastFrameDimming), _start(false), _needsSave(true),
      _needsMapRefresh(true)
18 {
19     map(keyMap);
20     // Duplicate animations
21     foreach(KbAnim* animation, other._animList)
22         _animList.append(new KbAnim(this, keyMap, *animation));
23 }
```

Here is the call graph for this function:

12.29.3.3 KbLight::~KbLight ( )

Definition at line 45 of file kblight.cpp.

References activeLights.

```
45                  {
46     activeLights.remove(this);
47 }
```

12.29.4 Member Function Documentation

12.29.4.1 KbAnim * KbLight::addAnim (const AnimScript * *base,* const QStringList & *keys,* const QString & *name,* const QMap< QString, QVariant > & *preset*)

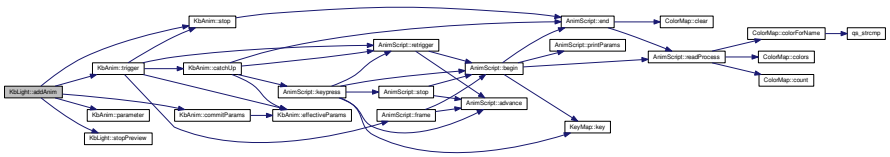
Definition at line 100 of file kblight.cpp.

References _animList, _map, _needsSave, _start, KbAnim::commitParams(), KbAnim::parameter(), KbAnim::stop(), stopPreview(), and KbAnim::trigger().

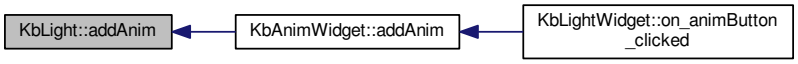
Referenced by KbAnimWidget::addAnim().

```
100
                            {
101     // Stop and restart all existing animations
102     stopPreview();
103     quint64 timestamp = QDateTime::currentMSecsSinceEpoch();
104     foreach(KbAnim* anim, _animList){
105         anim->stop();
106         anim->trigger(timestamp);
107     }
108     // Load the new animation and set preset parameters
109     KbAnim* anim = new KbAnim(this, _map, name, keys, base);
110     QMapIterator<QString, QVariant> i(preset);
111     while(i.hasNext()){
112         i.next();
113         anim->parameter(i.key(), i.value());
114     }
115     anim->commitParams();
116     // Add the animation and start it
117     _animList.append(anim);
118     anim->trigger(timestamp);
119     _start = true;
120     _needsSave = true;
121     return anim;
122 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.29.4.2 void KbLight::animKeypress (const QString & *key*, bool *down*)

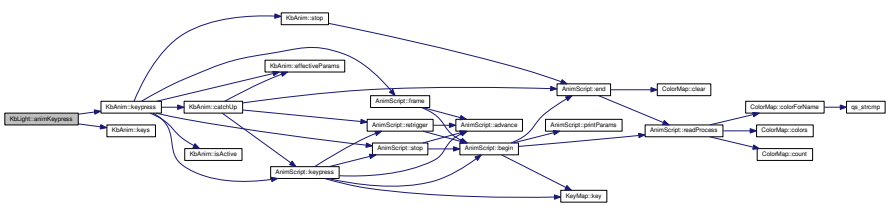
Definition at line 189 of file kblight.cpp.

References _animList, _previewAnim, KbAnim::keypress(), and KbAnim::keys().

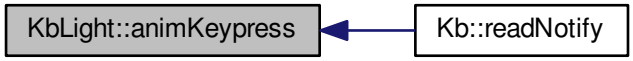
Referenced by Kb::readNotify().

```
189                                                         {
190     foreach(KbAnim* anim, _animList){
191         if(anim->keys().contains(key))
192             anim->keypress(key, down, QDateTime::currentMSecsSinceEpoch());
193     }
194     if(_previewAnim){
195         if(_previewAnim->keys().contains(key))
196             _previewAnim->keypress(key, down, QDateTime::currentMSecsSinceEpoch());
197     }
198 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.29.4.3 const AnimList& KbLight::animList ( ) [inline]

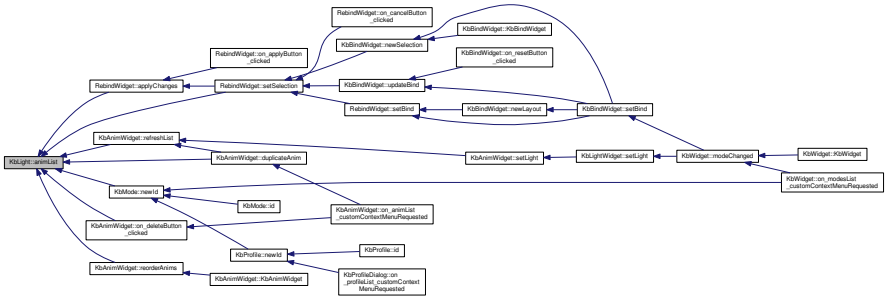
Definition at line 49 of file kblight.h.

References _animList.

Referenced by RebindWidget::applyChanges(), KbAnimWidget::duplicateAnim(), KbMode::newId(), KbAnimWidget::on_deleteButton_clicked(), KbAnimWidget::refreshList(), KbAnimWidget::reorderAnims(), and RebindWidget::setSelection().

```
49 { return _animList; }
```

Here is the caller graph for this function:



12.29.4.4 void KbLight::animList (const AnimList & *newAnimList*) [inline]

Definition at line 50 of file kblight.h.

References _animList, and _needsSave.

```
50 { _needsSave = true; _animList = newAnimList; }
```

12.29.4.5 void KbLight::base (QFile & *cmd*, bool *ignoreDim* = false, bool *monochrome* = false)

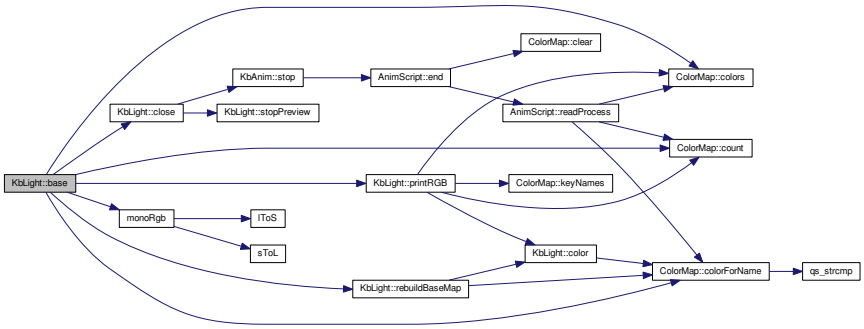
Definition at line 378 of file kblight.cpp.

References _animMap, _colorMap, _dimming, close(), ColorMap::colorForName(), ColorMap::colors(), ColorMap::count(), MAX_DIM, monoRgb(), printRGB(), and rebuildBaseMap().

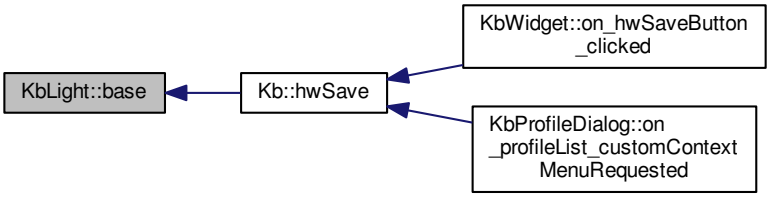
Referenced by Kb::hwSave().

```
378                                                                 {
379     close();
380     if(_dimming == MAX_DIM && !ignoreDim){
381         cmd.write(QString().sprintf("rgb 000000").toLatin1());
382         return;
383     }
384     // Set just the background color, ignoring any animation
385     rebuildBaseMap();
386     _animMap = _colorMap;
387     // If monochrome is active, create grayscale
388     if(monochrome){
389         int count = _animMap.count();
390         QRgb* colors = _animMap.colors();
391         for(int i = 0; i < count; i++){
392             QRgb& rgb = colors[i];
393             rgb = monoRgb(qRed(rgb), qGreen(rgb), qBlue(rgb));
394         }
395     }
396     // Set a few indicators to black as the hardware handles them differently
397     QRgb* mr = _animMap.colorForName("mr"), *m1 = _animMap.
      colorForName("m1"), *m2 = _animMap.colorForName("m2"), *m3 =
      _animMap.colorForName("m3"), *lock = _animMap.
      colorForName("lock");
398     if(mr) *mr = 0;
399     if(m1) *m1 = 0;
400     if(m2) *m2 = 0;
401     if(m3) *m3 = 0;
402     if(lock) *lock = 0;
403     // Send to driver
404     cmd.write("rgb");
405     printRGB(cmd, _animMap);
406 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.29.4.6 void KbLight::close ( )

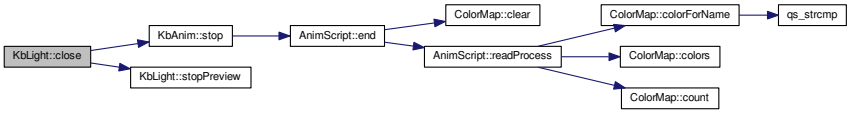
Definition at line 215 of file kblight.cpp.

References _animList, _start, activeLights, KbAnim::stop(), and stopPreview().

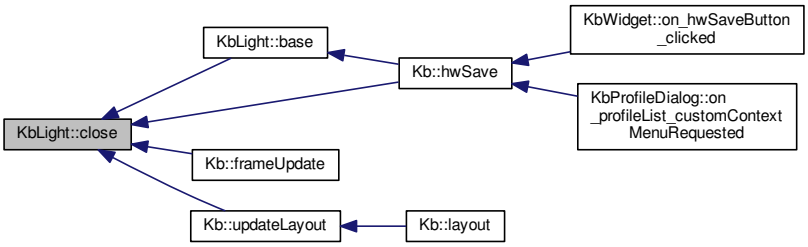
Referenced by base(), Kb::frameUpdate(), Kb::hwSave(), and Kb::updateLayout().

```
215                     {
216     activeLights.remove(this);
217     foreach(KbAnim* anim, _animList)
218         anim->stop();
219     stopPreview();
220     _start = false;
221 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.29.4.7 void KbLight::color (const QString & *key*, const QColor & *newColor*)

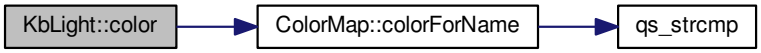
Definition at line 49 of file kblight.cpp.

References _colorMap, _needsMapRefresh, _needsSave, _qColorMap, and ColorMap::colorForName().

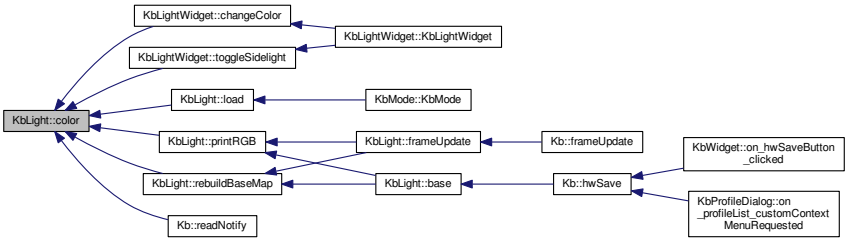
Referenced by KbLightWidget::changeColor(), load(), printRGB(), Kb::readNotify(), rebuildBaseMap(), and KbLightWidget::toggleSidelight().

```
49                                                      {
50     QRgb newRgb = newColor.rgb();
51     _qColorMap[key] = newRgb;
52     _needsSave = true;
53     if(!_needsMapRefresh){
54         // Update flat map if we're not scheduled to rebuild it
55         QByteArray rawName = key.toLatin1();
56         QRgb* rawRgb = _colorMap.colorForName(rawName.data());
57         if(rawRgb)
58             *rawRgb = newRgb;
59     }
60 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



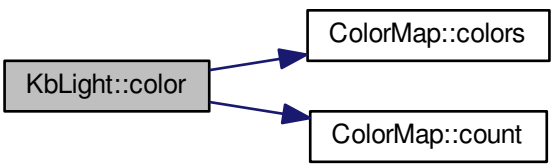
12.29.4.8 void KbLight::color (const QColor & *newColor*)

Definition at line 62 of file kblight.cpp.

References _colorMap, _needsMapRefresh, _needsSave, _qColorMap, ColorMap::colors(), and ColorMap::count().

```
62                                          {
63     QRgb newRgb = newColor.rgb();
64     QMutableColorMapIterator i(_qColorMap);
65     while(i.hasNext()){
66         i.next();
67         i.value() = newRgb;
68     }
69     _needsSave = true;
70     // Reset flat map
71     _needsMapRefresh = false;
72     int mapCount = _colorMap.count();
73     QRgb* flat = _colorMap.colors();
74     for(int i = 0; i < mapCount; i++)
75         flat[i] = mapCount;
76 }
```

Here is the call graph for this function:



12.29.4.9 const QColorMap& KbLight::colorMap ( ) [inline]

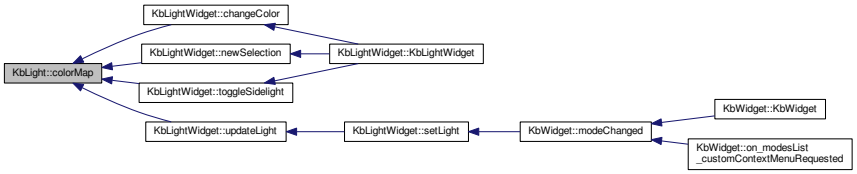
Definition at line 31 of file kblight.h.

References _qColorMap.

Referenced by KbLightWidget::changeColor(), KbLightWidget::newSelection(), KbLightWidget::toggleSidelight(), and KbLightWidget::updateLight().

```
31 { return _qColorMap; }
```

Here is the caller graph for this function:



12.29.4.10 void KbLight::didLoad ( ) [signal]

Definition at line 162 of file moc_kblight.cpp.

Referenced by load().

```
163 {
164     QMetaObject::activate(this, &staticMetaObject, 0, Q_NULLPTR);
165 }
```

Here is the caller graph for this function:

12.29.4.11 int KbLight::dimming ( ) [inline]

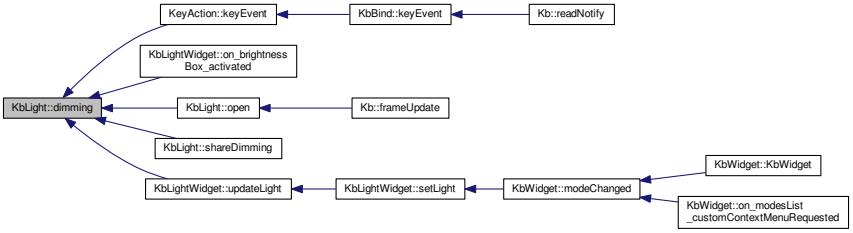
Definition at line 39 of file kblight.h.

References _dimming.

Referenced by KeyAction::keyEvent(), KbLightWidget::on_brightnessBox_activated(), open(), shareDimming(), and KbLightWidget::updateLight().

```
39 { return _dimming; }
```

Here is the caller graph for this function:



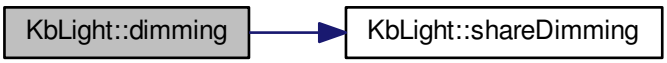
12.29.4.12 void KbLight::dimming (int *newDimming*)

Definition at line 92 of file kblight.cpp.

References _dimming, _needsSave, _shareDimming, shareDimming(), and updated().

```
92                                  {
93     if(_shareDimming != -1)
94         shareDimming(newDimming);
95     _needsSave = true;
96     _dimming = newDimming;
97     emit updated();
98 }
```

Here is the call graph for this function:



12.29.4.13 KbAnim * KbLight::duplicateAnim (KbAnim * *oldAnim*)

Definition at line 148 of file kblight.cpp.

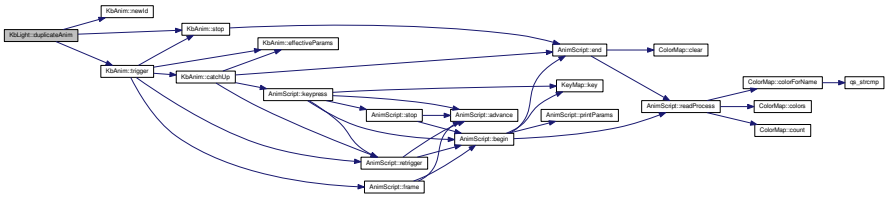
References _animList, _map, _needsSave, _start, KbAnim::newId(), KbAnim::stop(), and KbAnim::trigger().

Referenced by KbAnimWidget::duplicateAnim().

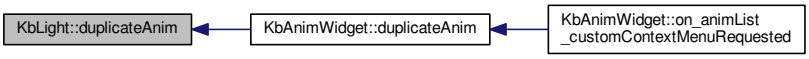
```
148                                          {
149     // Stop and restart all existing animations
150     quint64 timestamp = QDateTime::currentMSecsSinceEpoch();
151     foreach(KbAnim* anim, _animList){
152         anim->stop();
153         anim->trigger(timestamp);
154     }
155     // Same as addAnim, just duplicate the existing one
156     KbAnim* anim = new KbAnim(this, _map, *oldAnim);
157     anim->newId();
158     int index = _animList.indexOf(oldAnim);
159     if(index < 0)
160         _animList.append(anim);
161     else
162         _animList.insert(index + 1, anim);
```

```
163     anim->trigger(timestamp);
164     _start = true;
165     _needsSave = true;
166     return anim;
167 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.29.4.14 KbAnim* KbLight::findAnim (const QUuid & *guid*) const [inline]

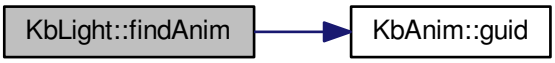
Definition at line 51 of file kblight.h.

References _animList, and KbAnim::guid().

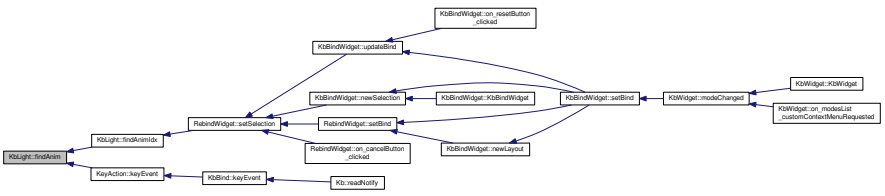
Referenced by findAnimIdx(), and KeyAction::keyEvent().

```
51 { foreach(KbAnim* anim, _animList) { if(anim->guid() == guid) return anim; } return 0; }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.29.4.15 int KbLight::findAnimIdx (const QUuid & *guid*) const [inline]

Definition at line 52 of file kblight.h.

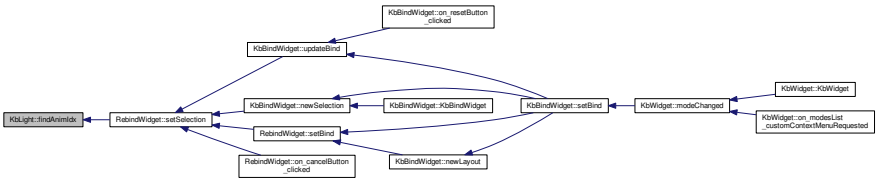
References _animList, and findAnim().

Referenced by RebindWidget::setSelection().

```
52 { return _animList.indexOf(findAnim(guid)); }
```

Here is the call graph for this function:

Here is the caller graph for this function:



12.29.4.16 void KbLight::frameDisplayed (const ColorMap & *animatedColors*, const QSet< QString > & *indicatorList*) [signal]

Definition at line 174 of file moc_kblight.cpp.

Referenced by frameUpdate().

```
175 {
176     void *_a[] = { Q_NULLPTR, const_cast<void*>(reinterpret_cast<const void*>(&_t1)), const_cast<void*>(
      reinterpret_cast<const void*>(&_t2)) };
177     QMetaObject::activate(this, &staticMetaObject, 2, _a);
178 }
```

Here is the caller graph for this function:

12.29.4.17 void KbLight::frameUpdate (QFile & *cmd*, bool *monochrome* = `false`)

Definition at line 296 of file kblight.cpp.

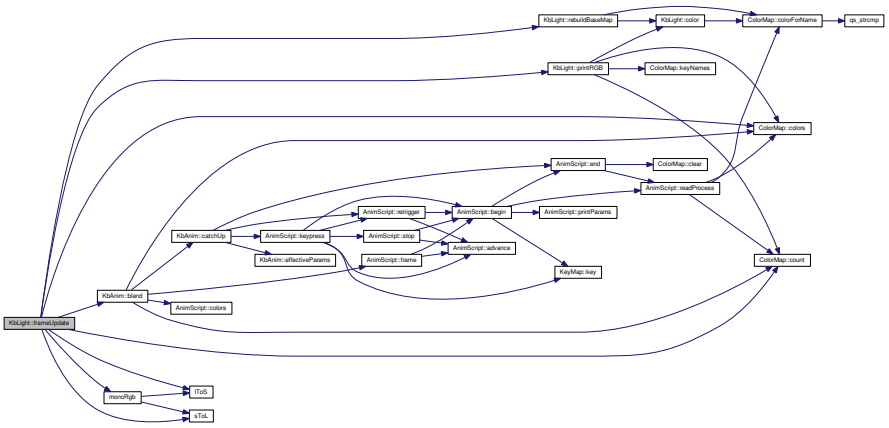
References _animList, _animMap, _colorMap, _dimming, _indicatorList, _indicatorMap, _lastFrameAnimMap, _lastFrameDimming, _lastFrameIndicatorMap, _previewAnim, KbAnim::blend(), ColorMap::colors(), ColorMap::count(), frameDisplayed(), lastFrameSignal, lToS(), monoRgb(), printRGB(), rebuildBaseMap(), and sToL().

Referenced by Kb::frameUpdate().

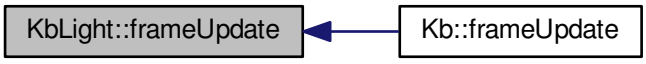
```
296                                                 {
297     rebuildBaseMap();
298     _animMap = _colorMap;
299     // Advance animations
300     quint64 timestamp = QDateTime::currentMSecsSinceEpoch();
301     foreach(KbAnim* anim, _animList)
302         anim->blend(_animMap, timestamp);
303     if(_previewAnim)
304         _previewAnim->blend(_animMap, timestamp);
305
306     // Avoid expensive processing if nothing has changed from the last frame.
307     if(_animMap == _lastFrameAnimMap && _indicatorMap ==
    _lastFrameIndicatorMap && _lastFrameDimming ==
    _dimming)
308         return;
309     _lastFrameAnimMap = _animMap;
310     _lastFrameIndicatorMap = _indicatorMap;
311     _lastFrameDimming = _dimming;
312
313     int count = _animMap.count();
314     QRgb* colors = _animMap.colors();
315     // Apply active indicators and/or perform monochrome conversion
316     if(monochrome || !_indicatorList.isEmpty()){
317         QRgb* indicators = _indicatorMap.colors();
318         for(int i = 0; i < count; i++){
319             QRgb& rgb = colors[i];
320             float r = qRed(rgb);
321             float g = qGreen(rgb);
322             float b = qBlue(rgb);
323             // Apply indicators
324             QRgb rgb2 = indicators[i];
325             if(qAlpha(rgb2) != 0){
326                 float r2 = qRed(rgb2);
327                 float g2 = qGreen(rgb2);
328                 float b2 = qBlue(rgb2);
329                 float a2 = qAlpha(rgb2) / 255.f;
330                 r = round(r2 * a2 + r * (1.f - a2));
331                 g = round(g2 * a2 + g * (1.f - a2));
332                 b = round(b2 * a2 + b * (1.f - a2));
333             }
334             // If monochrome mode is active, average the channels to get a grayscale image
335             if(monochrome)
336                 rgb = monoRgb(r, g, b);
337             else
338                 rgb = qRgb(r, g, b);
339         }
340     }
341
342     // Emit signals for the animation (only do this every 50ms - it can cause a lot of CPU usage)
343     if(timestamp >= lastFrameSignal + 50){
344         emit frameDisplayed(_animMap, _indicatorList);
345         lastFrameSignal = timestamp;
346     }
347
348     // If brightness is at 0%, turn off lighting entirely
349     if(_dimming == 3){
350         cmd.write("rgb 000000");
351         return;
352     }
353
354     float light = (3 - _dimming) / 3.f;
355     // Apply global dimming
356     if(light != 1.f || monochrome){
357         for(int i = 0; i < count; i++){
358             QRgb& rgb = colors[i];
359             // Like the monochrome conversion, this should be done in a linear colorspace
360             float r = sToL(qRed(rgb));
361             float g = sToL(qGreen(rgb));
362             float b = sToL(qBlue(rgb));
363             r *= light;
364             g *= light;
365             b *= light;
366             r = round(lToS(r));
367             g = round(lToS(g));
```

```
368             b = round(lToS(b));
369             rgb = qRgb(r, g, b);
370         }
371     }
372
373     // Apply light
374     cmd.write("rgb");
375     printRGB(cmd, _animMap);
376 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.29.4.18 bool KbLight::isStarted ( )

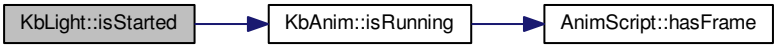
Definition at line 169 of file kblight.cpp.

References _animList, _start, and KbAnim::isRunning().

Referenced by Kb::frameUpdate().

```
169                         {
170     if(!_start)
171         return false;
172     foreach(KbAnim* animation, _animList){
173         if(!animation->isRunning())
174             return false;
175     }
176     return true;
177 }
```

Here is the call graph for this function:



Here is the caller graph for this function:

12.29.4.19 void KbLight::load (CkbSettings & *settings*)

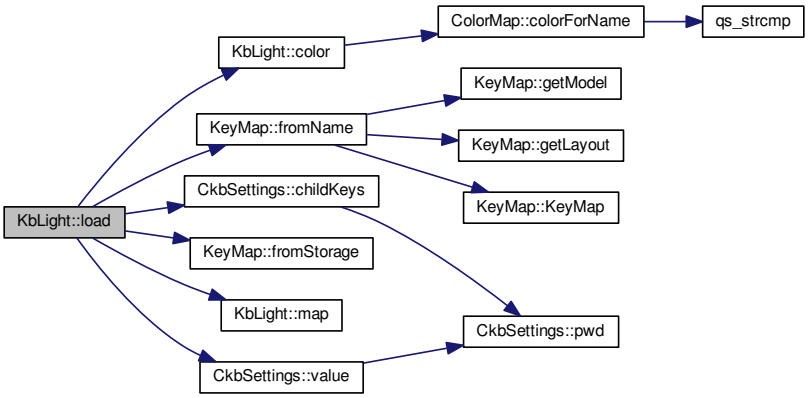
Definition at line 408 of file kblight.cpp.

References _animList, _dimming, _map, _needsMapRefresh, _needsSave, _qColorMap, CkbSettings::childKeys(), color(), didLoad(), KeyMap::fromName(), KeyMap::fromStorage(), map(), MAX_DIM, and CkbSettings::value().

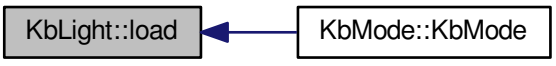
Referenced by KbMode::KbMode().

```
408                                     {
409     // Load light settings
410     _needsSave = false;
411     SGroup group(settings, "Lighting");
412     KeyMap currentMap = _map;
413     _map = KeyMap::fromName(settings.value("KeyMap").toString());
414     _dimming = settings.value("Brightness").toUInt();
415     if(_dimming > MAX_DIM)
416         _dimming = MAX_DIM;
417     // Load RGB settings
418     bool useReal = settings.value("UseRealNames").toBool();
419     {
420         SGroup group(settings, "Keys");
421         foreach(QString key, settings.childKeys()){
422             QString name = key.toLower();
423             if(!useReal)
424                 name = _map.fromStorage(name);
425             QColor color = settings.value(key).toString();
426             if(!color.isValid())
427                 color = QColor(255, 255, 255);
428             _qColorMap[name] = color.rgb();
429         }
430         _needsMapRefresh = true;
431     }
432     // Load animations
433     foreach(KbAnim* anim, _animList)
434         anim->deleteLater();
435     _animList.clear();
436     {
437         SGroup group(settings, "Animations");
438         foreach(QString anim, settings.value("List").toStringList()){
439             QUuid id = anim;
440             _animList.append(new KbAnim(this, _map, id, settings));
441         }
442     }
443     emit didLoad();
444     map(currentMap);
445 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.29.4.20 const KeyMap& KbLight::map ( ) [inline]

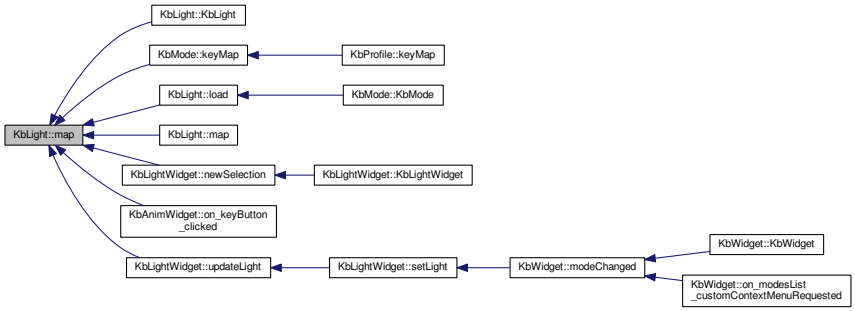
Definition at line 28 of file kblight.h.

References _map.

Referenced by KbLight(), KbMode::keyMap(), load(), map(), KbLightWidget::newSelection(), KbAnimWidget::on_keyButton_clicked(), and KbLightWidget::updateLight().

```
28 { return _map; }
```

Here is the caller graph for this function:



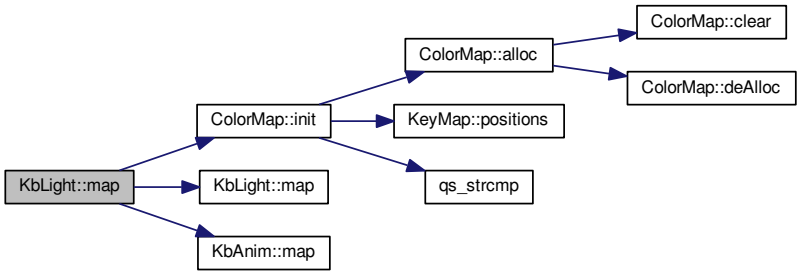
12.29.4.21 void KbLight::map (const KeyMap & *map*)

Definition at line 25 of file kblight.cpp.

References _animList, _animMap, _colorMap, _indicatorMap, _map, _needsMapRefresh, _needsSave, _qColorMap, ColorMap::init(), map(), KbAnim::map(), and updated().

```
25                                  {
26     // If any of the keys are missing from the color map, set them to white
27     QHashIterator<QString, Key> i(map);
28     while(i.hasNext()){
29         i.next();
30         const QString& key = i.key();
31         if(!_qColorMap.contains(key))
32             _qColorMap[key] = 0xFFFFFFFF;
33     }
34     // Set the new map
35     _map = map;
36     foreach(KbAnim* anim, _animList)
37         anim->map(map);
38     _colorMap.init(_map);
39     _animMap.init(_map);
40     _indicatorMap.init(_map);
41     _needsSave = _needsMapRefresh = true;
42     emit updated();
43 }
```

Here is the call graph for this function:



12.29.4.22 bool KbLight::needsSave ( ) const

Definition at line 474 of file kblight.cpp.

References _animList, _needsSave, and KbAnim::needsSave().

Referenced by KbMode::needsSave().

```
474                                {
475     if(_needsSave)
476         return true;
477     foreach(KbAnim* anim, _animList){
478         if(anim->needsSave())
479             return true;
480     }
481     return false;
482 }
```

Here is the call graph for this function:

Here is the caller graph for this function:

12.29.4.23 void KbLight::open ( )

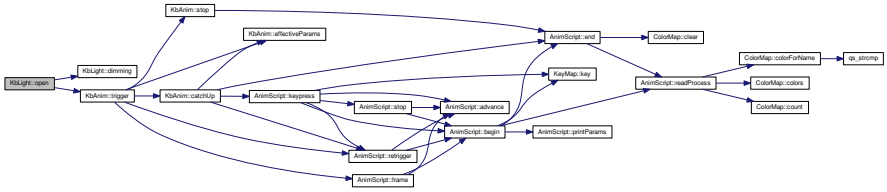
Definition at line 200 of file kblight.cpp.

References _animList, _dimming, _previewAnim, _shareDimming, _start, activeLights, dimming(), and KbAnim::trigger().

Referenced by Kb::frameUpdate().

```
200                 {
201     // Apply shared dimming if needed
202     if(_shareDimming != -1 && _shareDimming !=
    _dimming)
203         dimming(_shareDimming);
204     activeLights.insert(this);
205     if(_start)
206         return;
207     quint64 timestamp = QDateTime::currentMSecsSinceEpoch();
208     foreach(KbAnim* anim, _animList)
209         anim->trigger(timestamp);
210     if(_previewAnim)
211         _previewAnim->trigger(timestamp);
212     _start = true;
213 }
```

Here is the call graph for this function:



Here is the caller graph for this function:

12.29.4.24 void KbLight::previewAnim (const AnimScript * *base*, const QStringList & *keys*, const QMap< QString, QVariant > & *preset*)

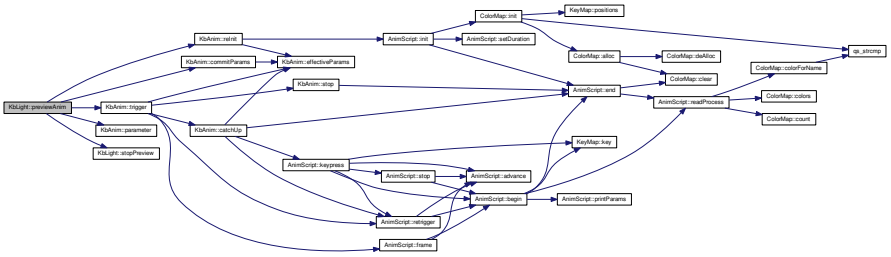
Definition at line 124 of file kblight.cpp.

References _map, _previewAnim, _start, KbAnim::commitParams(), KbAnim::parameter(), KbAnim::reInit(), stopPreview(), and KbAnim::trigger().

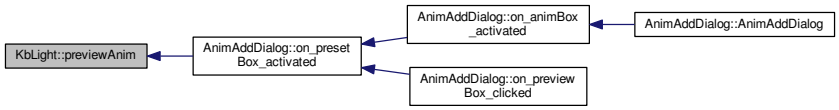
Referenced by AnimAddDialog::on_presetBox_activated().

```
124
          {
125     if(_previewAnim)
126         stopPreview();
127     quint64 timestamp = QDateTime::currentMSecsSinceEpoch();
128     // Load the new animation and set preset parameters
129     KbAnim* anim = new KbAnim(this, _map, "", keys, base);
130     QMapIterator<QString, QVariant> i(preset);
131     while(i.hasNext()){
132         i.next();
133         anim->parameter(i.key(), i.value());
134     }
135     anim->commitParams();
136     anim->reInit();
137     // Add the animation and start it
138     _previewAnim = anim;
139     anim->trigger(timestamp);
140     _start = true;
141 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.29.4.25 void KbLight::printRGB (QFile & *cmd,* const ColorMap & *animMap*) [private]

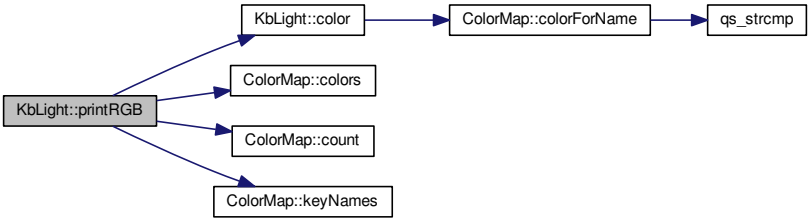
Definition at line 223 of file kblight.cpp.

References color(), ColorMap::colors(), ColorMap::count(), and ColorMap::keyNames().

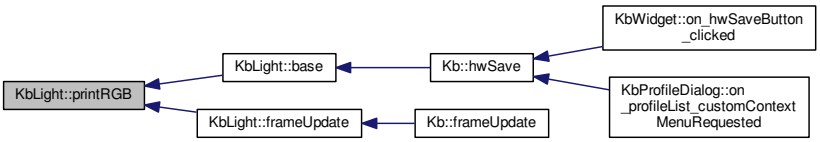
Referenced by base(), and frameUpdate().

```
223                                                          {
224     int count = animMap.count();
225     const char* const* names = animMap.keyNames();
226     const QRgb* colors = animMap.colors();
227     // Print each color and the corresponding RGB value
228     for(int i = 0; i < count; i++){
229         cmd.write(" ");
230         cmd.write(names[i]);
231         char output[8];
232         QRgb color = colors[i];
233         snprintf(output, sizeof(output), ":%02x%02x%02x", qRed(color), qGreen(color), qBlue(color));
234         cmd.write(output);
235     }
236 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.29.4.26 void KbLight::rebuildBaseMap ( ) [private]

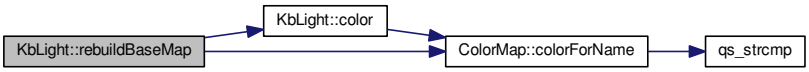
Definition at line 238 of file kblight.cpp.

References _colorMap, _needsMapRefresh, _qColorMap, color(), and ColorMap::colorForName().

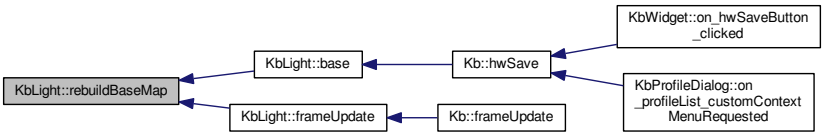
Referenced by base(), and frameUpdate().

```
238                             {
239     if(!_needsMapRefresh)
240         return;
241     _needsMapRefresh = false;
242     // Copy RGB values from QColorMap to ColorMap
243     QColorMapIterator i(_qColorMap);
244     while(i.hasNext()){
245         i.next();
246         QByteArray rawName = i.key().toLatin1();
247         QRgb color = i.value();
248         QRgb* rawColor = _colorMap.colorForName(rawName.data());
249         if(rawColor)
250             *rawColor = color;
251     }
252 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.29.4.27 void KbLight::resetIndicators ( )

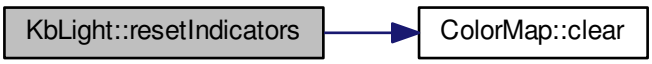
Definition at line 254 of file kblight.cpp.

References _indicatorList, _indicatorMap, and ColorMap::clear().

Referenced by KbPerf::applyIndicators().

```
254                                {
255     _indicatorMap.clear();
256     _indicatorList.clear();
257 }
```

Here is the call graph for this function:



Here is the caller graph for this function:

12.29.4.28 void KbLight::restartAnimation ( )

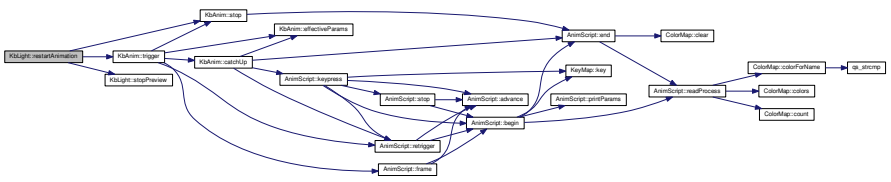
Definition at line 179 of file kblight.cpp.

References _animList, _start, KbAnim::stop(), stopPreview(), and KbAnim::trigger().

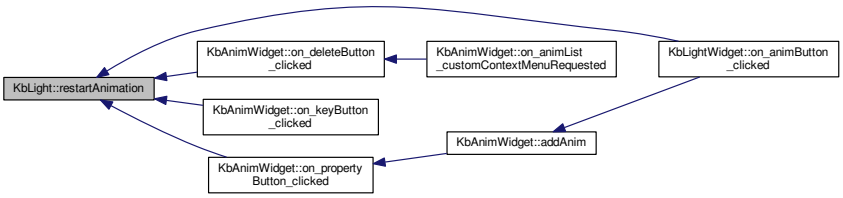
Referenced by KbLightWidget::on_animButton_clicked(), KbAnimWidget::on_deleteButton_clicked(), KbAnimWidget::on_keyButton_clicked(), and KbAnimWidget::on_propertyButton_clicked().

```
179                                 {
180     quint64 timestamp = QDateTime::currentMSecsSinceEpoch();
181     foreach(KbAnim* anim, _animList){
182         anim->stop();
183         anim->trigger(timestamp);
184     }
185     stopPreview();
186     _start = true;
187 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.29.4.29 void KbLight::save (CkbSettings & *settings*)

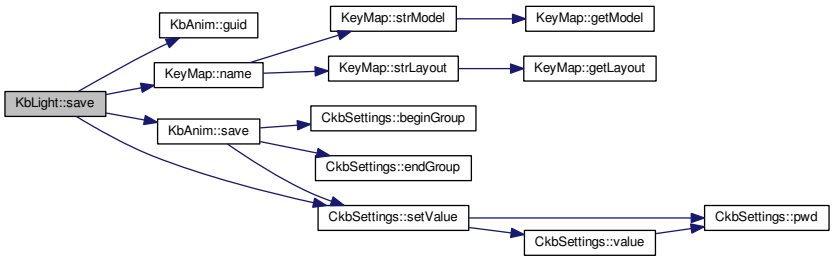
Definition at line 447 of file kblight.cpp.

References _animList, _dimming, _map, _needsSave, _qColorMap, KbAnim::guid(), KeyMap::name(), KbAnim::save(), and CkbSettings::setValue().

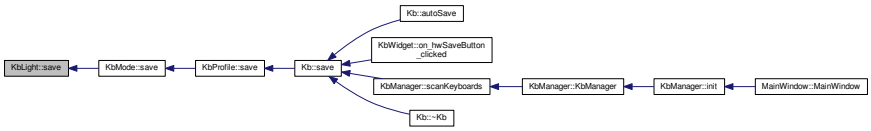
Referenced by KbMode::save().

```
447                                     {
448     _needsSave = false;
449     SGroup group(settings, "Lighting");
450     settings.setValue("KeyMap", _map.name());
451     settings.setValue("Brightness", _dimming);
452     settings.setValue("UseRealNames", true);
453     {
454         // Save RGB settings
455         SGroup group(settings, "Keys");
456         QMutableColorMapIterator i(_qColorMap);
457         while(i.hasNext()){
458             i.next();
459             settings.setValue(i.key(), QColor(i.value()).name());
460         }
461     }
462     {
463         // Save animations
464         SGroup group(settings, "Animations");
465         QStringList aList;
466         foreach(KbAnim* anim, _animList){
467             aList << anim->guid().toString().toUpper();
468             anim->save(settings);
469         }
470         settings.setValue("List", aList);
471     }
472 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.29.4.30 void KbLight::setIndicator (const char * *name*, QRgb *argb*)

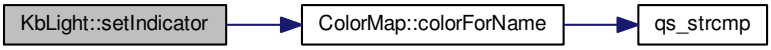
Definition at line 259 of file kblight.cpp.

References _indicatorList, _indicatorMap, and ColorMap::colorForName().

Referenced by KbPerf::lightIndicator().

```
259                                                     {
260     QRgb* dest = _indicatorMap.colorForName(name);
261     if(dest){
262         *dest = argb;
263         _indicatorList.insert(name);
264     }
265 }
```

Here is the call graph for this function:



Here is the caller graph for this function:

12.29.4.31 int KbLight::shareDimming ( ) [static]

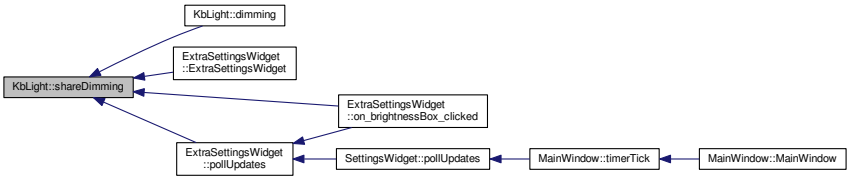
Definition at line 78 of file kblight.cpp.

References _shareDimming.

Referenced by dimming(), ExtraSettingsWidget::ExtraSettingsWidget(), ExtraSettingsWidget::on_brightnessBox_clicked(), and ExtraSettingsWidget::pollUpdates().

```
78                          {
79     return _shareDimming;
80 }
```

Here is the caller graph for this function:



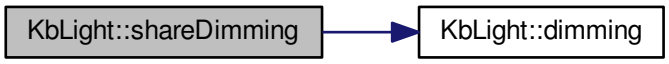
12.29.4.32 void KbLight::shareDimming (int *newShareDimming*) [static]

Definition at line 82 of file kblight.cpp.

References _shareDimming, activeLights, and dimming().

```
82                                              {
83     if(_shareDimming == newShareDimming)
84         return;
85     _shareDimming = newShareDimming;
86     if(newShareDimming != -1){
87         foreach(KbLight* light, activeLights)
88             light->dimming(newShareDimming);
89     }
90 }
```

Here is the call graph for this function:



12.29.4.33 void KbLight::stopPreview ( )

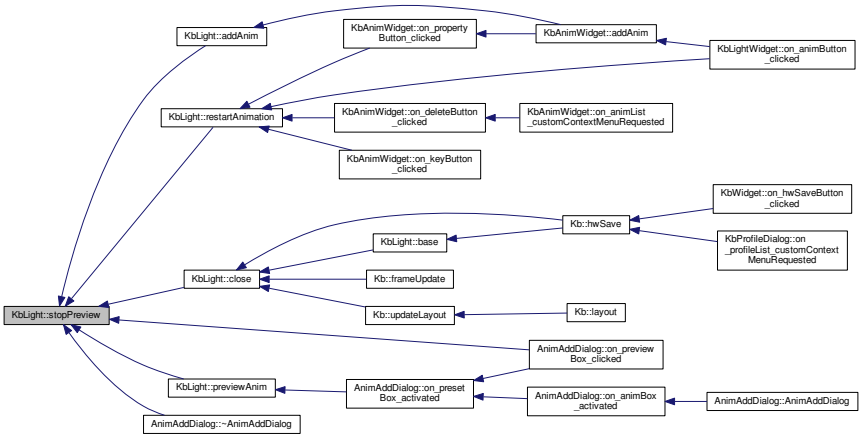
Definition at line 143 of file kblight.cpp.

References _previewAnim.

Referenced by addAnim(), close(), AnimAddDialog::on_previewBox_clicked(), previewAnim(), restartAnimation(), and AnimAddDialog::~AnimAddDialog().

```
143                         {
144     delete _previewAnim;
145     _previewAnim = 0;
146 }
```

Here is the caller graph for this function:



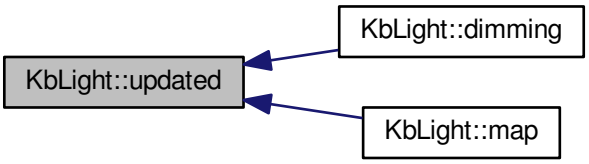
12.29.4.34 void KbLight::updated ( ) [signal]

Definition at line 168 of file moc_kblight.cpp.

Referenced by dimming(), and map().

```
169 {
170     QMetaObject::activate(this, &staticMetaObject, 1, Q_NULLPTR);
171 }
```

Here is the caller graph for this function:



12.29.5 Field Documentation

12.29.5.1 AnimList KbLight::_animList [private]

Definition at line 89 of file kblight.h.

Referenced by addAnim(), animKeypress(), animList(), close(), duplicateAnim(), findAnim(), findAnimIdx(), frameUpdate(), isStarted(), KbLight(), load(), map(), needsSave(), open(), restartAnimation(), and save().

12.29.5.2 ColorMap KbLight::_animMap [private]

Definition at line 93 of file kblight.h.

Referenced by base(), frameUpdate(), and map().

12.29.5.3 ColorMap KbLight::_colorMap [private]

Definition at line 93 of file kblight.h.

Referenced by base(), color(), frameUpdate(), map(), and rebuildBaseMap().

12.29.5.4 int KbLight::_dimming [private]

Definition at line 96 of file kblight.h.

Referenced by base(), dimming(), frameUpdate(), load(), open(), and save().

12.29.5.5 QSet<QString> KbLight::_indicatorList [private]

Definition at line 94 of file kblight.h.

Referenced by frameUpdate(), resetIndicators(), and setIndicator().

12.29.5.6 ColorMap KbLight::_indicatorMap [private]

Definition at line 93 of file kblight.h.

Referenced by frameUpdate(), map(), resetIndicators(), and setIndicator().

12.29.5.7 ColorMap KbLight::_lastFrameAnimMap [private]

Definition at line 93 of file kblight.h.

Referenced by frameUpdate().

12.29.5.8 int KbLight::_lastFrameDimming [private]

Definition at line 96 of file kblight.h.

Referenced by frameUpdate().

12.29.5.9 ColorMap KbLight::_lastFrameIndicatorMap [private]

Definition at line 93 of file kblight.h.

Referenced by frameUpdate().

12.29.5.10 KeyMap KbLight::_map [private]

Definition at line 91 of file kblight.h.

Referenced by addAnim(), duplicateAnim(), load(), map(), previewAnim(), and save().

12.29.5.11 bool KbLight::_needsMapRefresh [private]

Definition at line 98 of file kblight.h.

Referenced by color(), load(), map(), and rebuildBaseMap().

12.29.5.12 bool KbLight::_needsSave [private]

Definition at line 98 of file kblight.h.

Referenced by addAnim(), animList(), color(), dimming(), duplicateAnim(), load(), map(), needsSave(), and save().

12.29.5.13 KbAnim* KbLight::_previewAnim [private]

Definition at line 90 of file kblight.h.

Referenced by animKeypress(), frameUpdate(), open(), previewAnim(), and stopPreview().

12.29.5.14 QColorMap KbLight::_qColorMap [private]

Definition at line 92 of file kblight.h.

Referenced by color(), colorMap(), load(), map(), rebuildBaseMap(), and save().

12.29.5.15 bool KbLight::_start [private]

Definition at line 97 of file kblight.h.

Referenced by addAnim(), close(), duplicateAnim(), isStarted(), open(), previewAnim(), and restartAnimation().

12.29.5.16 quint64 KbLight::lastFrameSignal [private]

Definition at line 95 of file kblight.h.

Referenced by frameUpdate().

12.29.5.17 const int KbLight::MAX_DIM = 3 [static]

Definition at line 38 of file kblight.h.

Referenced by base(), ExtraSettingsWidget::ExtraSettingsWidget(), KeyAction::keyEvent(), and load().

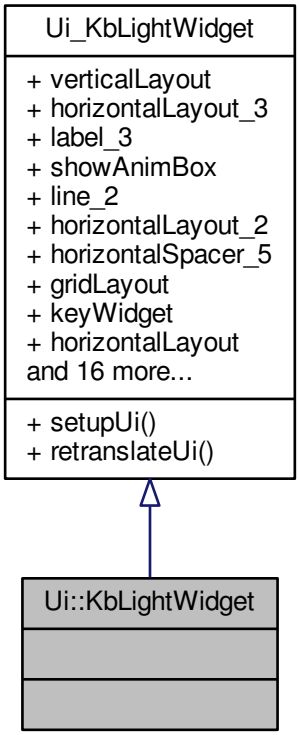
The documentation for this class was generated from the following files:

- src/ckb/kblight.h
- src/ckb/kblight.cpp
- src/ckb/release/moc_kblight.cpp

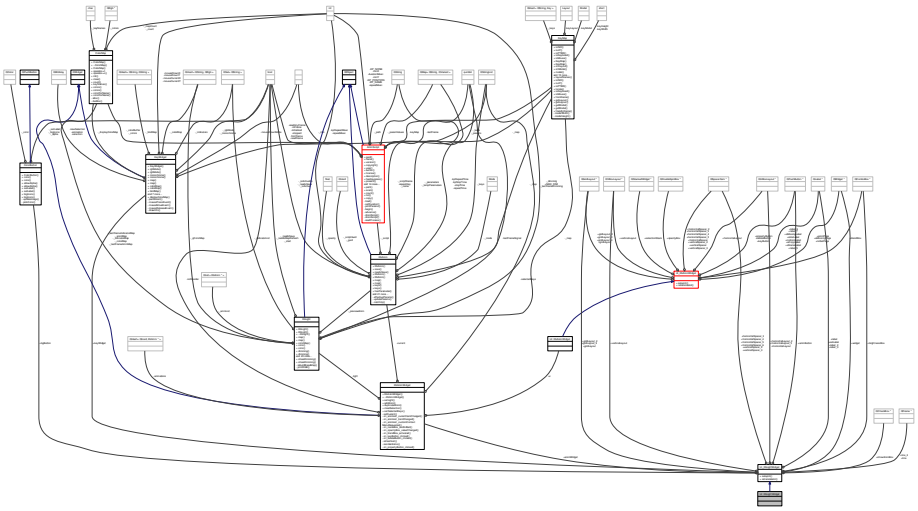
12.30 Ui::KbLightWidget Class Reference

`#include <src/ckb/ui_kblightwidget.h>`

Inheritance diagram for Ui::KbLightWidget:



Collaboration diagram for Ui::KbLightWidget:



Public Member Functions

- void setupUi (QWidget *KbLightWidget)

- void retranslateUi (QWidget *KbLightWidget)

Data Fields

- QVBoxLayout * verticalLayout
- QHBoxLayout * horizontalLayout_3
- QLabel * label_3
- QCheckBox * showAnimBox
- QFrame * line_2
- QHBoxLayout * horizontalLayout_2
- QSpacerItem * horizontalSpacer_5
- QGridLayout * gridLayout
- KeyWidget * keyWidget
- QHBoxLayout * horizontalLayout
- ColorButton * bgButton
- QLabel * selLabel
- QPushButton * animButton
- QSpacerItem * horizontalSpacer_6
- QSpacerItem * verticalSpacer_2
- QGridLayout * gridLayout_2
- QComboBox * brightnessBox
- QLabel * label
- QSpacerItem * horizontalSpacer_3
- QSpacerItem * horizontalSpacer_4
- QSpacerItem * verticalSpacer_3
- QLabel * label_2
- QFrame * line
- QWidget * widget
- QGridLayout * gridLayout_4
- KbAnimWidget * animWidget

12.30.1 Detailed Description

Definition at line 266 of file ui_kblightwidget.h.

12.30.2 Member Function Documentation

12.30.2.1 void Ui_KbLightWidget::retranslateUi (QWidget * *KbLightWidget*) [inline],[inherited]

Definition at line 241 of file ui_kblightwidget.h.

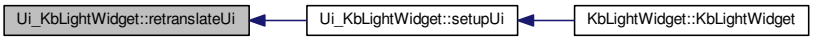
References Ui_KbLightWidget::animButton, Ui_KbLightWidget::bgButton, Ui_KbLightWidget::brightnessBox, Ui_KbLightWidget::label, Ui_KbLightWidget::label_2, Ui_KbLightWidget::label_3, Ui_KbLightWidget::selLabel, and Ui_KbLightWidget::showAnimBox.

Referenced by Ui_KbLightWidget::setupUi().

```
242     {
243         KbLightWidget->setWindowTitle(QApplication::translate("KbLightWidget", "Form", Q_NULLPTR));
244         label_3->setText(QApplication::translate("KbLightWidget", "Colors", Q_NULLPTR));
245 #ifndef QT_NO_TOOLTIP
246         showAnimBox->setToolTip(QApplication::translate("KbLightWidget", "If enabled, the same
       colors will be displayed here as on the keyboard. If disabled, the base colors will be shown.", Q_NULLPTR));
247 #endif // QT_NO_TOOLTIP
248         showAnimBox->setText(QApplication::translate("KbLightWidget", "Show animated", Q_NULLPTR
     ));
249         bgButton->setText(QApplication::translate("KbLightWidget", "Change color...", Q_NULLPTR));
250         selLabel->setText(QApplication::translate("KbLightWidget", "Click to select", Q_NULLPTR));
251         animButton->setText(QApplication::translate("KbLightWidget", "New animation...",
```

```
    Q_NULLPTR));
252         brightnessBox->clear();
253         brightnessBox->insertItems(0, QStringList()
254          << QApplication::translate("KbLightWidget", "100%", Q_NULLPTR)
255          << QApplication::translate("KbLightWidget", "67%", Q_NULLPTR)
256          << QApplication::translate("KbLightWidget", "33%", Q_NULLPTR)
257          << QApplication::translate("KbLightWidget", "Off", Q_NULLPTR)
258         );
259         label->setText(QApplication::translate("KbLightWidget", "\342\230\274", Q_NULLPTR));
260         label_2->setText(QApplication::translate("KbLightWidget", "Animations", Q_NULLPTR));
261     } // retranslateUi
```

Here is the caller graph for this function:



12.30.2.2 void Ui_KbLightWidget::setupUi (QWidget * *KbLightWidget*) [inline], [inherited]

Definition at line 63 of file ui_kblightwidget.h.

References Ui_KbLightWidget::animButton, Ui_KbLightWidget::animWidget, Ui_KbLightWidget::bgButton, Ui_KbLightWidget::brightnessBox, Ui_KbLightWidget::gridLayout, Ui_KbLightWidget::gridLayout_2, Ui_KbLightWidget::gridLayout_4, Ui_KbLightWidget::horizontalLayout, Ui_KbLightWidget::horizontalLayout_2, Ui_KbLightWidget::horizontalLayout_3, Ui_KbLightWidget::horizontalSpacer_3, Ui_KbLightWidget::horizontalSpacer_4, Ui_KbLightWidget::horizontalSpacer_5, Ui_KbLightWidget::horizontalSpacer_6, Ui_KbLightWidget::keyWidget, Ui_KbLightWidget::label, Ui_KbLightWidget::label_2, Ui_KbLightWidget::label_3, Ui_KbLightWidget::line, Ui_KbLightWidget::line_2, Ui_KbLightWidget::retranslateUi(), Ui_KbLightWidget::selLabel, Ui_KbLightWidget::showAnimBox, Ui_KbLightWidget::verticalLayout, Ui_KbLightWidget::verticalSpacer_2, Ui_KbLightWidget::verticalSpacer_3, and Ui_KbLightWidget::widget.

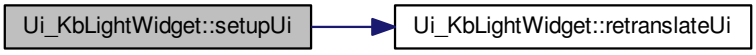
Referenced by KbLightWidget::KbLightWidget().

```
64     {
65         if (KbLightWidget->objectName().isEmpty())
66             KbLightWidget->setObjectName(QStringLiteral("KbLightWidget"));
67         KbLightWidget->resize(632, 540);
68         KbLightWidget->setFocusPolicy(Qt::ClickFocus);
69         verticalLayout = new QVBoxLayout(KbLightWidget);
70         verticalLayout->setObjectName(QStringLiteral("verticalLayout"));
71         verticalLayout->setContentsMargins(0, 0, 0, -1);
72         horizontalLayout_3 = new QHBoxLayout();
73         horizontalLayout_3->setObjectName(QStringLiteral("horizontalLayout_3"));
74         label_3 = new QLabel(KbLightWidget);
75         label_3->setObjectName(QStringLiteral("label_3"));
76         QSizePolicy sizePolicy(QSizePolicy::Expanding, QSizePolicy::Preferred);
77         sizePolicy.setHorizontalStretch(0);
78         sizePolicy.setVerticalStretch(0);
79         sizePolicy.setHeightForWidth(label_3->sizePolicy().hasHeightForWidth());
80         label_3->setSizePolicy(sizePolicy);
81         QFont font;
82         font.setBold(true);
83         font.setWeight(75);
84         label_3->setFont(font);
85
86         horizontalLayout_3->addWidget(label_3);
87
88         showAnimBox = new QCheckBox(KbLightWidget);
89         showAnimBox->setObjectName(QStringLiteral("showAnimBox"));
90         showAnimBox->setChecked(true);
91
92         horizontalLayout_3->addWidget(showAnimBox);
93
94
95         verticalLayout->addLayout(horizontalLayout_3);
96
97         line_2 = new QFrame(KbLightWidget);
98         line_2->setObjectName(QStringLiteral("line_2"));
99         line_2->setFrameShape(QFrame::HLine);
```

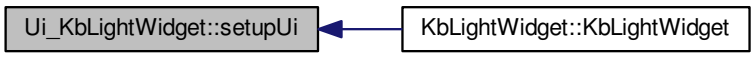
```
100         line_2->setFrameShadow(QFrame::Sunken);
101 
102         verticalLayout->addWidget(line_2);
103 
104         horizontalLayout_2 = new QHBoxLayout();
105         horizontalLayout_2->setObjectName(QStringLiteral("horizontalLayout_2"));
106         horizontalSpacer_5 = new QSpacerItem(40, 20, QSizePolicy::Expanding,
    QSizePolicy::Minimum);
107 
108         horizontalLayout_2->addItem(horizontalSpacer_5);
109 
110         gridLayout = new QGridLayout();
111         gridLayout->setObjectName(QStringLiteral("gridLayout"));
112         keyWidget = new KeyWidget(KbLightWidget);
113         keyWidget->setObjectName(QStringLiteral("keyWidget"));
114         QSizePolicy sizePolicy1(QSizePolicy::Expanding, QSizePolicy::Expanding);
115         sizePolicy1.setHorizontalStretch(0);
116         sizePolicy1.setVerticalStretch(0);
117         sizePolicy1.setHeightForWidth(keyWidget->sizePolicy().hasHeightForWidth());
118         keyWidget->setSizePolicy(sizePolicy1);
119         keyWidget->setMinimumSize(QSize(100, 100));
120         keyWidget->setFocusPolicy(Qt::ClickFocus);
121 
122         gridLayout->addWidget(keyWidget, 0, 0, 1, 3);
123 
124         horizontalLayout = new QHBoxLayout();
125         horizontalLayout->setObjectName(QStringLiteral("horizontalLayout"));
126         bgButton = new ColorButton(KbLightWidget);
127         bgButton->setObjectName(QStringLiteral("bgButton"));
128         QSizePolicy sizePolicy2(QSizePolicy::Fixed, QSizePolicy::Fixed);
129         sizePolicy2.setHorizontalStretch(0);
130         sizePolicy2.setVerticalStretch(0);
131         sizePolicy2.setHeightForWidth(bgButton->sizePolicy().hasHeightForWidth());
132         bgButton->setSizePolicy(sizePolicy2);
133         bgButton->setMinimumSize(QSize(140, 0));
134         bgButton->setMaximumSize(QSize(140, 16777215));
135 
136         horizontalLayout->addWidget(bgButton);
137 
138         selLabel = new QLabel(KbLightWidget);
139         selLabel->setObjectName(QStringLiteral("selLabel"));
140         sizePolicy.setHeightForWidth(selLabel->sizePolicy().hasHeightForWidth());
141         selLabel->setSizePolicy(sizePolicy);
142         selLabel->setAlignment(Qt::AlignCenter);
143 
144         horizontalLayout->addWidget(selLabel);
145 
146         animButton = new QPushButton(KbLightWidget);
147         animButton->setObjectName(QStringLiteral("animButton"));
148         sizePolicy2.setHeightForWidth(animButton->sizePolicy().hasHeightForWidth());
149         animButton->setSizePolicy(sizePolicy2);
150         animButton->setMinimumSize(QSize(140, 0));
151         animButton->setMaximumSize(QSize(140, 16777215));
152 
153         horizontalLayout->addWidget(animButton);
154 
155 
156         gridLayout->addLayout(horizontalLayout, 1, 0, 1, 3);
157 
158 
159         horizontalLayout_2->addLayout(gridLayout);
160 
161         horizontalSpacer_6 = new QSpacerItem(40, 20, QSizePolicy::Expanding,
    QSizePolicy::Minimum);
162 
163         horizontalLayout_2->addItem(horizontalSpacer_6);
164 
165 
166         verticalLayout->addLayout(horizontalLayout_2);
167 
168         verticalSpacer_2 = new QSpacerItem(20, 40, QSizePolicy::Minimum,
    QSizePolicy::Expanding);
169 
170         verticalLayout->addItem(verticalSpacer_2);
171 
172         gridLayout_2 = new QGridLayout();
173         gridLayout_2->setObjectName(QStringLiteral("gridLayout_2"));
174         brightnessBox = new QComboBox(KbLightWidget);
175         brightnessBox->setObjectName(QStringLiteral("brightnessBox"));
176         sizePolicy2.setHeightForWidth(brightnessBox->sizePolicy().hasHeightForWidth());
177         brightnessBox->setSizePolicy(sizePolicy2);
178 
179         gridLayout_2->addWidget(brightnessBox, 1, 3, 1, 1);
180 
181         label = new QLabel(KbLightWidget);
182         label->setObjectName(QStringLiteral("label"));
183         QFont font1;
```

```
184         font1.setPointSize(20);
185         label->setFont(font1);
186         label->setAlignment(Qt::AlignRight|Qt::AlignTrailing|Qt::AlignVCenter);
187 
188         gridLayout_2->addWidget(label, 1, 2, 1, 1);
189 
190         horizontalSpacer_3 = new QSpacerItem(0, 0, QSizePolicy::Expanding,
    QSizePolicy::Minimum);
191 
192         gridLayout_2->addItem(horizontalSpacer_3, 0, 4, 1, 1);
193 
194         horizontalSpacer_4 = new QSpacerItem(0, 0, QSizePolicy::Expanding,
    QSizePolicy::Minimum);
195 
196         gridLayout_2->addItem(horizontalSpacer_4, 0, 2, 1, 1);
197 
198 
199         verticalLayout->addLayout(gridLayout_2);
200 
201         verticalSpacer_3 = new QSpacerItem(20, 40, QSizePolicy::Minimum,
    QSizePolicy::Expanding);
202 
203         verticalLayout->addItem(verticalSpacer_3);
204 
205         label_2 = new QLabel(KbLightWidget);
206         label_2->setObjectName(QStringLiteral("label_2"));
207         label_2->setFont(font);
208 
209         verticalLayout->addWidget(label_2);
210 
211         line = new QFrame(KbLightWidget);
212         line->setObjectName(QStringLiteral("line"));
213         line->setFrameShape(QFrame::HLine);
214         line->setFrameShadow(QFrame::Sunken);
215 
216         verticalLayout->addWidget(line);
217 
218         widget = new QWidget(KbLightWidget);
219         widget->setObjectName(QStringLiteral("widget"));
220         widget->setMinimumSize(QSize(0, 170));
221         gridLayout_4 = new QGridLayout(widget);
222         gridLayout_4->setObjectName(QStringLiteral("gridLayout_4"));
223         gridLayout_4->setContentsMargins(-1, 0, -1, 0);
224         animWidget = new KbAnimWidget(widget);
225         animWidget->setObjectName(QStringLiteral("animWidget"));
226 
227         gridLayout_4->addWidget(animWidget, 0, 0, 1, 1);
228 
229 
230         verticalLayout->addWidget(widget);
231 
232         QWidget::setTabOrder(showAnimBox, bgButton);
233         QWidget::setTabOrder(bgButton, animButton);
234         QWidget::setTabOrder(animButton, brightnessBox);
235 
236         retranslateUi(KbLightWidget);
237 
238         QMetaObject::connectSlotsByName(KbLightWidget);
239     } // setupUi
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.30.3 Field Documentation

12.30.3.1 QPushButton* Ui_KbLightWidget::animButton [inherited]

Definition at line 48 of file ui_kblightwidget.h.

Referenced by KbLightWidget::KbLightWidget(), Ui_KbLightWidget::retranslateUi(), and Ui_KbLightWidget::setupUi().

12.30.3.2 KbAnimWidget* Ui_KbLightWidget::animWidget [inherited]

Definition at line 61 of file ui_kblightwidget.h.

Referenced by KbLightWidget::KbLightWidget(), KbLightWidget::newSelection(), KbLightWidget::on_animButton_clicked(), KbLightWidget::setLight(), and Ui_KbLightWidget::setupUi().

12.30.3.3 ColorButton* Ui_KbLightWidget::bgButton [inherited]

Definition at line 46 of file ui_kblightwidget.h.

Referenced by KbLightWidget::KbLightWidget(), KbLightWidget::newSelection(), Ui_KbLightWidget::retranslateUi(), and Ui_KbLightWidget::setupUi().

12.30.3.4 QComboBox* Ui_KbLightWidget::brightnessBox [inherited]

Definition at line 52 of file ui_kblightwidget.h.

Referenced by Ui_KbLightWidget::retranslateUi(), Ui_KbLightWidget::setupUi(), and KbLightWidget::updateLight().

12.30.3.5 QGridLayout* Ui_KbLightWidget::gridLayout [inherited]

Definition at line 43 of file ui_kblightwidget.h.

Referenced by Ui_KbLightWidget::setupUi().

12.30.3.6 QGridLayout* Ui_KbLightWidget::gridLayout_2 [inherited]

Definition at line 51 of file ui_kblightwidget.h.

Referenced by Ui_KbLightWidget::setupUi().

12.30.3.7 QGridLayout* Ui_KbLightWidget::gridLayout_4 [inherited]

Definition at line 60 of file ui_kblightwidget.h.

Referenced by Ui_KbLightWidget::setupUi().

12.30.3.8 QHBoxLayout* Ui_KbLightWidget::horizontalLayout [inherited]

Definition at line 45 of file ui_kblightwidget.h.

Referenced by Ui_KbLightWidget::setupUi().

12.30.3.9 QHBoxLayout* Ui_KbLightWidget::horizontalLayout_2 [inherited]

Definition at line 41 of file ui_kblightwidget.h.

Referenced by Ui_KbLightWidget::setupUi().

12.30.3.10 QHBoxLayout* Ui_KbLightWidget::horizontalLayout_3 [inherited]

Definition at line 37 of file ui_kblightwidget.h.

Referenced by Ui_KbLightWidget::setupUi().

12.30.3.11 QSpacerItem* Ui_KbLightWidget::horizontalSpacer_3 [inherited]

Definition at line 54 of file ui_kblightwidget.h.

Referenced by Ui_KbLightWidget::setupUi().

12.30.3.12 QSpacerItem* Ui_KbLightWidget::horizontalSpacer_4 [inherited]

Definition at line 55 of file ui_kblightwidget.h.

Referenced by Ui_KbLightWidget::setupUi().

12.30.3.13 QSpacerItem* Ui_KbLightWidget::horizontalSpacer_5 [inherited]

Definition at line 42 of file ui_kblightwidget.h.

Referenced by Ui_KbLightWidget::setupUi().

12.30.3.14 QSpacerItem* Ui_KbLightWidget::horizontalSpacer_6 [inherited]

Definition at line 49 of file ui_kblightwidget.h.

Referenced by Ui_KbLightWidget::setupUi().

12.30.3.15 KeyWidget* Ui_KbLightWidget::keyWidget [inherited]

Definition at line 44 of file ui_kblightwidget.h.

Referenced by KbLightWidget::changeAnim(), KbLightWidget::changeAnimKeys(), KbLightWidget::changeColor(), KbLightWidget::KbLightWidget(), KbLightWidget::on_animButton_clicked(), KbLightWidget::on_bgButton_clicked(), KbLightWidget::on_showAnimBox_clicked(), KbLightWidget::setLight(), KbLightWidget::setMonochrome(), Ui_KbLightWidget::setupUi(), KbLightWidget::toggleSidelight(), and KbLightWidget::updateLight().

12.30.3.16 QLabel* Ui_KbLightWidget::label [inherited]

Definition at line 53 of file ui_kblightwidget.h.

Referenced by Ui_KbLightWidget::retranslateUi(), and Ui_KbLightWidget::setupUi().

12.30.3.17 QLabel* Ui_KbLightWidget::label_2 [inherited]

Definition at line 57 of file ui_kblightwidget.h.

Referenced by Ui_KbLightWidget::retranslateUi(), and Ui_KbLightWidget::setupUi().

12.30.3.18 QLabel* Ui_KbLightWidget::label_3 [inherited]

Definition at line 38 of file ui_kblightwidget.h.

Referenced by Ui_KbLightWidget::retranslateUi(), and Ui_KbLightWidget::setupUi().

12.30.3.19 QFrame* Ui_KbLightWidget::line [inherited]

Definition at line 58 of file ui_kblightwidget.h.

Referenced by Ui_KbLightWidget::setupUi().

12.30.3.20 QFrame* Ui_KbLightWidget::line_2 [inherited]

Definition at line 40 of file ui_kblightwidget.h.

Referenced by Ui_KbLightWidget::setupUi().

12.30.3.21 QLabel* Ui_KbLightWidget::selLabel [inherited]

Definition at line 47 of file ui_kblightwidget.h.

Referenced by KbLightWidget::newSelection(), Ui_KbLightWidget::retranslateUi(), and Ui_KbLightWidget::setupUi().

12.30.3.22 QCheckBox* Ui_KbLightWidget::showAnimBox [inherited]

Definition at line 39 of file ui_kblightwidget.h.

Referenced by KbLightWidget::KbLightWidget(), Ui_KbLightWidget::retranslateUi(), KbLightWidget::setLight(), and Ui_KbLightWidget::setupUi().

12.30.3.23 QVBoxLayout* Ui_KbLightWidget::verticalLayout [inherited]

Definition at line 36 of file ui_kblightwidget.h.

Referenced by Ui_KbLightWidget::setupUi().

12.30.3.24 QSpacerItem* Ui_KbLightWidget::verticalSpacer_2 [inherited]

Definition at line 50 of file ui_kblightwidget.h.

Referenced by Ui_KbLightWidget::setupUi().

12.30.3.25 QSpacerItem* Ui_KbLightWidget::verticalSpacer_3 [inherited]

Definition at line 56 of file ui_kblightwidget.h.

Referenced by Ui_KbLightWidget::setupUi().

12.30.3.26 QWidget* Ui_KbLightWidget::widget [inherited]

Definition at line 59 of file ui_kblightwidget.h.

Referenced by Ui_KbLightWidget::setupUi().

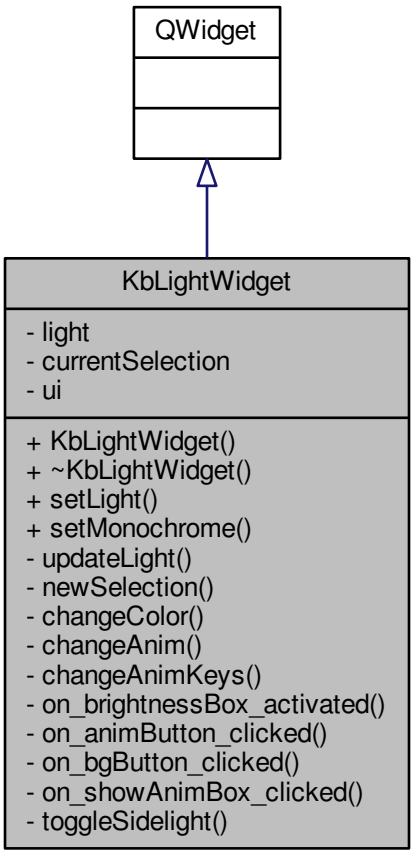
The documentation for this class was generated from the following file:

- src/ckb/ui_kblightwidget.h

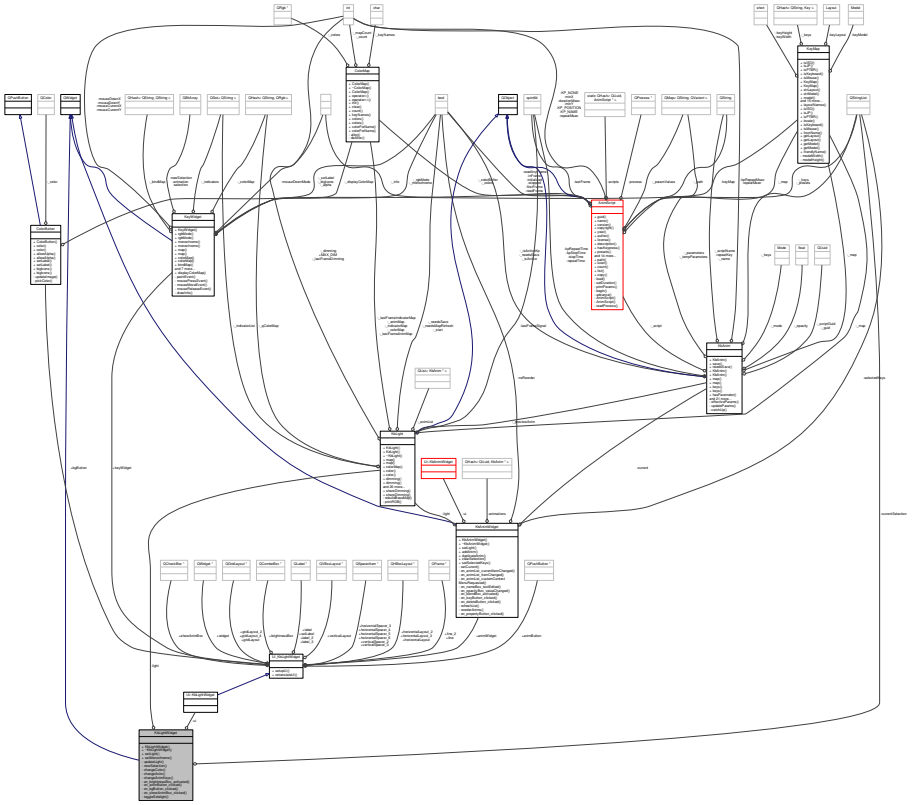
12.31 KbLightWidget Class Reference

`#include <src/ckb/kblightwidget.h>`

Inheritance diagram for KbLightWidget:



Collaboration diagram for KbLightWidget:



Public Member Functions

- KbLightWidget (QWidget *parent=0)
- ~KbLightWidget ()
- void setLight (KbLight *newLight)
- void setMonochrome ()

Private Slots

- void updateLight ()
- void newSelection (QStringList selection)
- void changeColor (QColor newColor)
- void changeAnim (KbAnim *newAnim)
- void changeAnimKeys (QStringList keys)
- void on_brightnessBox_activated (int index)
- void on_animButton_clicked ()
- void on_bgButton_clicked ()
- void on_showAnimBox_clicked (bool checked)
- void toggleSidelight ()

Private Attributes

- KbLight * light
- QStringList currentSelection
- Ui::KbLightWidget * ui

12.31.1 Detailed Description

Definition at line 15 of file kblightwidget.h.

12.31.2 Constructor & Destructor Documentation

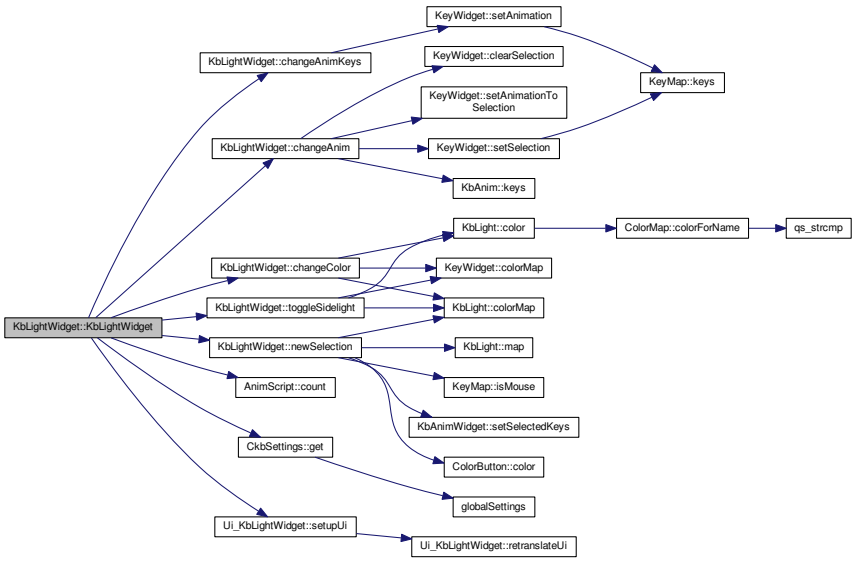
12.31.2.1 KbLightWidget::KbLightWidget (QWidget * *parent* = 0) [explicit]

Definition at line 6 of file kblightwidget.cpp.

References Ui_KbLightWidget::animButton, Ui_KbLightWidget::animWidget, Ui_KbLightWidget::bgButton, changeAnim(), changeAnimKeys(), changeColor(), AnimScript::count(), CkbSettings::get(), Ui_KbLightWidget::keyWidget, newSelection(), Ui_KbLightWidget::setupUi(), Ui_KbLightWidget::showAnimBox, toggleSidelight(), and ui.

```
6                                                :
7     QWidget(parent), light(0),
8     ui(new Ui::KbLightWidget)
9 {
10     ui->setupUi(this);
11     if(AnimScript::count() == 0)
12         ui->animButton->setVisible(false);
13
14     connect(ui->bgButton, SIGNAL(colorChanged(QColor)), this, SLOT(
      changeColor(QColor)));
15     connect(ui->keyWidget, SIGNAL(selectionChanged(QStringList)), this, SLOT(
      newSelection(QStringList)));
16     connect(ui->keyWidget, SIGNAL(sidelightToggled()), this, SLOT(
      toggleSidelight())); // click on a toggle button, like sidelight
17     connect(ui->animWidget, SIGNAL(animChanged(KbAnim*)), this, SLOT(
      changeAnim(KbAnim*)));
18     connect(ui->animWidget, SIGNAL(didUpdateSelection(QStringList)), this, SLOT(
      changeAnimKeys(QStringList)));
19
20     // Restore "show animated" setting
21     ui->showAnimBox->setChecked(!CkbSettings::get("UI/Light/ShowBaseOnly").
      toBool());
22 }
```

Here is the call graph for this function:



12.31.2.2 KbLightWidget::~KbLightWidget ( )

Definition at line 24 of file kblightwidget.cpp.

References ui.

```
24                                {
25     delete ui;
26 }
```

12.31.3 Member Function Documentation

12.31.3.1 void KbLightWidget::changeAnim (KbAnim * *newAnim*) [private], [slot]

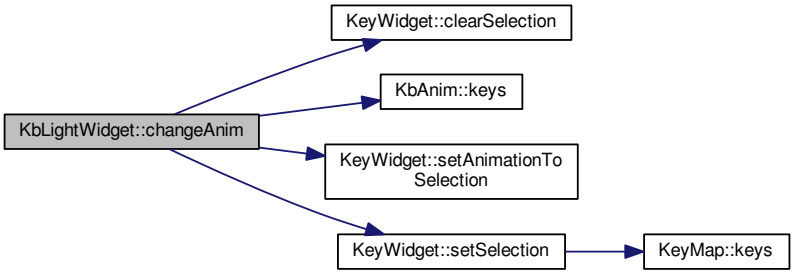
Definition at line 121 of file kblightwidget.cpp.

References KeyWidget::clearSelection(), KbAnim::keys(), Ui_KbLightWidget::keyWidget, KeyWidget::setAnimationToSelection(), KeyWidget::setSelection(), and ui.

Referenced by KbLightWidget().

```
121                                              {
122     if(newAnim)
123         ui->keyWidget->setSelection(newAnim->keys());
124     else
125         ui->keyWidget->clearSelection();
126     ui->keyWidget->setAnimationToSelection();
127 }
```

Here is the call graph for this function:



Here is the caller graph for this function:

12.31.3.2 void KbLightWidget::changeAnimKeys (QStringList *keys*) [private], [slot]

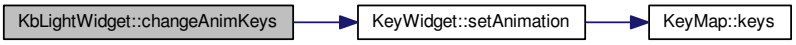
Definition at line 129 of file kblightwidget.cpp.

References Ui_KbLightWidget::keyWidget, KeyWidget::setAnimation(), and ui.

Referenced by KbLightWidget().

```
129                                                 {
130     ui->keyWidget->setAnimation(keys);
131 }
```

Here is the call graph for this function:



Here is the caller graph for this function:

12.31.3.3 void KbLightWidget::changeColor (QColor *newColor*) [private], [slot]

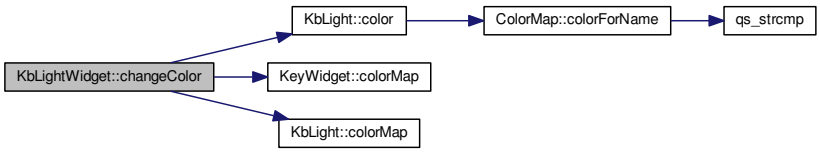
Definition at line 99 of file kblightwidget.cpp.

References KbLight::color(), KeyWidget::colorMap(), KbLight::colorMap(), currentSelection, Ui_KbLightWidget::keyWidget, light, and ui.

Referenced by KbLightWidget().

```
99                                                  {
100     if(light){
101         foreach(QString key, currentSelection)
102             light->color(key, newColor);
103         ui->keyWidget->colorMap(light->colorMap());
104     }
105 }
```

Here is the call graph for this function:



Here is the caller graph for this function:

12.31.3.4 void KbLightWidget::newSelection (QStringList *selection*) [private], [slot]

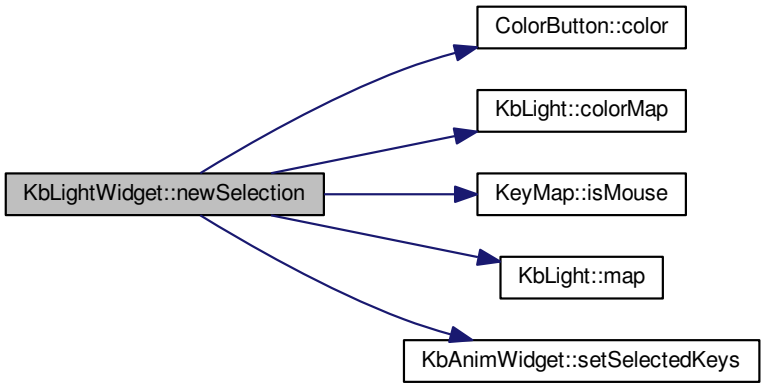
Definition at line 72 of file kblightwidget.cpp.

References Ui_KbLightWidget::animWidget, Ui_KbLightWidget::bgButton, ColorButton::color(), KbLight::colorMap(), currentSelection, KeyMap::isMouse(), light, KbLight::map(), Ui_KbLightWidget::selLabel, KbAnimWidget::setSelectedKeys(), and ui.

Referenced by KbLightWidget().

```
72                                                  {
73     // Determine selected color (invalid color if no selection or if they're not all the same)
74     QColor selectedColor;
75     const QColorMap& colorMap = light->colorMap();
76     foreach(const QString& key, selection){
77         QColor color = colorMap.value(key);
78         if(!selectedColor.isValid())
79             selectedColor = color;
80         else if(color != selectedColor){
81             selectedColor = QColor();
82             break;
83         }
84     }
85
86     currentSelection = selection;
87     ui->animWidget->setSelectedKeys(selection);
88     ui->bgButton->color(selectedColor);
89     int count = selection.count();
90     if(count == 0){
91         ui->selLabel->setText("Click to select");
92         return;
93     } else if(count == 1)
94         ui->selLabel->setText(tr("1 %1 selected").arg(light->map().
      isMouse() ? "zone" : "key"));
95     else
96         ui->selLabel->setText(tr("%1 %2 selected").arg(count).arg(
      light->map().isMouse() ? "zones" : "keys"));
97 }
```

Here is the call graph for this function:



Here is the caller graph for this function:

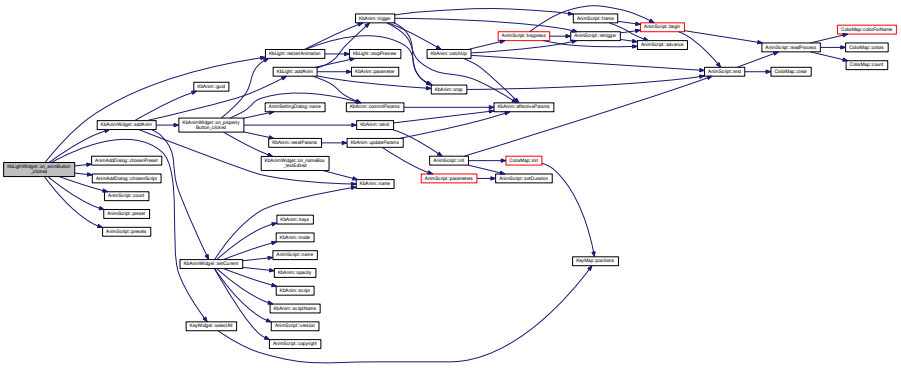
12.31.3.5 void KbLightWidget::on_animButton_clicked ( ) [private],[slot]

Definition at line 143 of file kblightwidget.cpp.

References KbAnimWidget::addAnim(), Ui_KbLightWidget::animWidget, AnimAddDialog::chosenPreset(), AnimAddDialog::chosenScript(), AnimScript::count(), currentSelection, Ui_KbLightWidget::keyWidget, light, AnimScript::preset(), AnimScript::presets(), KbLight::restartAnimation(), KeyWidget::selectAll(), and ui.

```
143                                         {
144     if(AnimScript::count() == 0)
145         return;
146     if(currentSelection.isEmpty())
147         ui->keyWidget->selectAll();
148     AnimAddDialog dialog(this, light, currentSelection);
149     dialog.exec();
150     if(dialog.result() != QDialog::Accepted)
151         return;
152     const AnimScript* script = dialog.chosenScript();
153     int presetId = dialog.chosenPreset();
154     ui->animWidget->addAnim(script, currentSelection, script->
      presets()[presetId], script->preset(presetId));
155     light->restartAnimation();
156 }
```

Here is the call graph for this function:



12.31.3.6 void KbLightWidget::on_bgButton_clicked ( ) [private], [slot]

Definition at line 138 of file kblightwidget.cpp.

References currentSelection, Ui_KbLightWidget::keyWidget, KeyWidget::selectAll(), and ui.

```
138                                         {
139     if(currentSelection.isEmpty())
140         ui->keyWidget->selectAll();
141 }
```

Here is the call graph for this function:

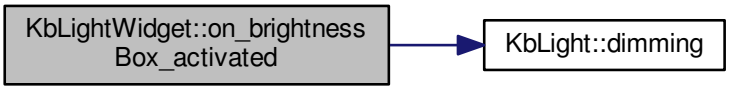
12.31.3.7 void KbLightWidget::on_brightnessBox_activated (int *index*) [private], [slot]

Definition at line 133 of file kblightwidget.cpp.

References KbLight::dimming(), and light.

```
133                                                     {
134     if(light)
135         light->dimming(index);
136 }
```

Here is the call graph for this function:



12.31.3.8 void KbLightWidget::on_showAnimBox_clicked (bool *checked*) [private], [slot]

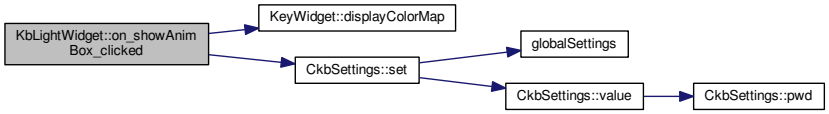
Definition at line 52 of file kblightwidget.cpp.

References KeyWidget::displayColorMap(), Ui_KbLightWidget::keyWidget, light, CkbSettings::set(), and ui.

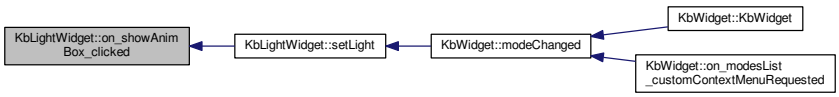
Referenced by setLight().

```
52                                                  {
53     // FIXME: set globally if there's more than one KbLightWidget active
54     // Connect/disconnect animation slot
55     if(checked){
56         if(light)
57             connect(light, SIGNAL(frameDisplayed(const ColorMap&,const QSet<QString>&)),
      ui->keyWidget, SLOT(displayColorMap(const ColorMap&,const QSet<QString>&)));
58     } else {
59         if(light)
60             disconnect(light, SIGNAL(frameDisplayed(const ColorMap&,const QSet<QString>&)),
      ui->keyWidget, SLOT(displayColorMap(const ColorMap&,const QSet<QString>&)));
61         ui->keyWidget->displayColorMap(ColorMap());
62     }
63     CkbSettings::set("UI/Light/ShowBaseOnly", !checked);
64 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.31.3.9 void KbLightWidget::setLight (KbLight * *newLight*)

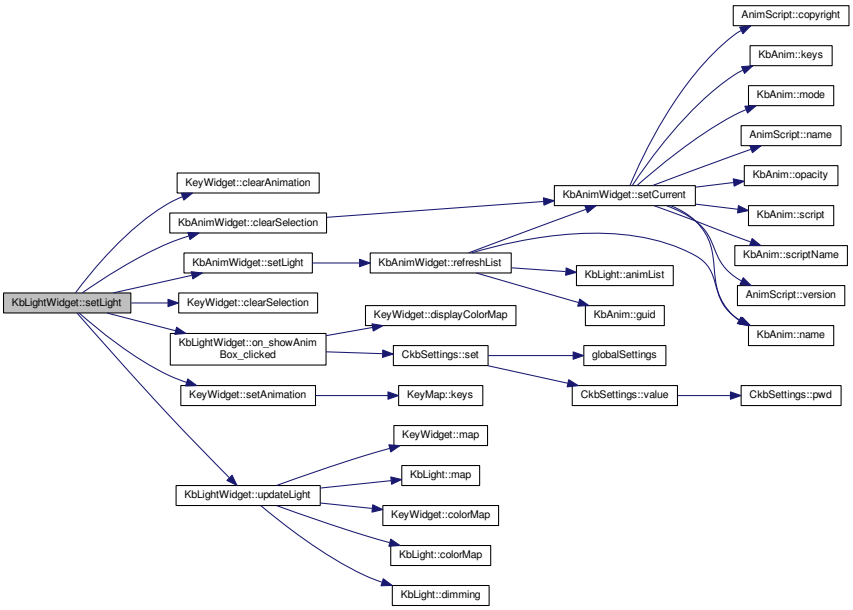
Definition at line 28 of file kblightwidget.cpp.

References Ui_KbLightWidget::animWidget, KeyWidget::clearAnimation(), KbAnimWidget::clearSelection(), KeyWidget::clearSelection(), Ui_KbLightWidget::keyWidget, light, on_showAnimBox_clicked(), KeyWidget::setAnimation(), KbAnimWidget::setLight(), Ui_KbLightWidget::showAnimBox, ui, and updateLight().

Referenced by KbWidget::modeChanged().

```
28                                          {
29     ui->keyWidget->clearSelection();
30     ui->keyWidget->clearAnimation();
31     ui->animWidget->clearSelection();
32     if(light == newLight)
33         return;
34     if(light){
35         disconnect(light, SIGNAL(updated()), this, SLOT(updateLight()));
36         // Disabled animated colors for previous light
37         on_showAnimBox_clicked(false);
38     }
39     connect(newLight, SIGNAL(updated()), this, SLOT(updateLight()));
40     light = newLight;
41     updateLight();
42     ui->keyWidget->setAnimation(QStringList());
43     ui->animWidget->setLight(newLight);
44     if(ui->showAnimBox->isChecked())
45         on_showAnimBox_clicked(true);
46 }
```

Here is the call graph for this function:



Here is the caller graph for this function:

12.31.3.10 void KbLightWidget::setMonochrome ( )

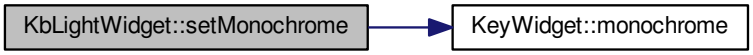
Definition at line 48 of file kblightwidget.cpp.

References Ui_KbLightWidget::keyWidget, KeyWidget::monochrome(), and ui.

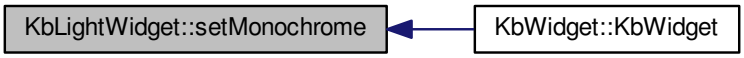
Referenced by KbWidget::KbWidget().

```
48                                  {
49     ui->keyWidget->monochrome(true);
50 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.31.3.11 void KbLightWidget::toggleSidelight ( ) [private], [slot]

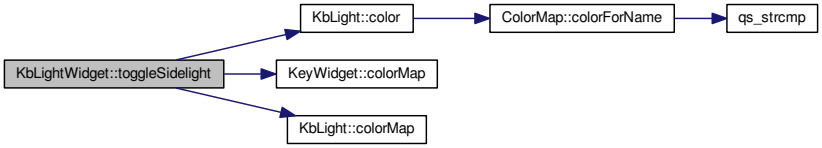
Definition at line 107 of file kblightwidget.cpp.

References KbLight::color(), KeyWidget::colorMap(), KbLight::colorMap(), Ui_KbLightWidget::keyWidget, light, and ui.

Referenced by KbLightWidget().

```
107                                      {
108     if(light){
109         if (light->colorMap()["lsidel"] == 0xFF000000){
110                 light->color("lsidel",QRgb(0xFFFFFFFF));
111                 light->color("rsidel",QRgb(0xFFFFFFFF));
112         } else {
113                 light->color("lsidel", QRgb(0xFF000000));
114                 light->color("rsidel", QRgb(0xFF000000));
115         }
116         ui->keyWidget->colorMap(light->colorMap());
117     }
118 }
```

Here is the call graph for this function:



Here is the caller graph for this function:

12.31.3.12 void KbLightWidget::updateLight ( ) [private],[slot]

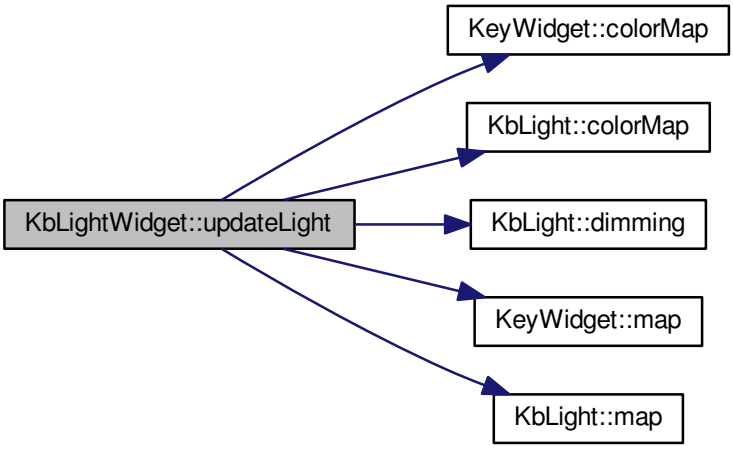
Definition at line 66 of file kblightwidget.cpp.

References Ui_KbLightWidget::brightnessBox, KeyWidget::colorMap(), KbLight::colorMap(), KbLight::dimming(), Ui_KbLightWidget::keyWidget, light, KeyWidget::map(), KbLight::map(), and ui.

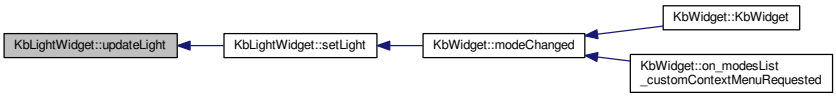
Referenced by setLight().

```
66                                {
67     ui->keyWidget->map(light->map());
68     ui->keyWidget->colorMap(light->colorMap());
69     ui->brightnessBox->setCurrentIndex(light->dimming());
70 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.31.4 Field Documentation

12.31.4.1 QStringList KbLightWidget::currentSelection [private]

Definition at line 44 of file kblightwidget.h.

Referenced by changeColor(), newSelection(), on_animButton_clicked(), and on_bgButton_clicked().

12.31.4.2 KbLight* KbLightWidget::light [private]

Definition at line 43 of file kblightwidget.h.

Referenced by changeColor(), newSelection(), on_animButton_clicked(), on_brightnessBox_activated(), on_showAnimBox_clicked(), setLight(), toggleSidelight(), and updateLight().

12.31.4.3 Ui::KbLightWidget* KbLightWidget::ui [private]

Definition at line 46 of file kblightwidget.h.

Referenced by changeAnim(), changeAnimKeys(), changeColor(), KbLightWidget(), newSelection(), on_animButton_clicked(), on_bgButton_clicked(), on_showAnimBox_clicked(), setLight(), setMonochrome(), toggleSidelight(), updateLight(), and ~KbLightWidget().

The documentation for this class was generated from the following files:

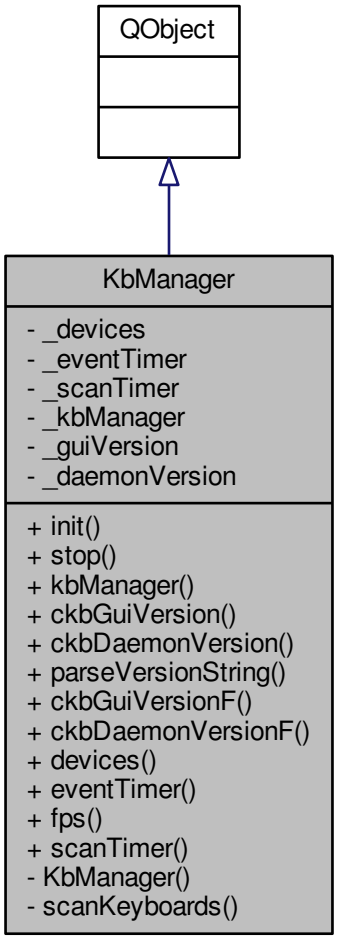
- src/ckb/kblightwidget.h

- src/ckb/kblightwidget.cpp

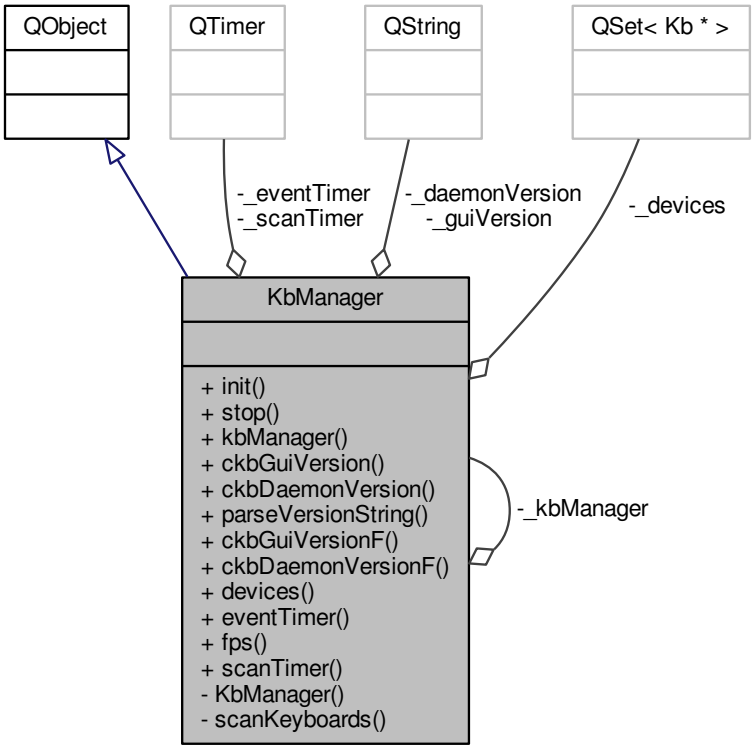
12.32 KbManager Class Reference

`#include <src/ckb/kbmanager.h>`

Inheritance diagram for KbManager:



Collaboration diagram for KbManager:



Signals

- void kbConnected (Kb ∗device)
- void kbDisconnected (Kb ∗device)
- void versionUpdated ()

Static Public Member Functions

- static void init (QString guiVersion)
- static void stop ()
- static KbManager ∗ kbManager ()
- static QString ckbGuiVersion ()
- static QString ckbDaemonVersion ()
- static float parseVersionString (QString version)
- static float ckbGuiVersionF ()
- static float ckbDaemonVersionF ()
- static const QSet< Kb ∗ > devices ()
- static QTimer ∗ eventTimer ()
- static void fps (int framerate)
- static QTimer ∗ scanTimer ()

Private Slots

- void scanKeyboards ()

Private Member Functions

- KbManager (QObject *parent=0)

Private Attributes

- QSet< Kb * > _devices
- QTimer * _eventTimer
- QTimer * _scanTimer

Static Private Attributes

- static KbManager * _kbManager = 0
- static QString _guiVersion
- static QString _daemonVersion = "<unavailable>"

12.32.1 Detailed Description

Definition at line 14 of file kbmanager.h.

12.32.2 Constructor & Destructor Documentation

12.32.2.1 KbManager::KbManager (QObject * *parent* = 0) [explicit], [private]

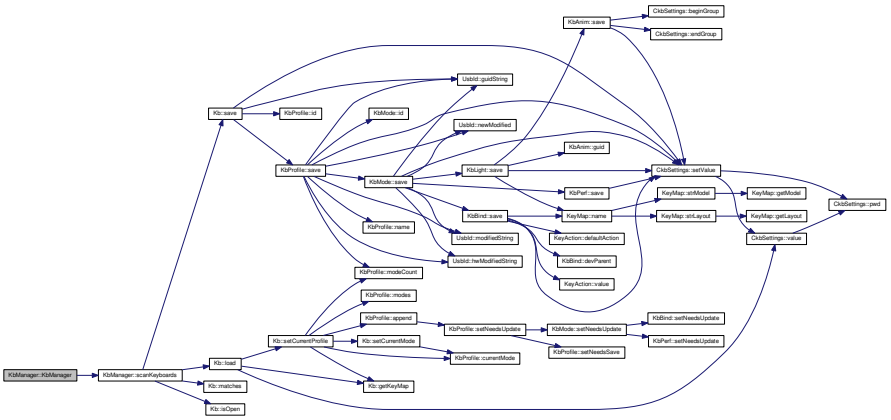
Definition at line 26 of file kbmanager.cpp.

References _eventTimer, _scanTimer, and scanKeyboards().

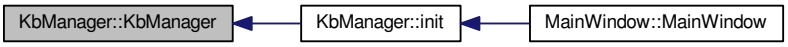
Referenced by init().

```
26                                  : QObject(parent){
27     // Set up the timers
28     _eventTimer = new QTimer(this);
29     _eventTimer->setTimerType(Qt::PreciseTimer);
30     _scanTimer = new QTimer(this);
31     _scanTimer->start(1000);
32     connect(_scanTimer, SIGNAL(timeout()), this, SLOT(scanKeyboards()));
33
34     // Scan for keyboards immediately so they show up as soon as the GUI appears.
35     QTimer::singleShot(0,this,SLOT(scanKeyboards()));
36 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.32.3 Member Function Documentation

12.32.3.1 static QString KbManager::ckbDaemonVersion ( ) [inline], [static]

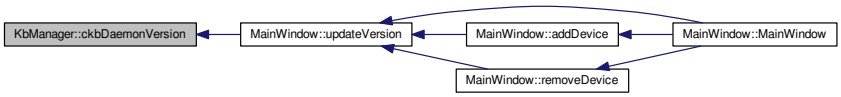
Definition at line 27 of file kbmanager.h.

References _daemonVersion.

Referenced by MainWindow::updateVersion().

```
27 { return _daemonVersion; }
```

Here is the caller graph for this function:



12.32.3.2 static float KbManager::ckbDaemonVersionF ( ) [inline], [static]

Definition at line 31 of file kbmanager.h.

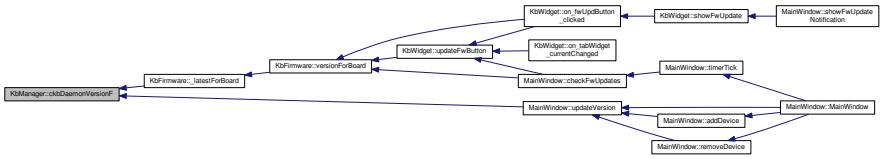
References _daemonVersion, DAEMON_UNAVAILABLE_STR, and parseVersionString().

Referenced by KbFirmware::_latestForBoard(), and MainWindow::updateVersion().

```
31 { return _daemonVersion == DAEMON_UNAVAILABLE_STR ? INFINITY :
      parseVersionString(_daemonVersion); }
```

Here is the call graph for this function:

Here is the caller graph for this function:



12.32.3.3 static QString KbManager::ckbGuiVersion ( ) [inline], [static]

Definition at line 26 of file kbmanager.h.

References _guiVersion.

```
26 { return _guiVersion; }
```

12.32.3.4 static float KbManager::ckbGuiVersionF ( ) [inline], [static]

Definition at line 30 of file kbmanager.h.

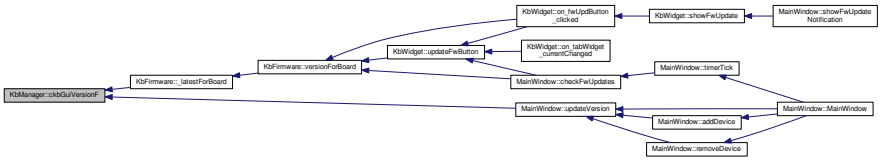
References _guiVersion, and parseVersionString().

Referenced by KbFirmware::_latestForBoard(), and MainWindow::updateVersion().

```
30 { return parseVersionString(_guiVersion); }
```

Here is the call graph for this function:

Here is the caller graph for this function:



12.32.3.5 static const QSet<Kb*> KbManager::devices ( ) [inline], [static]

Definition at line 34 of file kbmanager.h.

References _devices, and _kbManager.

```
34 { return _kbManager ? _kbManager->_devices : QSet<Kb*>(); }
```

12.32.3.6 static QTimer* KbManager::eventTimer ( ) [inline], [static]

Definition at line 38 of file kbmanager.h.

References _eventTimer, and _kbManager.

Referenced by fps().

```
38 { return _kbManager ? _kbManager->_eventTimer : 0; }
```

Here is the caller graph for this function:

12.32.3.7 void KbManager::fps (int *framerate*) [static]

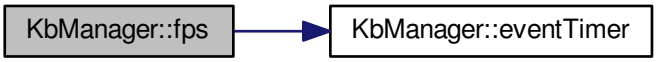
Definition at line 38 of file kbmanager.cpp.

References eventTimer().

Referenced by Kb::frameRate().

```
38                                {
39     QTimer* timer = eventTimer();
40     if(!timer)
41         return;
42     if(timer->isActive())
43         timer->setInterval(1000 / framerate);
44     else
45         timer->start(1000 / framerate);
46 }
```

Here is the call graph for this function:



Here is the caller graph for this function:

12.32.3.8 void KbManager::init (QString *guiVersion*) [static]

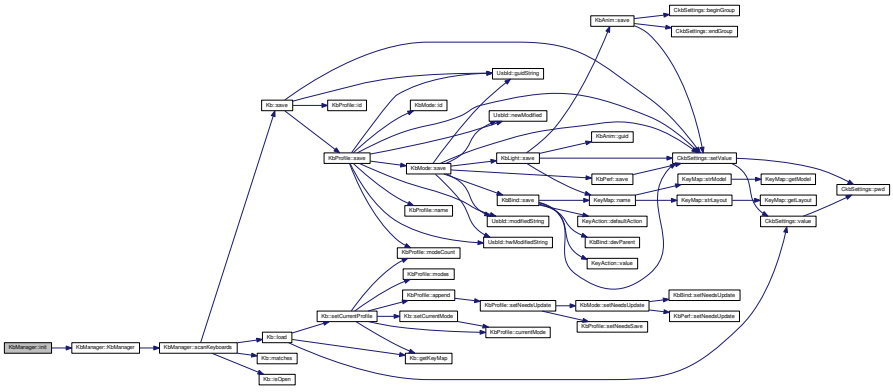
Definition at line 12 of file kbmanager.cpp.

References _guiVersion, _kbManager, and KbManager().

Referenced by MainWindow::MainWindow().

```
12                                          {
13     _guiVersion = guiVersion;
14     if(_kbManager)
15         return;
16     _kbManager = new KbManager();
17 }
```

Here is the call graph for this function:



Here is the caller graph for this function:

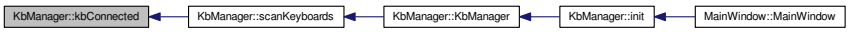
12.32.3.9 void KbManager::kbConnected (Kb * *device*) [signal]

Definition at line 174 of file moc_kbmanager.cpp.

Referenced by scanKeyboards().

```
175 {
176     void *_a[] = { Q_NULLPTR, const_cast<void*>(reinterpret_cast<const void*>(&_t1)) };
177     QMetaObject::activate(this, &staticMetaObject, 0, _a);
178 }
```

Here is the caller graph for this function:



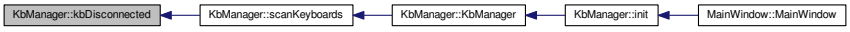
12.32.3.10 void KbManager::kbDisconnected (Kb * *device*) [signal]

Definition at line 181 of file moc_kbmanager.cpp.

Referenced by scanKeyboards().

```
182 {
183     void *_a[] = { Q_NULLPTR, const_cast<void*>(reinterpret_cast<const void*>(&_t1)) };
184     QMetaObject::activate(this, &staticMetaObject, 1, _a);
185 }
```

Here is the caller graph for this function:



12.32.3.11 static KbManager* KbManager::kbManager ( ) [inline], [static]

Definition at line 23 of file kbmanager.h.

References _kbManager.

Referenced by MainWindow::MainWindow().

```
23 { return _kbManager; }
```

Here is the caller graph for this function:

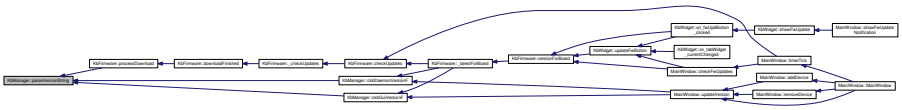
12.32.3.12 float KbManager::parseVersionString (QString *version*) [static]

Definition at line 48 of file kbmanager.cpp.

Referenced by ckbDaemonVersionF(), ckbGuiVersionF(), and KbFirmware::processDownload().

```
48                                                  {
49     // Remove extraneous info (anything after a +, anything non-numeric)
50     QStringList dots = version.replace(QRegExp("\\+.+"), "").replace(QRegExp("[^\\d\\.]"), "").split(".");
51     float base = 1.f;
52     float res = 0.f;
53     // A number like "1.2.3" will result in 1.0203
54     // NB: will fail if a point version goes over 99 or if using more than two dots. floats can only
       reliably encode 7 decimal digits.
55     foreach(const QString& dot, dots){
56         res += dot.toFloat() * base;
57         base /= 100.f;
58     }
59     return res;
60 }
```

Here is the caller graph for this function:



12.32.3.13 void KbManager::scanKeyboards ( ) [private], [slot]

Definition at line 62 of file kbmanager.cpp.

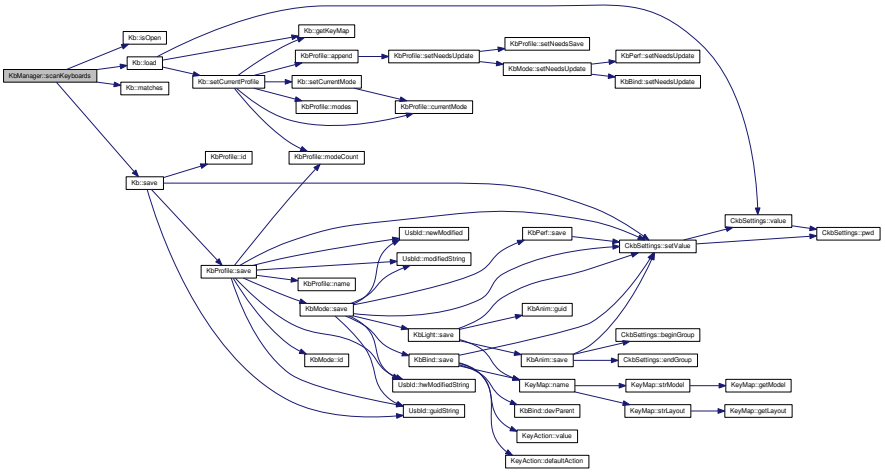
References _daemonVersion, _devices, _eventTimer, _scanTimer, DAEMON_UNAVAILABLE_STR, devpath, Kb::isOpen(), kbConnected(), kbDisconnected(), Kb::load(), Kb::matches(), Kb::save(), and versionUpdated().

Referenced by KbManager().

```
62                                {
63     QString rootdev = devpath.arg(0);
64     QFile connected(rootdev + "/connected");
65     if(!connected.open(QIODevice::ReadOnly)){
66         // No root controller - remove all keyboards
67         foreach(Kb* kb, _devices){
68             emit kbDisconnected(kb);
69             kb->save();
70             delete kb;
71         }
72         _devices.clear();
73         if(_daemonVersion != DAEMON_UNAVAILABLE_STR){
```

```
74              _daemonVersion = DAEMON_UNAVAILABLE_STR;
75              emit versionUpdated();
76          }
77          return;
78      }
79      // Check daemon version
80      QFile version(rootdev + "/version");
81      QString vString;
82      if(version.open(QIODevice::ReadOnly)){
83          vString = QString::fromUtf8(version.readLine()).trimmed();
84          version.close();
85      } else
86          vString = DAEMON_UNAVAILABLE_STR;
87      if(_daemonVersion != vString){
88          _daemonVersion = vString;
89          emit versionUpdated();
90      }
91
92      // Scan connected devices
93      QList<QStringList> lines;
94      while(1){
95          QString line = connected.readLine().trimmed();
96          if(line.isEmpty())
97              break;
98          QStringList components = line.split(" ");
99          if(components.length() < 2)             // "<path> <serial> <name>" (we're only interested in the
    first two)
100             continue;
101         lines.append(components);
102     }
103     connected.close();
104
105     // Remove any active devices not in the list
106     QMutableSetIterator<Kb*> i(_devices);
107     while(i.hasNext()){
108         Kb* kb = i.next();
109         bool matched = false;
110         foreach(const QStringList& line, lines){
111             if(kb->matches(line[0], line[1])){
112                 matched = true;
113                 break;
114             }
115         }
116         if(matched)
117             continue;
118         // Device not found, remove
119         i.remove();
120         emit kbDisconnected(kb);
121         kb->save();
122         delete kb;
123     }
124
125     // Add any new devices found in the list
126     foreach(const QStringList& line, lines){
127         bool matched = false;
128         foreach(Kb* kb, _devices){
129             if(kb->matches(line[0], line[1])){
130                 matched = true;
131                 break;
132             }
133         }
134         if(matched)
135             continue;
136         // Device not found, create new
137         Kb* kb = new Kb(this, line[0]);
138         if(!kb->isOpen()){
139             delete kb;
140             continue;
141         }
142         _devices.insert(kb);
143         // Load preferences and send signal
144         emit kbConnected(kb);
145         kb->load();
146         connect(_eventTimer, SIGNAL(timeout()), kb, SLOT(frameUpdate()));
147         connect(_scanTimer, SIGNAL(timeout()), kb, SLOT(autoSave()));
148     }
149 }
```

Here is the call graph for this function:



Here is the caller graph for this function:

12.32.3.14 static QTimer* KbManager::scanTimer ( ) [inline], [static]

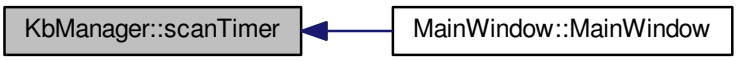
Definition at line 43 of file kbmanager.h.

References _kbManager, and _scanTimer.

Referenced by MainWindow::MainWindow().

```
43 { return _kbManager ? _kbManager->_scanTimer : 0; }
```

Here is the caller graph for this function:



12.32.3.15 void KbManager::stop ( ) [static]

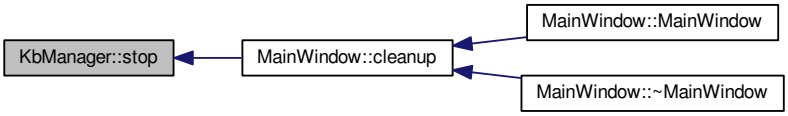
Definition at line 19 of file kbmanager.cpp.

References _kbManager.

Referenced by MainWindow::cleanup().

```
19                      {
20     if(!_kbManager)
21         return;
22     delete _kbManager;
23     _kbManager = 0;
24 }
```

Here is the caller graph for this function:



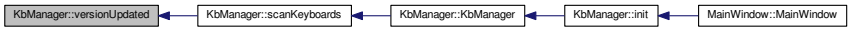
12.32.3.16 void KbManager::versionUpdated ( ) [signal]

Definition at line 188 of file moc_kbmanager.cpp.

Referenced by scanKeyboards().

```
189 {
190     QMetaObject::activate(this, &staticMetaObject, 2, Q_NULLPTR);
191 }
```

Here is the caller graph for this function:



12.32.4 Field Documentation

12.32.4.1 QString KbManager::_daemonVersion = "<unavailable>" [static], [private]

Definition at line 59 of file kbmanager.h.

Referenced by ckbDaemonVersion(), ckbDaemonVersionF(), and scanKeyboards().

12.32.4.2 QSet<Kb*> KbManager::_devices [private]

Definition at line 63 of file kbmanager.h.

Referenced by devices(), and scanKeyboards().

12.32.4.3 QTimer* KbManager::_eventTimer [private]

Definition at line 64 of file kbmanager.h.

Referenced by eventTimer(), KbManager(), and scanKeyboards().

12.32.4.4 QString KbManager::_guiVersion [static],[private]

Definition at line 59 of file kbmanager.h.

Referenced by ckbGuiVersion(), ckbGuiVersionF(), and init().

12.32.4.5 KbManager * KbManager::_kbManager = 0 [static],[private]

Definition at line 58 of file kbmanager.h.

Referenced by devices(), eventTimer(), init(), kbManager(), scanTimer(), and stop().

12.32.4.6 QTimer * KbManager::_scanTimer [private]

Definition at line 64 of file kbmanager.h.

Referenced by KbManager(), scanKeyboards(), and scanTimer().

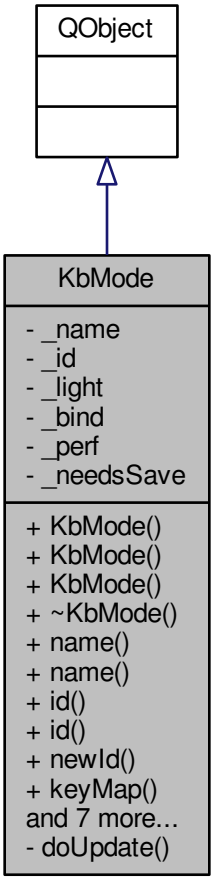
The documentation for this class was generated from the following files:

- src/ckb/kbmanager.h
- src/ckb/kbmanager.cpp
- src/ckb/release/moc_kbmanager.cpp

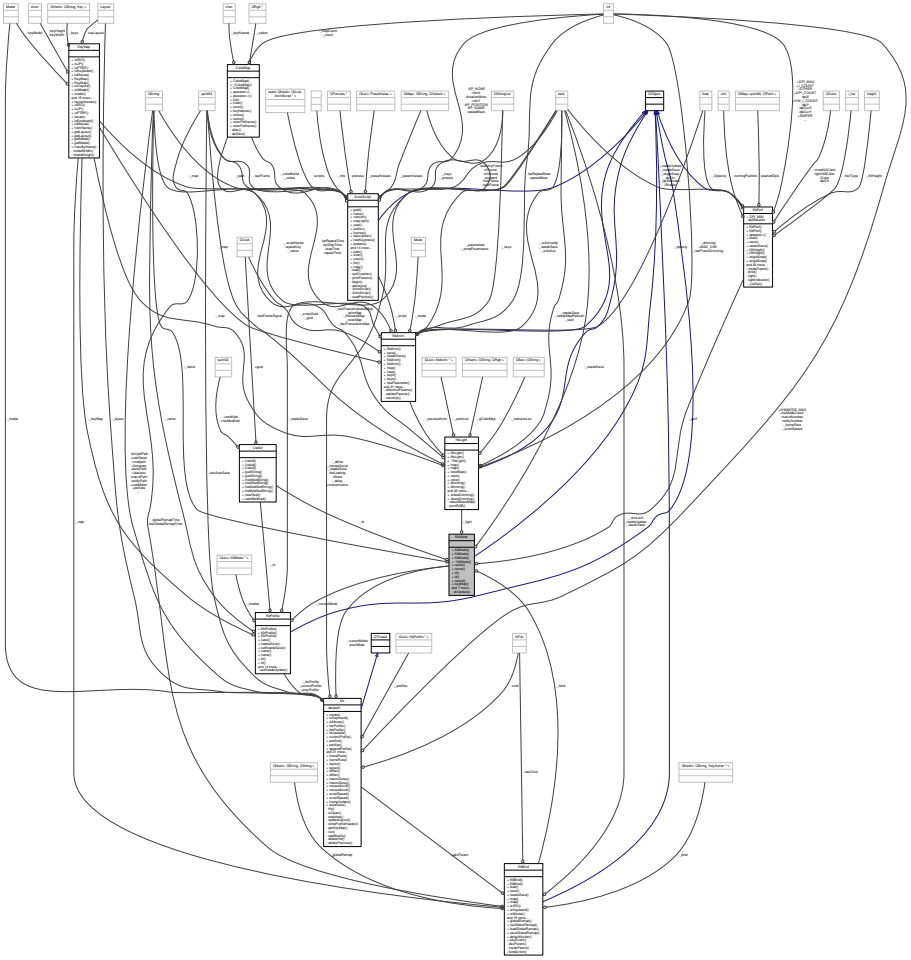
12.33 KbMode Class Reference

`#include <src/ckb/kbmode.h>`

Inheritance diagram for KbMode:



Collaboration diagram for KbMode:



Signals

- void updated ()

Public Member Functions

- KbMode (Kb *parent, const KeyMap &keyMap, const QString &guid="", const QString &modified="")
- KbMode (Kb *parent, const KeyMap &keyMap, CkbSettings &settings)
- KbMode (Kb *parent, const KeyMap &keyMap, const KbMode &other)

  KbMode::KbMode.
- ~KbMode ()

  KbMode::~KbMode Destructor may be used for Debugging (issue #38 with SIGSEGV). Insert qDebug-statement.
- const QString & name () const
- void name (const QString &newName)
- UsbId & id ()
- void id (const UsbId &newId)
- void newId ()
- void keyMap (const KeyMap &keyMap)
- KbLight * light ()
- KbBind * bind ()
- KbPerf * perf ()

- void save (CkbSettings &settings)
- bool needsSave () const
- void setNeedsSave ()
- void setNeedsUpdate ()

Private Slots

- void doUpdate ()

Private Attributes

- QString _name
- UsbId _id
- KbLight * _light
- KbBind * _bind
- KbPerf * _perf
- bool _needsSave

12.33.1 Detailed Description

Definition at line 36 of file kbmode.h.

12.33.2 Constructor & Destructor Documentation

12.33.2.1 KbMode::KbMode (Kb * *parent*, const KeyMap & *keyMap*, const QString & *guid* = "", const QString & *modified* = "")

Definition at line 4 of file kbmode.cpp.

References _id, _light, doUpdate(), UsbId::guid, and updated().

```
4                                                                                                    :
5     QObject(parent),
6     _name("Unnamed"), _id(guid, modified),
7     _light(new KbLight(this, keyMap)), _bind(new KbBind(this, parent, keyMap)),
      _perf(new KbPerf(this)),
8     _needsSave(true)
9 {
10     connect(_light, SIGNAL(updated()), this, SLOT(doUpdate()));
11     if(_id.guid.isNull())
12         _id.guid = QUuid::createUuid();
13 }
```

Here is the call graph for this function:

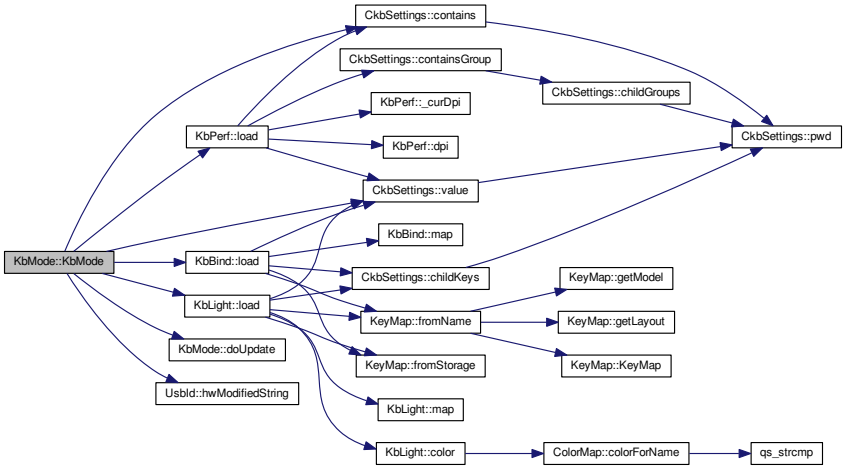
12.33.2.2 KbMode::KbMode (Kb * *parent*, const KeyMap & *keyMap*, CkbSettings & *settings*)

Definition at line 30 of file kbmode.cpp.

References _bind, _id, _light, _name, _perf, CkbSettings::contains(), doUpdate(), UsbId::guid, UsbId::hwModified, UsbId::hwModifiedString(), KbPerf::load(), KbBind::load(), KbLight::load(), UsbId::modified, updated(), and CkbSettings::value().

```
30                                                                          :
31     QObject(parent),
32     _name(settings.value("Name").toString().trimmed()),
33     _id(settings.value("GUID").toString().trimmed(), settings.value("Modified").toString().
       trimmed()),
34     _light(new KbLight(this, keyMap)), _bind(new KbBind(this, parent, keyMap)),
       _perf(new KbPerf(this)),
35     _needsSave(false)
36 {
37     if(settings.contains("HwModified"))
38         _id.hwModifiedString(settings.value("HwModified").toString());
39     else
40         _id.hwModified = _id.modified;
41
42     connect(_light, SIGNAL(updated()), this, SLOT(doUpdate()));
43     if(_id.guid.isNull())
44         _id.guid = QUuid::createUuid();
45     if(_name == "")
46         _name = "Unnamed";
47     _light->load(settings);
48     _bind->load(settings);
49     _perf->load(settings);
50 }
```

Here is the call graph for this function:



12.33.2.3 KbMode::KbMode (Kb * *parent*, const KeyMap & *keyMap*, const KbMode & *other*)

Parameters

parent	Kb as parent (append to the Keyboard list
keyMap	Map to copy from

other	Mode to copy from Constructor to copy an existing Keyboard-Mode KbMode &other

Definition at line 21 of file kbmode.cpp.

References _light, doUpdate(), and updated().

```
21                                                                    :
22     QObject(parent),
23     _name(other._name), _id(other._id),
24     _light(new KbLight(this, keyMap, *other._light)), _bind(new
       KbBind(this, parent, keyMap, *other._bind)), _perf(new KbPerf(this, *other.
       _perf)),
25     _needsSave(true)
26 {
27     connect(_light, SIGNAL(updated()), this, SLOT(doUpdate()));
28 }
```

Here is the call graph for this function:

12.33.2.4 KbMode::~KbMode ( )

Definition at line 89 of file kbmode.cpp.

```
89                 {
90 }
```

12.33.3 Member Function Documentation

12.33.3.1 KbBind* KbMode::bind ( ) [inline]

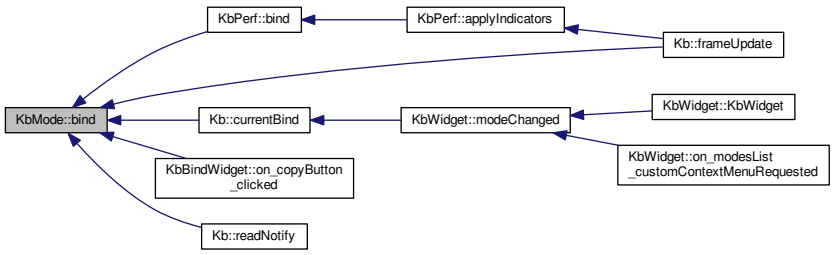
Definition at line 61 of file kbmode.h.

References _bind.

Referenced by KbPerf::bind(), Kb::currentBind(), Kb::frameUpdate(), KbBindWidget::on_copyButton_clicked(), and Kb::readNotify().

```
61 { return _bind; }
```

Here is the caller graph for this function:



12.33.3.2 void KbMode::doUpdate ( ) [private], [slot]

Definition at line 82 of file kbmode.cpp.

References updated().

Referenced by KbMode().

```
82                      {
83     emit updated();
84 }
```

Here is the caller graph for this function:

12.33.3.3 UsbId& KbMode::id ( ) [inline]

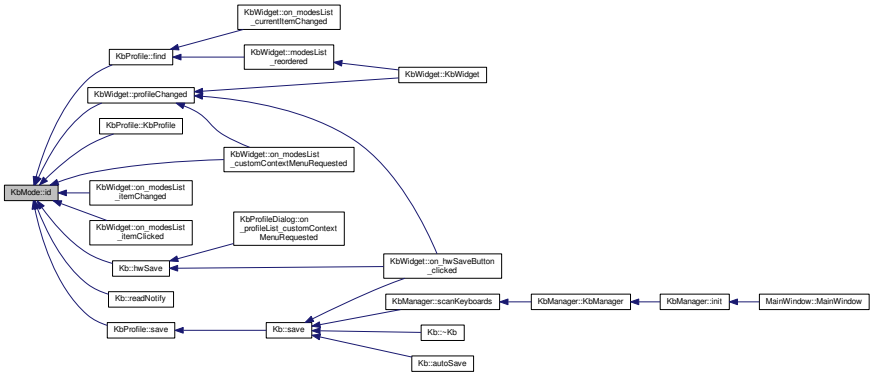
Definition at line 52 of file kbmode.h.

References _id.

Referenced by KbProfile::find(), Kb::hwSave(), KbProfile::KbProfile(), KbWidget::on_modesList_customContextMenuRequested(), KbWidget::on_modesList_itemChanged(), KbWidget::on_modesList_itemClicked(), KbWidget::profileChanged(), Kb::readNotify(), and KbProfile::save().

```
52 { return _id; }
```

Here is the caller graph for this function:



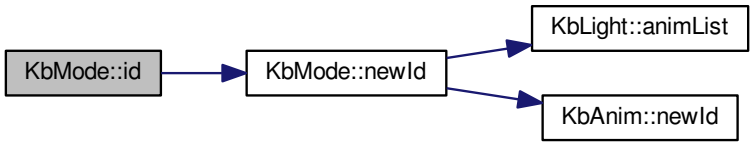
12.33.3.4 void KbMode::id (const UsbId & *newId*) [inline]

Definition at line 53 of file kbmode.h.

References _id, _needsSave, and newId().

```
53 { _needsSave = true; _id = newId; }
```

Here is the call graph for this function:



12.33.3.5 void KbMode::keyMap (const KeyMap & *keyMap*)

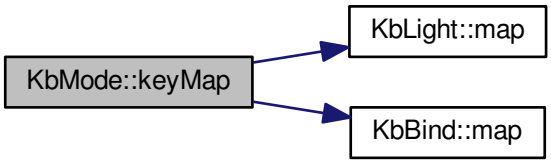
Definition at line 60 of file kbmode.cpp.

References _bind, _light, _needsSave, KbLight::map(), and KbBind::map().

Referenced by KbProfile::keyMap().

```
60                                      {
61     _needsSave = true;
62     _light->map(keyMap);
63     _bind->map(keyMap);
64 }
```

Here is the call graph for this function:



Here is the caller graph for this function:

12.33.3.6 KbLight* KbMode::light ( ) [inline]

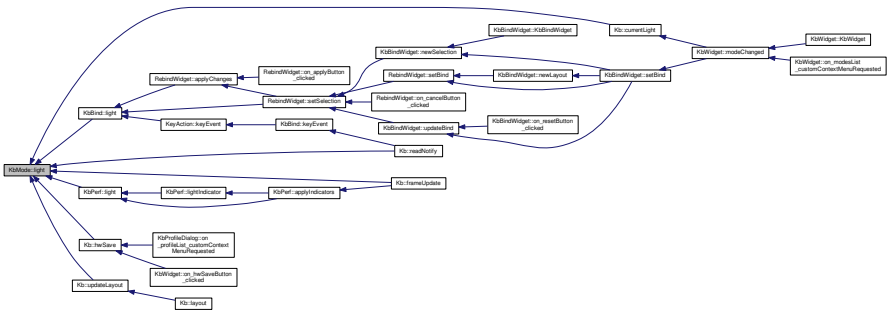
Definition at line 60 of file kbmode.h.

References _light.

Referenced by Kb::currentLight(), Kb::frameUpdate(), Kb::hwSave(), KbBind::light(), KbPerf::light(), Kb::readNotify(), and Kb::updateLayout().

```
60 { return _light; }
```

Here is the caller graph for this function:



12.33.3.7 const QString& KbMode::name ( ) const [inline]

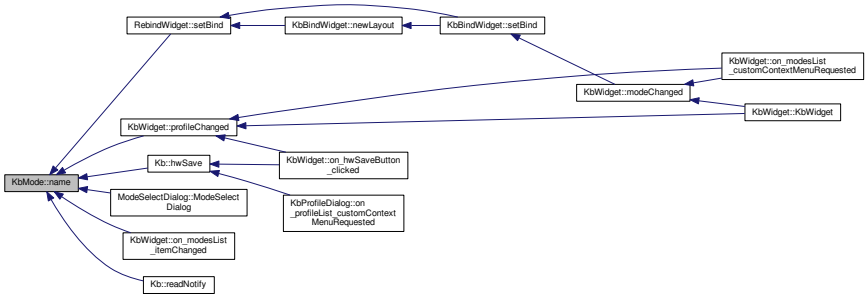
Definition at line 50 of file kbmode.h.

References _name.

Referenced by Kb::hwSave(), ModeSelectDialog::ModeSelectDialog(), KbWidget::on_modesList_itemChanged(), KbWidget::profileChanged(), Kb::readNotify(), and RebindWidget::setBind().

```
50 { return _name; }
```

Here is the caller graph for this function:



12.33.3.8 void KbMode::name (const QString & *newName*) [inline]

Definition at line 51 of file kbmode.h.

References _name, and _needsSave.

```
51 { _needsSave = true; _name = newName.trimmed(); if(_name == "")
       _name = "Unnamed"; }
```

12.33.3.9 bool KbMode::needsSave ( ) const

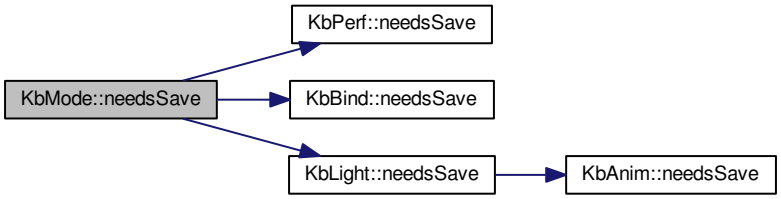
Definition at line 78 of file kbmode.cpp.

References _bind, _light, _needsSave, _perf, KbPerf::needsSave(), KbBind::needsSave(), and KbLight::needsSave().

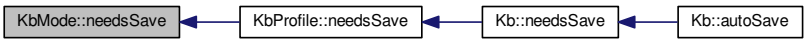
Referenced by KbProfile::needsSave().

```
78                              {
79     return _needsSave || _light->needsSave() || _bind->
       needsSave() || _perf->needsSave();
80 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.33.3.10 void KbMode::newId ( )

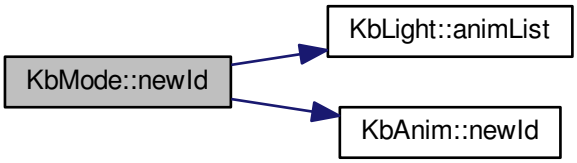
Definition at line 52 of file kbmode.cpp.

References _id, _light, _needsSave, KbLight::animList(), and KbAnim::newId().

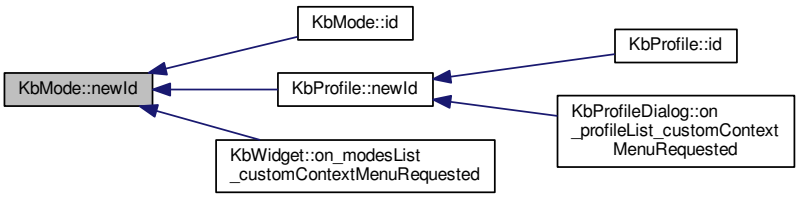
Referenced by id(), KbProfile::newId(), and KbWidget::on_modesList_customContextMenuRequested().

```
52                  {
53      _needsSave = true;
54      _id = UsbId();
55      // Create new IDs for animations
56      foreach(KbAnim* anim, _light->animList())
57          anim->newId();
58  }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.33.3.11 KbPerf* KbMode::perf ( ) [inline]

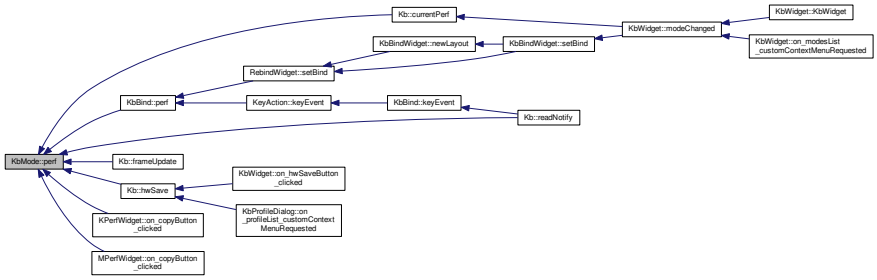
Definition at line 62 of file kbmode.h.

References _perf.

Referenced by Kb::currentPerf(), Kb::frameUpdate(), Kb::hwSave(), KPerfWidget::on_copyButton_clicked(), MPerfWidget::on_copyButton_clicked(), KbBind::perf(), and Kb::readNotify().

```
62 { return _perf; }
```

Here is the caller graph for this function:



12.33.3.12 void KbMode::save (CkbSettings & *settings*)

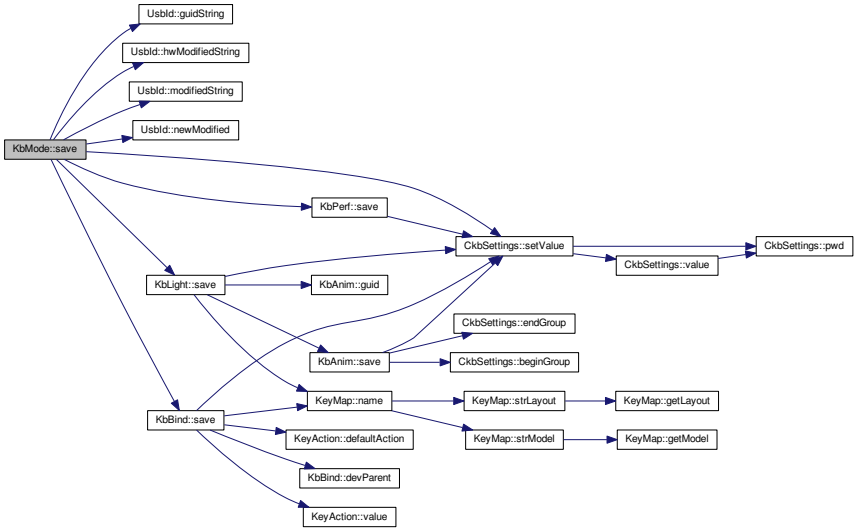
Definition at line 66 of file kbmode.cpp.

References _bind, _id, _light, _name, _needsSave, _perf, UsbId::guidString(), UsbId::hwModifiedString(), UsbId::modifiedString(), UsbId::newModified(), KbPerf::save(), KbBind::save(), KbLight::save(), and CkbSettings::setValue().

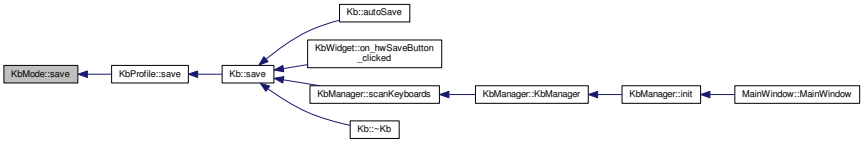
Referenced by KbProfile::save().

```
66                                  {
67     _needsSave = false;
68     _id.newModified();
69     settings.setValue("GUID", _id.guidString());
70     settings.setValue("Modified", _id.modifiedString());
71     settings.setValue("HwModified", _id.hwModifiedString());
72     settings.setValue("Name", _name);
73     _light->save(settings);
74     _bind->save(settings);
75     _perf->save(settings);
76 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.33.3.13 void KbMode::setNeedsSave ( ) [inline]

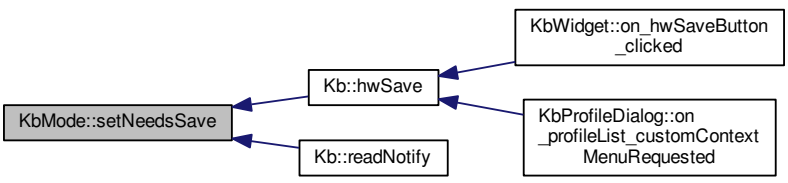
Definition at line 67 of file kbmode.h.

References _needsSave.

Referenced by Kb::hwSave(), and Kb::readNotify().

```
67 { _needsSave = true; }
```

Here is the caller graph for this function:



12.33.3.14 void KbMode::setNeedsUpdate ( ) [inline]

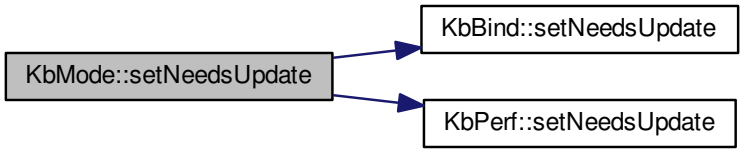
Definition at line 68 of file kbmode.h.

References _bind, _perf, KbBind::setNeedsUpdate(), and KbPerf::setNeedsUpdate().

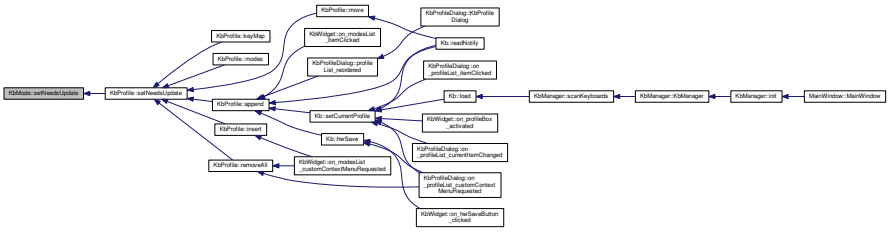
Referenced by KbProfile::setNeedsUpdate().

```
68 { _bind->setNeedsUpdate(); _perf->setNeedsUpdate(); }
```

Here is the call graph for this function:



Here is the caller graph for this function:



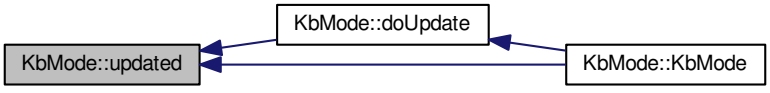
12.33.3.15 void KbMode::updated ( ) [signal]

Definition at line 131 of file moc_kbmode.cpp.

Referenced by doUpdate(), and KbMode().

```
132 {
133     QMetaObject::activate(this, &staticMetaObject, 0, Q_NULLPTR);
134 }
```

Here is the caller graph for this function:



12.33.4 Field Documentation

12.33.4.1 KbBind* KbMode::_bind [private]

Definition at line 78 of file kbmode.h.

Referenced by bind(), KbMode(), keyMap(), needsSave(), save(), and setNeedsUpdate().

12.33.4.2 UsbId KbMode::_id [private]

Definition at line 75 of file kbmode.h.

Referenced by id(), KbMode(), newId(), and save().

12.33.4.3 KbLight* KbMode::_light [private]

Definition at line 77 of file kbmode.h.

Referenced by KbMode(), keyMap(), light(), needsSave(), newId(), and save().

12.33.4.4 QString KbMode::_name [private]

Definition at line 74 of file kbmode.h.

Referenced by KbMode(), name(), and save().

12.33.4.5 bool KbMode::_needsSave [private]

Definition at line 81 of file kbmode.h.

Referenced by id(), keyMap(), name(), needsSave(), newId(), save(), and setNeedsSave().

12.33.4.6 KbPerf* KbMode::_perf [private]

Definition at line 79 of file kbmode.h.

Referenced by KbMode(), needsSave(), perf(), save(), and setNeedsUpdate().

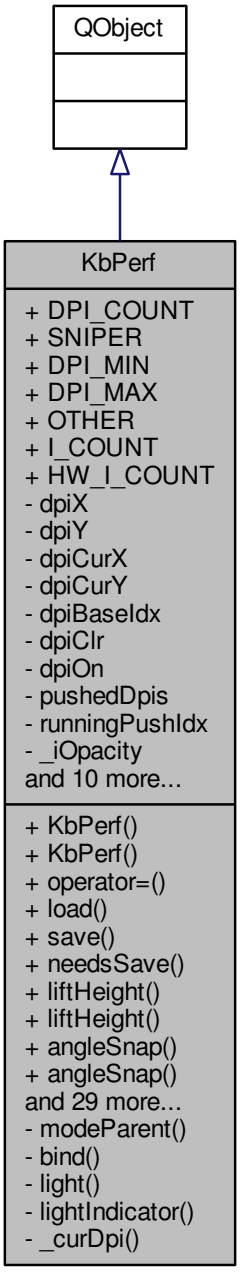
The documentation for this class was generated from the following files:

- src/ckb/kbmode.h
- src/ckb/kbmode.cpp
- src/ckb/release/moc_kbmode.cpp

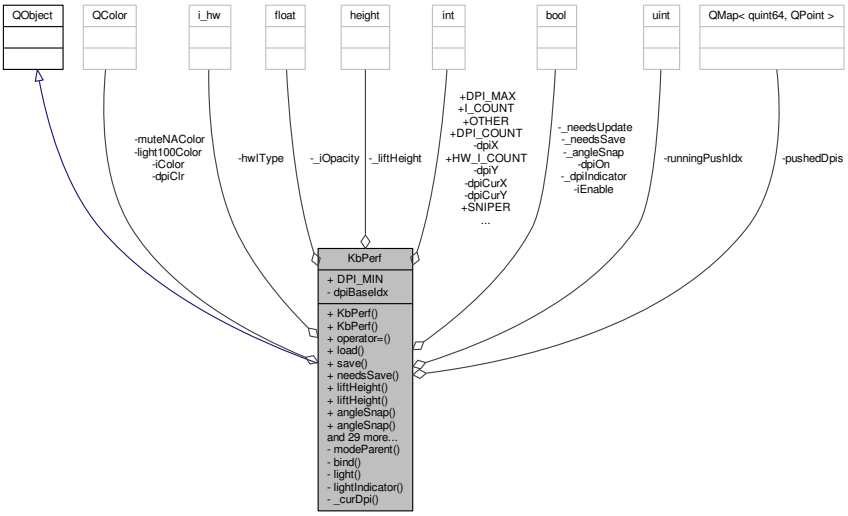
12.34 KbPerf Class Reference

`#include <src/ckb/kbperf.h>`

Inheritance diagram for KbPerf:



Collaboration diagram for KbPerf:



Public Types

- enum height {
  LOW = 1, LOWMED, MEDIUM, MEDHIGH,
  HIGH }
- enum indicator {
  NUM, CAPS, SCROLL, HW_IMAX = SCROLL,
  MODE, MACRO, LIGHT, LOCK,
  MUTE }
- enum i_hw { NONE = -1, NORMAL, ON, OFF }

Signals

- void didLoad ()
- void settingsUpdated ()

Public Member Functions

- KbPerf (KbMode *parent)
- KbPerf (KbMode *parent, const KbPerf &other)
- const KbPerf & operator= (const KbPerf &rhs)
- void load (CkbSettings &settings)
- void save (CkbSettings &settings)
- bool needsSave () const
- height liftHeight () const
- void liftHeight (height newHeight)
- bool angleSnap () const
- void angleSnap (bool newAngleSnap)
- QPoint dpi (int index) const
- void dpi (int index, const QPoint &newValue)
- QPoint sniperDpi () const
- void sniperDpi (const QPoint &newValue)

- void dpi (int index, int newValue)
- void sniperDpi (int newValue)
- int baseDpiIdx () const
- void baseDpiIdx (int newIdx)
- void dpiUp ()
- void dpiDown ()
- void dpiCycleUp ()
- void dpiCycleDown ()
- bool dpiEnabled (int index) const
- void dpiEnabled (int index, bool newEnabled)
- quint64 pushDpi (const QPoint &newDpi)
- quint64 pushDpi (int newDpi)
- quint64 pushSniper ()
- void popDpi (quint64 pushIdx)
- float iOpacity () const
- void iOpacity (float newIOpacity)
- bool dpiIndicator () const
- void dpiIndicator (bool newDpiIndicator)
- QColor dpiColor (int index) const
- void dpiColor (int index, const QColor &newColor)
- void getIndicator (indicator index, QColor &color1, QColor &color2, QColor &color3, bool &software_enable, i_hw &hardware_enable)
- void setIndicator (indicator index, const QColor &color1, const QColor &color2, const QColor &color3=QColor(), bool software_enable=true, i_hw hardware_enable=NORMAL)
- void update (QFile &cmd, int notifyNumber, bool force, bool saveCustomDpi)
- void setNeedsUpdate ()
- void applyIndicators (int modeIndex, const bool indicatorState[HW_I_COUNT])

Static Public Attributes

- static const int DPI_COUNT = 6
- static const int SNIPER = 0
- static const int DPI_MIN = 100
- static const int DPI_MAX = 12000
- static const int OTHER = DPI_COUNT
- static const int I_COUNT = (int)MUTE + 1
- static const int HW_I_COUNT = (int)HW_IMAX + 1

Private Member Functions

- KbMode * modeParent () const
- KbBind * bind () const
- KbLight * light () const
- void lightIndicator (const char *name, QRgb rgba)
- void _curDpi (const QPoint &newDpi)

Private Attributes

- int dpiX [DPI_COUNT]
- int dpiY [DPI_COUNT]
- int dpiCurX
- int dpiCurY
- int dpiBaseIdx
- QColor dpiClr [DPI_COUNT+1]
- bool dpiOn [DPI_COUNT]
- QMap< quint64, QPoint > pushedDpis
- uint runningPushIdx
- float _iOpacity
- QColor iColor [I_COUNT][2]
- QColor light100Color
- QColor muteNAColor
- bool iEnable [I_COUNT]
- i_hw hwIType [HW_I_COUNT]
- bool _dpiIndicator
- height _liftHeight
- bool _angleSnap
- bool _needsUpdate
- bool _needsSave

12.34.1 Detailed Description

Definition at line 15 of file kbperf.h.

12.34.2 Member Enumeration Documentation

12.34.2.1 enum KbPerf::height

Enumerator

LOW

LOWMED

MEDIUM

MEDHIGH

HIGH

Definition at line 31 of file kbperf.h.

```
31                  {
32          LOW = 1,
33          LOWMED,
34          MEDIUM,
35          MEDHIGH,
36          HIGH
37      };
```

12.34.2.2 enum KbPerf::i_hw

Enumerator

NONE

NORMAL

ON

OFF

Definition at line 101 of file kbperf.h.

```
101             {
102         NONE = -1,  // For non-hardware indicators
103         NORMAL,
104         ON,
105         OFF
106     };
```

12.34.2.3 enum KbPerf::indicator

Enumerator

NUM

CAPS

SCROLL

HW_IMAX

MODE

MACRO

LIGHT

LOCK

MUTE

Definition at line 88 of file kbperf.h.

```
88                  {
89         // Hardware
90         NUM,
91         CAPS,
92         SCROLL, HW_IMAX = SCROLL,
93         // Software
94         MODE,
95         MACRO,
96         LIGHT,
97         LOCK,
98         MUTE
99     };
```

12.34.3 Constructor & Destructor Documentation

12.34.3.1 KbPerf::KbPerf (KbMode * *parent*) [explicit]

Definition at line 13 of file kbperf.cpp.

References CAPS, DPI_COUNT, dpiBaseIdx, dpiClr, dpiCurX, dpiCurY, dpiOn, dpiX, dpiY, HW_I_COUNT, hwIType, I_COUNT, iColor, iEnable, LIGHT, light100Color, LOCK, MACRO, MODE, MUTE, muteNAColor, NORMAL, NUM, and SCROLL.

```
13                              :
14     QObject(parent), runningPushIdx(1),
15     _iOpacity(1.f), _dpiIndicator(true), _liftHeight(
       MEDIUM), _angleSnap(false),
16     _needsUpdate(true), _needsSave(true) {
17     // Default DPI settings
18     dpiX[0] = dpiY[0] = 400;
19     dpiX[1] = dpiY[1] = 450;
20     dpiX[2] = dpiY[2] = 800;
21     dpiX[3] = dpiY[3] = 1500;
22     dpiX[4] = dpiY[4] = 3000;
23     dpiX[5] = dpiY[5] = 6000;
24     dpiClr[0] = QColor(255, 0, 0);
25     dpiClr[1] = QColor(255, 192, 0);
26     dpiClr[2] = QColor(255, 255, 0);
27     dpiClr[3] = QColor(0, 255, 0);
28     dpiClr[4] = QColor(0, 255, 255);
29     dpiClr[5] = QColor(255, 255, 255);
30     dpiClr[6] = QColor(192, 192, 192);
31     for(int i = 0; i < DPI_COUNT; i++)
32         dpiOn[i] = true;
33     dpiBaseIdx = 3;
34     dpiCurX = dpiX[dpiBaseIdx];
35     dpiCurY = dpiY[dpiBaseIdx];
36     // Default indicators
37     iColor[NUM][0] = iColor[CAPS][0] = iColor[SCROLL][0] = QColor(0, 255, 0)
       ;   // Lock lights: on = green, off = black
38     iColor[NUM][1] = iColor[CAPS][1] = iColor[SCROLL][1] = QColor(0, 0, 0);
39     iColor[MACRO][0] = QColor(255, 0, 0);                                    // Macro: on =
       red, off = black
40     iColor[LOCK][0] = QColor(255, 255, 255);                                 // Win lock: on =
       white, off = black
41     iColor[MODE][0] = iColor[MUTE][0] = QColor(255, 255, 255, 0);            //
       Mode, mute: on = transparent, off = black
42     iColor[MODE][1] = iColor[MACRO][1] = iColor[LOCK][1] =
       iColor[MUTE][1] = muteNAColor = QColor(0, 0, 0);
43     iColor[LIGHT][0] = QColor(255, 0, 0);       // Brightness: red, yellow, white
44     iColor[LIGHT][1] = QColor(255, 255, 0);
45     light100Color = QColor(255, 255, 255);
46     for(int i = 0; i < HW_I_COUNT; i++){
47         // Set all lock lights to HW mode
48         iEnable[i] = false;
49         hwIType[i] = NORMAL;
50     }
51     for(int i = HW_I_COUNT; i < I_COUNT; i++){
52         // Turn all other indicators on
53         iEnable[i] = true;
54     }
55 }
```

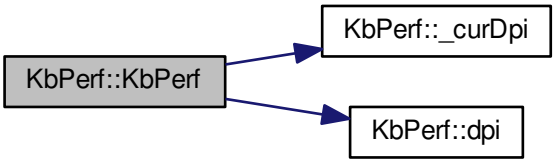
12.34.3.2 KbPerf::KbPerf (KbMode * *parent*, const KbPerf & *other*)

Definition at line 57 of file kbperf.cpp.

References _curDpi(), dpi(), DPI_COUNT, dpiBaseIdx, dpiClr, dpiOn, dpiX, dpiY, hwIType, I_COUNT, iColor, and iEnable.

```
57                                              :
58     QObject(parent), dpiCurX(other.dpiCurX), dpiCurY(other.
       dpiCurY), dpiBaseIdx(other.dpiBaseIdx),
       runningPushIdx(1),
59     _iOpacity(other._iOpacity), light100Color(other.
       light100Color), muteNAColor(other.muteNAColor),
       _dpiIndicator(other._dpiIndicator),
60     _liftHeight(other._liftHeight), _angleSnap(other.
       _angleSnap),
61     _needsUpdate(true), _needsSave(true) {
62     memcpy(dpiX, other.dpiX, sizeof(dpiX));
63     memcpy(dpiY, other.dpiY, sizeof(dpiY));
64     for(int i = 0; i < DPI_COUNT + 1; i++)
65         dpiClr[i] = other.dpiClr[i];
66     memcpy(dpiOn, other.dpiOn, sizeof(dpiOn));
67     for(int i = 0; i < I_COUNT; i++){
68         iColor[i][0] = other.iColor[i][0];
69         iColor[i][1] = other.iColor[i][1];
70     }
71     memcpy(iEnable, other.iEnable, sizeof(iEnable));
72     memcpy(hwIType, other.hwIType, sizeof(hwIType));
73     // Don't copy pushed DPI states. If the other mode has any, restore the original DPI
74     _curDpi(dpi(dpiBaseIdx));
75 }
```

Here is the call graph for this function:



12.34.4 Member Function Documentation

12.34.4.1 void KbPerf::_curDpi (const QPoint & *newDpi*) [private]

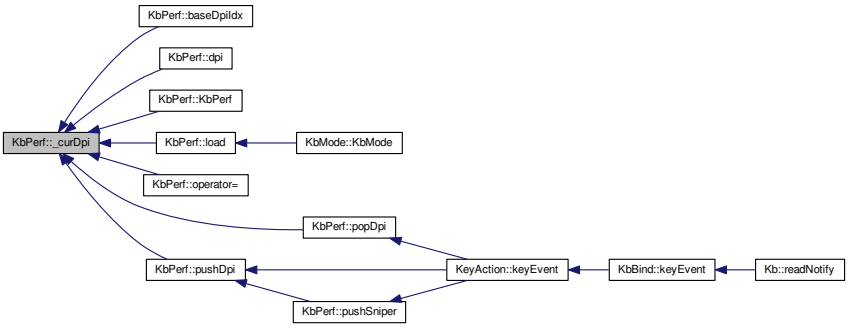
Definition at line 259 of file kbperf.cpp.

References _needsSave, _needsUpdate, dpiCurX, and dpiCurY.

Referenced by baseDpiIdx(), dpi(), KbPerf(), load(), operator=(), popDpi(), and pushDpi().

```
259                                         {
260     dpiCurX = newDpi.x();
261     dpiCurY = newDpi.y();
262     _needsUpdate = _needsSave = true;
263 }
```

Here is the caller graph for this function:



12.34.4.2 bool KbPerf::angleSnap ( ) const [inline]

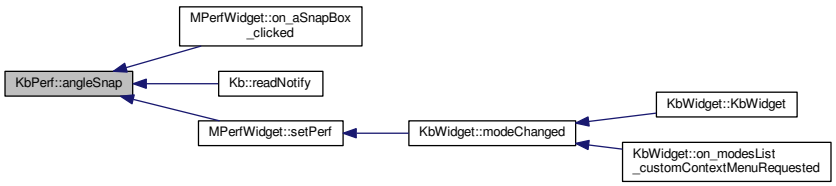
Definition at line 42 of file kbperf.h.

References _angleSnap.

Referenced by MPerfWidget::on_aSnapBox_clicked(), Kb::readNotify(), and MPerfWidget::setPerf().

```
42 { return _angleSnap; }
```

Here is the caller graph for this function:



12.34.4.3 void KbPerf::angleSnap (bool *newAngleSnap*)

Definition at line 380 of file kbperf.cpp.

References _angleSnap, _needsSave, and _needsUpdate.

```
380                                         {
381     _angleSnap = newAngleSnap;
382     _needsUpdate = _needsSave = true;
383 }
```

12.34.4.4 void KbPerf::applyIndicators (int *modeIndex,* const bool *indicatorState[HW_I_COUNT]*)

Definition at line 439 of file kbperf.cpp.

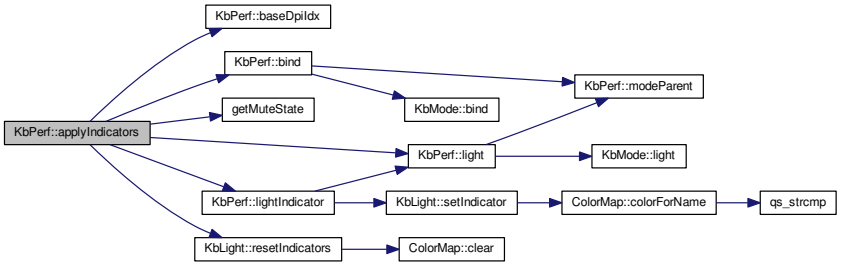
References _dpiIndicator, _iOpacity, baseDpiIdx(), bind(), CAPS, dpiClr, getMuteState(), Kb::HWMODE_MAX, iColor, iEnable, LIGHT, light(), light100Color, lightIndicator(), LOCK, MACRO, MODE, MUTE, MUTED, muteNAColor, NUM, OTHER, KbLight::resetIndicators(), SCROLL, and UNMUTED.

Referenced by Kb::frameUpdate().

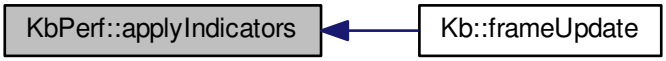
```
439                                                                         {
440     light()->resetIndicators();
441     if(_iOpacity <= 0.f)
442         return;
443     if(_dpiIndicator){
444         // Set DPI indicator according to index
445         int index = baseDpiIdx();
446         if(index == -1 || index > OTHER)
447             index = OTHER;
448         lightIndicator("dpi", dpiClr[index].rgba());
449     }
450     // KB indicators
451     if(iEnable[MODE]){
452         for(int i = 0; i < Kb::HWMODE_MAX; i++){
453             char name[4];
454             snprintf(name, sizeof(name), "m%d", i + 1);
455             if(modeIndex == i)
456                 lightIndicator(name, iColor[MODE][0].rgba());
457             else
458                 lightIndicator(name, iColor[MODE][1].rgba());
459         }
460     }
461     if(iEnable[MACRO])
462         lightIndicator("mr", iColor[MUTE][1].rgba());
463     if(iEnable[LIGHT]){
464         switch(light()->dimming()){
465         case 0: // 100%
466             lightIndicator("light", light100Color.rgba());
467             break;
468         case 1: // 67%
469             lightIndicator("light", iColor[LIGHT][1].rgba());
470             break;
471         case 2: // 33%
472         case 3: // light off
473             lightIndicator("light", iColor[LIGHT][0].rgba());
474             break;
```

```
475         }
476     }
477     if(iEnable[LOCK]){
478         if(bind()->winLock())
479             lightIndicator("lock", iColor[LOCK][0].rgba());
480         else
481             lightIndicator("lock", iColor[LOCK][1].rgba());
482     }
483     if(iEnable[MUTE]){
484         switch(getMuteState()){
485         case MUTED:
486             lightIndicator("mute", iColor[MUTE][0].rgba());
487             break;
488         case UNMUTED:
489             lightIndicator("mute", iColor[MUTE][1].rgba());
490             break;
491         default:
492             lightIndicator("mute", muteNAColor.rgba());
493             break;
494         }
495     }
496     // Lock lights
497     if(iEnable[NUM])
498         lightIndicator("numlock", indicatorState[0] ? iColor[NUM][0].rgba() :
    iColor[NUM][1].rgba());
499     if(iEnable[CAPS])
500         lightIndicator("caps", indicatorState[1] ? iColor[CAPS][0].rgba() :
    iColor[CAPS][1].rgba());
501     if(iEnable[SCROLL])
502         lightIndicator("scroll", indicatorState[2] ? iColor[SCROLL][0].rgba() :
    iColor[SCROLL][1].rgba());
503 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.34.4.5 int KbPerf::baseDpiIdx ( ) const [inline]

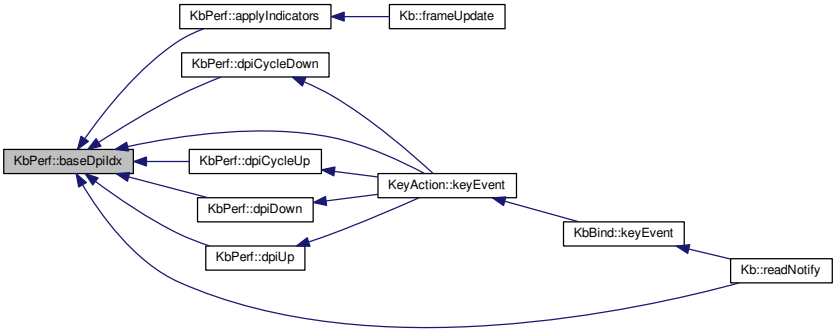
Definition at line 61 of file kbperf.h.

References dpiBaseIdx.

Referenced by applyIndicators(), dpiCycleDown(), dpiCycleUp(), dpiDown(), dpiUp(), KeyAction::keyEvent(), and Kb::readNotify().

```
61 { return dpiBaseIdx; }
```

Here is the caller graph for this function:



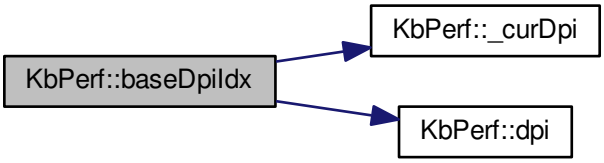
12.34.4.6 void KbPerf::baseDpiIdx (int *newIdx*)

Definition at line 265 of file kbperf.cpp.

References _curDpi(), _needsSave, _needsUpdate, dpi(), dpiBaseIdx, and pushedDpis.

```
265                                 {
266     if (pushedDpis.isEmpty() && dpiBaseIdx == newIdx)
267         return;
268     pushedDpis.clear();
269     dpiBaseIdx = newIdx;
270     _curDpi(dpi(dpiBaseIdx));
271     _needsUpdate = _needsSave = true;
272 }
```

Here is the call graph for this function:



12.34.4.7 KbBind * KbPerf::bind ( ) const [private]

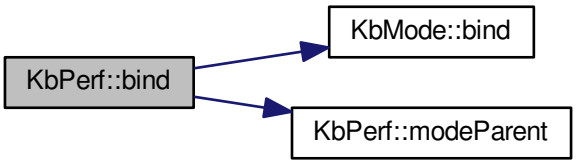
Definition at line 98 of file kbperf.cpp.

References KbMode::bind(), and modeParent().

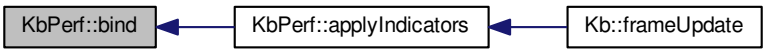
Referenced by applyIndicators().

```
98                              {
99      return modeParent()->bind();
100 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.34.4.8 void KbPerf::didLoad ( ) [signal]

Definition at line 134 of file moc_kbperf.cpp.

Referenced by load().

```
135 {
136     QMetaObject::activate(this, &staticMetaObject, 0, Q_NULLPTR);
137 }
```

Here is the caller graph for this function:

12.34.4.9 QPoint KbPerf::dpi (int *index*) const [inline]

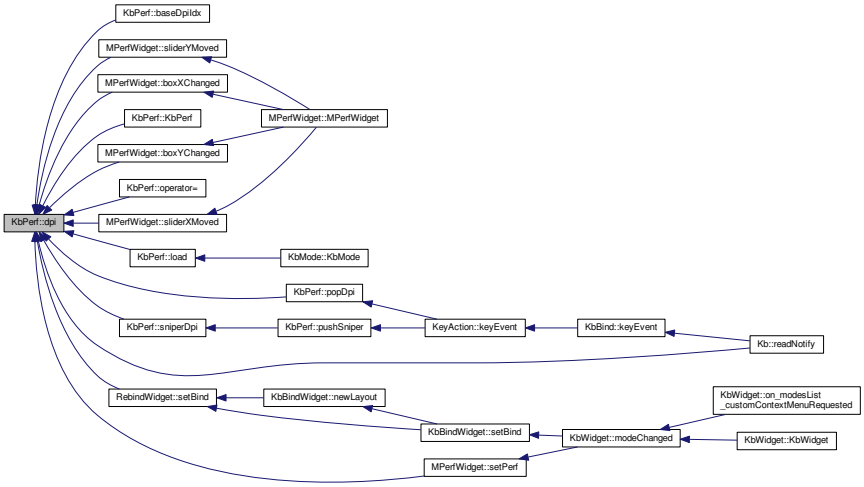
Definition at line 50 of file kbperf.h.

References DPI_COUNT, dpiX, and dpiY.

Referenced by baseDpiIdx(), MPerfWidget::boxXChanged(), MPerfWidget::boxYChanged(), KbPerf(), load(), operator=(), popDpi(), Kb::readNotify(), RebindWidget::setBind(), MPerfWidget::setPerf(), MPerfWidget::sliderXMoved(), MPerfWidget::sliderYMoved(), and sniperDpi().

```
50 { if(index < 0 || index >= DPI_COUNT) return QPoint(); return QPoint(
      dpiX[index], dpiY[index]); }
```

Here is the caller graph for this function:



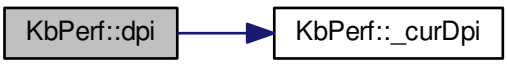
12.34.4.10 void KbPerf::dpi (int *index*, const QPoint & *newValue*)

Definition at line 247 of file kbperf.cpp.

References _curDpi(), _needsSave, _needsUpdate, DPI_COUNT, dpiBaseIdx, dpiX, dpiY, and pushedDpis.

```
247                                                 {
248     if(index < 0 || index >= DPI_COUNT)
249         return;
250     dpiX[index] = newValue.x();
251     dpiY[index] = newValue.y();
252     // Update current DPI if needed
253     if(dpiBaseIdx == index && pushedDpis.isEmpty()) {
254         _curDpi(QPoint(dpiX[index], dpiY[index]));
255     }
256     _needsUpdate = _needsSave = true;
257 }
```

Here is the call graph for this function:



12.34.4.11 void KbPerf::dpi (int *index*, int *newValue*) [inline]

Definition at line 55 of file kbperf.h.

References dpi().

Referenced by dpi().

```
55 { dpi(index, QPoint(newValue, newValue)); }
```

Here is the call graph for this function:

Here is the caller graph for this function:

12.34.4.12 QColor KbPerf::dpiColor (int *index*) const [inline]

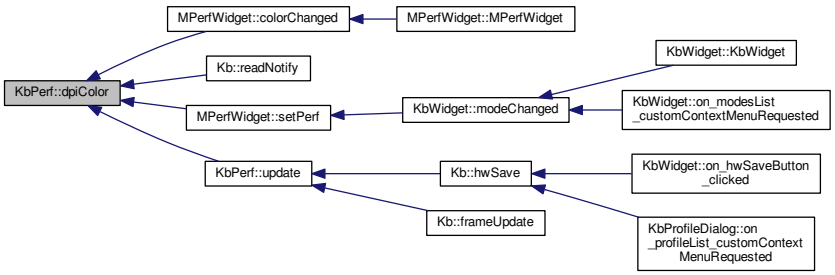
Definition at line 85 of file kbperf.h.

References dpiClr.

Referenced by MPerfWidget::colorChanged(), Kb::readNotify(), MPerfWidget::setPerf(), and update().

```
85 { return dpiClr[index]; }
```

Here is the caller graph for this function:



12.34.4.13 void KbPerf::dpiColor (int *index*, const QColor & *newColor*) [inline]

Definition at line 86 of file kbperf.h.

References _needsSave, _needsUpdate, and dpiClr.

```
86 { dpiClr[index] = newColor; _needsUpdate = _needsSave = true; }
```

12.34.4.14 void KbPerf::dpiCycleDown ( )

Definition at line 329 of file kbperf.cpp.

References baseDpiIdx(), DPI_COUNT, dpiOn, and SNIPER.

Referenced by KeyAction::keyEvent().

```
329                         {
330     int idx = baseDpiIdx();
331     do {
332         idx--;
333         if(idx <= SNIPER)
334             idx = DPI_COUNT - 1;
335     if(idx == baseDpiIdx())
336             return;
337     } while(!dpiOn[idx]);
338     baseDpiIdx(idx);
339 }
```

Here is the call graph for this function:

Here is the caller graph for this function:

12.34.4.15 void KbPerf::dpiCycleUp ( )

Definition at line 317 of file kbperf.cpp.

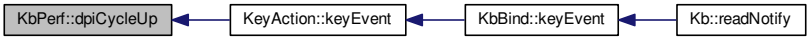
References baseDpiIdx(), DPI_COUNT, dpiOn, and SNIPER.

Referenced by KeyAction::keyEvent().

```
317                         {
318     int idx = baseDpiIdx();
319     do {
320         idx++;
321         if(idx >= DPI_COUNT)
322             idx = SNIPER + 1;
323         if(idx == baseDpiIdx())
324             return;
325     } while(!dpiOn[idx]);
326     baseDpiIdx(idx);
327 }
```

Here is the call graph for this function:

Here is the caller graph for this function:



12.34.4.16 void KbPerf::dpiDown ( )

Definition at line 306 of file kbperf.cpp.

References baseDpiIdx(), dpiOn, and SNIPER.

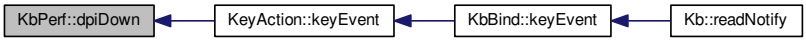
Referenced by KeyAction::keyEvent().

```
306                       {
```

```
307     int idx = baseDpiIdx();
308     do {
309         idx--;
310         if(idx <= SNIPER)
311             return;
312     } while(!dpiOn[idx]);
313     baseDpiIdx(idx);
314 }
```

Here is the call graph for this function:

Here is the caller graph for this function:



12.34.4.17 bool KbPerf::dpiEnabled (int *index*) const [inline]

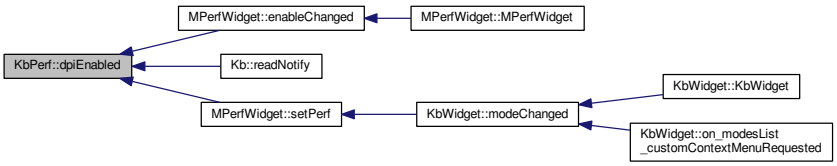
Definition at line 68 of file kbperf.h.

References dpiOn.

Referenced by MPerfWidget::enableChanged(), Kb::readNotify(), and MPerfWidget::setPerf().

```
68 { return dpiOn[index]; }
```

Here is the caller graph for this function:



12.34.4.18 void KbPerf::dpiEnabled (int *index*, bool *newEnabled*) [inline]

Definition at line 69 of file kbperf.h.

References _needsSave, _needsUpdate, and dpiOn.

```
69 { if(index <= 0) return; dpiOn[index] = newEnabled; _needsUpdate =
      _needsSave = true; }
```

12.34.4.19 bool KbPerf::dpiIndicator ( ) const [inline]

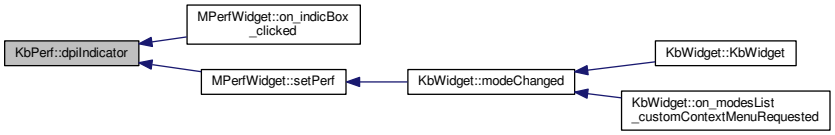
Definition at line 82 of file kbperf.h.

References _dpiIndicator.

Referenced by MPerfWidget::on_indicBox_clicked(), and MPerfWidget::setPerf().

```
82 { return _dpiIndicator; }
```

Here is the caller graph for this function:



12.34.4.20 void KbPerf::dpiIndicator (bool *newDpiIndicator*) [inline]

Definition at line 83 of file kbperf.h.

References _dpiIndicator, and _needsSave.

```
83 { _dpiIndicator = newDpiIndicator; _needsSave = true; }
```

12.34.4.21 void KbPerf::dpiUp ( )

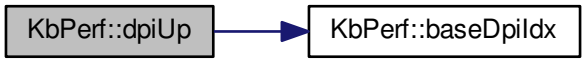
Definition at line 295 of file kbperf.cpp.

References baseDpiIdx(), DPI_COUNT, and dpiOn.

Referenced by KeyAction::keyEvent().

```
295                 {
296     // Scroll past disabled DPIs and choose the next one up
297     int idx = baseDpiIdx();
298     do {
299         idx++;
300         if(idx >= DPI_COUNT)
301             return;
302     } while(!dpiOn[idx]);
303     baseDpiIdx(idx);
304 }
```

Here is the call graph for this function:



Here is the caller graph for this function:

12.34.4.22 void KbPerf::getIndicator (indicator *index*, QColor & *color1*, QColor & *color2*, QColor & *color3*, bool & *software_enable*, i_hw & *hardware_enable*)

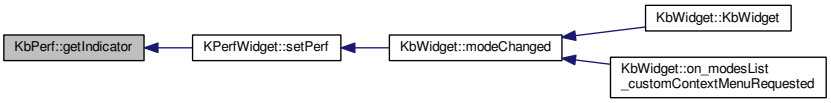
Definition at line 341 of file kbperf.cpp.

References HW_IMAX, hwIType, I_COUNT, iColor, iEnable, LIGHT, light100Color, MUTE, muteNAColor, and NONE.

Referenced by KPerfWidget::setPerf().

```
341
                                {
342     if(index < 0 || index >= I_COUNT)
343         return;
344     color1 = iColor[index][0];
345     color2 = iColor[index][1];
346     if(index == LIGHT)
347         color3 = light100Color;
348     else if(index == MUTE)
349         color3 = muteNAColor;
350     else
351         color3 = QColor();
352     software_enable = iEnable[index];
353     hardware_enable = (index <= HW_IMAX ? hwIType[index] : NONE);
354 }
```

Here is the caller graph for this function:



12.34.4.23 float KbPerf::iOpacity ( ) const [inline]

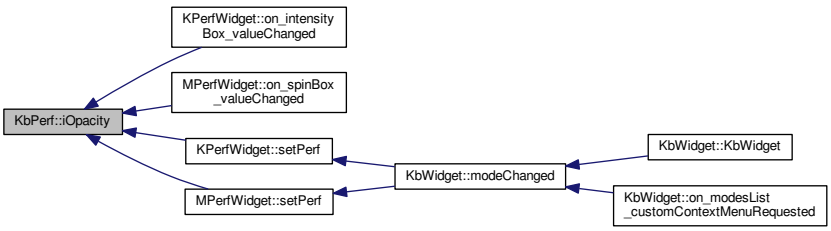
Definition at line 79 of file kbperf.h.

References _iOpacity.

Referenced by KPerfWidget::on_intensityBox_valueChanged(), MPerfWidget::on_spinBox_valueChanged(), KPerfWidget::setPerf(), and MPerfWidget::setPerf().

```
79 { return _iOpacity; }
```

Here is the caller graph for this function:



12.34.4.24 void KbPerf::iOpacity (float *newIOpacity*) [inline]

Definition at line 80 of file kbperf.h.

References _iOpacity, and _needsSave.

```
80 { _iOpacity = newIOpacity; _needsSave = true; }
```

12.34.4.25 height KbPerf::liftHeight ( ) const [inline]

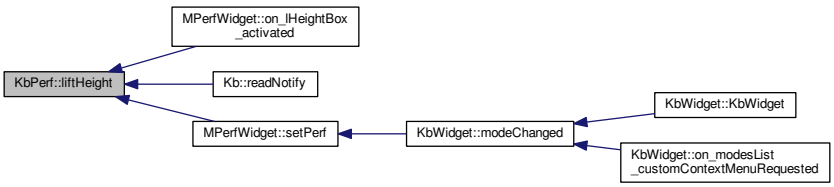
Definition at line 38 of file kbperf.h.

References _liftHeight.

Referenced by MPerfWidget::on_lHeightBox_activated(), Kb::readNotify(), and MPerfWidget::setPerf().

```
38 { return _liftHeight; }
```

Here is the caller graph for this function:



12.34.4.26 void KbPerf::liftHeight (height *newHeight*)

Definition at line 373 of file kbperf.cpp.

References _liftHeight, _needsSave, _needsUpdate, and HIGH.

```
373                                     {
374     if(newHeight < LOW || newHeight > HIGH)
375         return;
376     _liftHeight = newHeight;
377     _needsUpdate = _needsSave = true;
378 }
```

12.34.4.27 KbLight * KbPerf::light ( ) const [private]

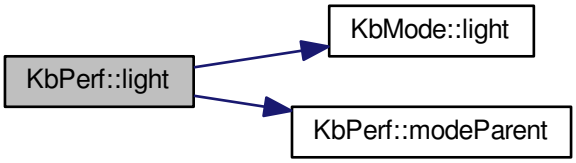
Definition at line 102 of file kbperf.cpp.

References KbMode::light(), and modeParent().

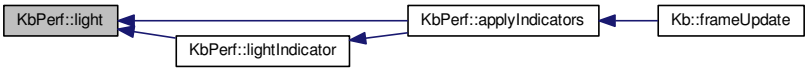
Referenced by applyIndicators(), and lightIndicator().

```
102                                {
103     return modeParent()->light();
104 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.34.4.28 void KbPerf::lightIndicator (const char * *name*, QRgb *rgba*) [private]

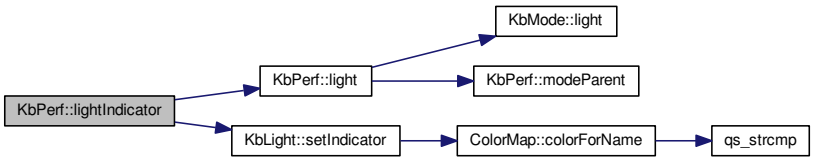
Definition at line 432 of file kbperf.cpp.

References _iOpacity, light(), and KbLight::setIndicator().

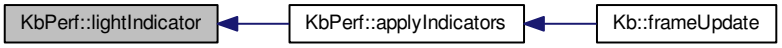
Referenced by applyIndicators().

```
432                                                    {
433     int a = round(qAlpha(rgba) * _iOpacity);
434     if(a <= 0)
435         return;
436     light()->setIndicator(name, qRgba(qRed(rgba), qGreen(rgba), qBlue(rgba), a));
437 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.34.4.29 void KbPerf::load (CkbSettings & *settings*)

Definition at line 106 of file kbperf.cpp.

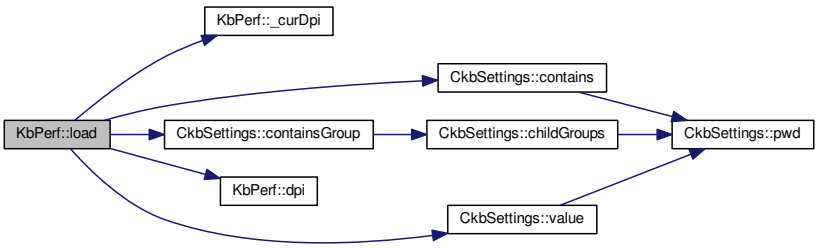
References _angleSnap, _curDpi(), _dpiIndicator, _iOpacity, _liftHeight, _needsSave, CkbSettings::contains(), CkbSettings::containsGroup(), didLoad(), dpi(), DPI_COUNT, dpiBaseIdx, dpiClr, dpiOn, dpiX, dpiY, HIGH, HW_IMAX, hwIType, I_COUNT, iColor, iEnable, LIGHT, light100Color, LOCK, MACRO, MEDIUM, MODE, MUTE, muteNAColor, NORMAL, OTHER, pushedDpis, runningPushIdx, and CkbSettings::value().

Referenced by KbMode::KbMode().

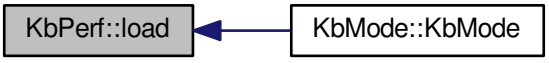
```
106                                     {
107     pushedDpis.clear();
108     runningPushIdx = 1;
109     _needsSave = false;
110     bool readIndicators = true;
111     if(!settings.containsGroup("Performance/Indicators")){
112         // Read old indicator settings from the lighting group, if present
113         // (ckb <= v0.2.0)
114         SGroup group(settings, "Lighting");
115         if(settings.contains("InactiveIndicators")){
116             bool inOk = false;
117             int inactive = settings.value("InactiveIndicators").toInt(&inOk);
118             if(!inOk || inactive > 2)
119                 inactive = 2;
120             if(inactive == 1)
121                 _iOpacity = 0.75f;
122             else if(inactive == 0)
123                 _iOpacity = 0.5f;
124             else if(inactive < 0){
125                 // Indicators disabled
126                 iEnable[MODE] = iEnable[MACRO] = iEnable[
      LIGHT] = iEnable[LOCK]  = iEnable[MUTE] = false;
127             }
128             bool showMute = (settings.value("ShowMute").toInt(&inOk) != 0);
129             if(inOk && !showMute)
130                 iEnable[MUTE] = false;
131             readIndicators = false;
132         }
133     }
134     SGroup group(settings, "Performance");
135     // Read DPI settings
136     {
137         SGroup group(settings, "DPI");
138         for(int i = 0; i < DPI_COUNT; i++){
139             QString iStr = QString::number(i);
```

```
140             QPoint value = settings.value(iStr).toPoint();
141             if(value.isNull())
142                 continue;
143             dpiX[i] = value.x(); dpiY[i] = value.y();
144             QColor color = settings.value(iStr + "RGB").toString();
145             if(color.isValid())
146                 dpiClr[i] = color;
147             if(i != 0)
148                 dpiOn[i] = !settings.value(iStr + "Disabled").toBool();
149         }
150         QColor color = settings.value("6RGB").toString();
151         if(color.isValid())
152             dpiClr[OTHER] = color;
153         if (settings.contains("CurIdx")) {
154             dpiBaseIdx = settings.value("CurIdx").toInt();
155         } else {
156             // If there isn't a setting for current DPI stage, pick the first
157             // enabled one. Failing that just pick stage 1.
158             dpiBaseIdx = 1;
159             for (int i = 1; i < DPI_COUNT; i++) {
160                 if (dpiOn[i]) {
161                     dpiBaseIdx = i;
162                     break;
163                 }
164             }
165         }
166         _curDpi(dpi(dpiBaseIdx));
167     }
168     // Read misc. mouse settings
169     _liftHeight = (height)settings.value("LiftHeight").toInt();
170     if(_liftHeight < LOW || _liftHeight > HIGH)
171         _liftHeight = MEDIUM;
172     _angleSnap = settings.value("AngleSnap").toBool();
173     if(settings.contains("NoIndicator")){
174         // ckb <= v0.2.0
175         _dpiIndicator = !settings.value("NoIndicator").toBool();
176     } else {
177         _dpiIndicator = settings.value("Indicators/DPI", true).toBool();
178     }
179     // Read indicator settings
180     if(readIndicators){
181         SGroup group(settings, "Indicators");
182         _iOpacity = settings.value("Opacity", 100).toInt() / 100.f;
183         for(int i = 0; i < I_COUNT; i++){
184             SGroup group(settings, QString::number(i));
185             QColor color = settings.value("RGB0").toString();
186             if(color.isValid())
187                 iColor[i][0] = color;
188             color = settings.value("RGB1").toString();
189             if(color.isValid())
190                 iColor[i][1] = color;
191             if(i == LIGHT){
192                 color = settings.value("RGB2").toString();
193                 if(color.isValid())
194                     light100Color = color;
195             } else if(i == MUTE){
196                 color = settings.value("RGB2").toString();
197                 if(color.isValid())
198                     muteNAColor = color;
199             }
200             if(i <= HW_IMAX){
201                 iEnable[i] = settings.value("Enable", false).toBool();
202                 hwIType[i] = (i_hw)settings.value("Hardware", (int)
    NORMAL).toInt();
203             } else {
204                 iEnable[i] = settings.value("Enable", true).toBool();
205             }
206         }
207     }
208     emit didLoad();
209 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



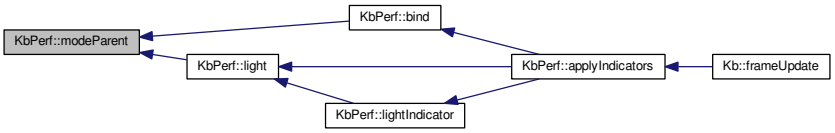
12.34.4.30 KbMode* KbPerf::modeParent ( ) const [inline], [private]

Definition at line 127 of file kbperf.h.

Referenced by bind(), and light().

```
127 { return (KbMode*)parent(); }
```

Here is the caller graph for this function:



12.34.4.31 bool KbPerf::needsSave ( ) const [inline]

Definition at line 28 of file kbperf.h.

References _needsSave.

Referenced by KbMode::needsSave().

```
28 { return _needsSave; }
```

Here is the caller graph for this function:

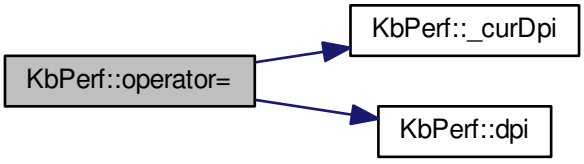
12.34.4.32 const KbPerf & KbPerf::operator= (const KbPerf & *rhs*)

Definition at line 77 of file kbperf.cpp.

References _angleSnap, _curDpi(), _dpiIndicator, _iOpacity, _liftHeight, _needsSave, _needsUpdate, dpi(), DPI_COUNT, dpiBaseIdx, dpiClr, dpiCurX, dpiCurY, dpiOn, dpiX, dpiY, hwIType, I_COUNT, iColor, iEnable, light100Color, muteNAColor, and runningPushIdx.

```
77                                                  {
78     dpiCurX = other.dpiCurX; dpiCurY = other.dpiCurY; dpiBaseIdx = other.dpiBaseIdx
       ; runningPushIdx = 1;
79     _iOpacity = other._iOpacity; light100Color = other.light100Color;
       muteNAColor = other.muteNAColor; _dpiIndicator = other._dpiIndicator;
80     _liftHeight = other._liftHeight; _angleSnap = other._angleSnap;
81     _needsUpdate = true; _needsSave = true;
82     memcpy(dpiX, other.dpiX, sizeof(dpiX));
83     memcpy(dpiY, other.dpiY, sizeof(dpiY));
84     for(int i = 0; i < DPI_COUNT + 1; i++)
85         dpiClr[i] = other.dpiClr[i];
86     memcpy(dpiOn, other.dpiOn, sizeof(dpiOn));
87     for(int i = 0; i < I_COUNT; i++){
88         iColor[i][0] = other.iColor[i][0];
89         iColor[i][1] = other.iColor[i][1];
90     }
91     memcpy(iEnable, other.iEnable, sizeof(iEnable));
92     memcpy(hwIType, other.hwIType, sizeof(hwIType));
93     // Don't copy pushed DPI states. If the other mode has any, restore the original DPI
94     _curDpi(dpi(dpiBaseIdx));
95     return other;
96 }
```

Here is the call graph for this function:



12.34.4.33 void KbPerf::popDpi (quint64 *pushIdx*)

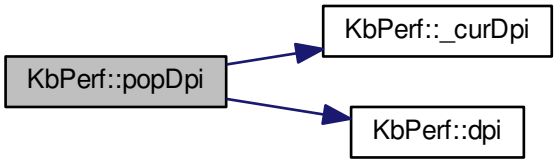
Definition at line 281 of file kbperf.cpp.

References _curDpi(), _needsSave, _needsUpdate, dpi(), dpiBaseIdx, map_last, and pushedDpis.

Referenced by KeyAction::keyEvent().

```
281                                 {
282     if(!pushedDpis.contains(pushIdx)) {
283       return;
284     }
285     pushedDpis.remove(pushIdx);
286     if (pushedDpis.isEmpty()) {
287         _curDpi(dpi(dpiBaseIdx));
288     } else {
289         // Set the DPI to the last-pushed value still on the stack
290         _curDpi(map_last(pushedDpis));
291     }
292     _needsUpdate = _needsSave = true;
293 }
```

Here is the call graph for this function:



Here is the caller graph for this function:

12.34.4.34 quint64 KbPerf::pushDpi (const QPoint & *newDpi*)

Definition at line 274 of file kbperf.cpp.

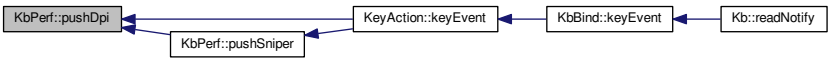
References _curDpi(), pushedDpis, and runningPushIdx.

Referenced by KeyAction::keyEvent(), and pushSniper().

```
274                                         {
275     quint64 index = runningPushIdx++;
276     pushedDpis[index] = newDpi;
277     _curDpi(newDpi);
278     return index;
279 }
```

Here is the call graph for this function:

Here is the caller graph for this function:



12.34.4.35 quint64 KbPerf::pushDpi (int *newDpi*) [inline]

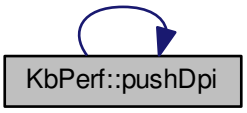
Definition at line 74 of file kbperf.h.

References pushDpi().

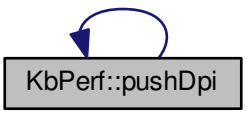
Referenced by pushDpi().

```
74 { return pushDpi(QPoint(newDpi, newDpi)); }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.34.4.36 quint64 KbPerf::pushSniper ( ) [inline]

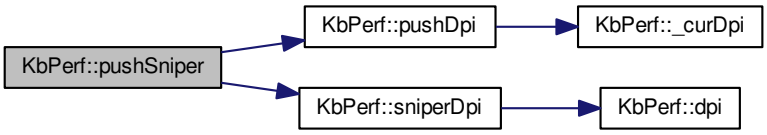
Definition at line 75 of file kbperf.h.

References pushDpi(), and sniperDpi().

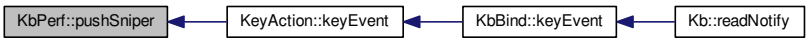
Referenced by KeyAction::keyEvent().

```
75 { return pushDpi(sniperDpi()); }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.34.4.37 void KbPerf::save (CkbSettings & *settings*)

Definition at line 211 of file kbperf.cpp.

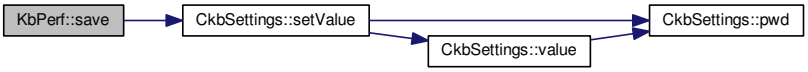
References _angleSnap, _dpiIndicator, _liftHeight, _needsSave, DPI_COUNT, dpiBaseIdx, dpiClr, dpiOn, dpiX, dpiY, HW_IMAX, hwIType, I_COUNT, iColor, iEnable, LIGHT, light100Color, MUTE, muteNAColor, OTHER, and CkbSettings::setValue().

Referenced by KbMode::save().

```
211                                     {
212     _needsSave = false;
213     SGroup group(settings, "Performance");
214     {
215         SGroup group(settings, "DPI");
216         for(int i = 0; i < DPI_COUNT; i++){
217             QString iStr = QString::number(i);
218             settings.setValue(iStr, QPoint(dpiX[i], dpiY[i]));
219             settings.setValue(iStr + "RGB", dpiClr[i].name(QColor::HexArgb));
220             if(i != 0)
221                 settings.setValue(iStr + "Disabled", !dpiOn[i]);
222         }
223         settings.setValue("6RGB", dpiClr[OTHER].name(QColor::HexArgb));
224     // Ignore pushed modes when saving current DPI.
225         settings.setValue("CurIdx", dpiBaseIdx);
226     }
227     settings.setValue("LiftHeight", _liftHeight);
228     settings.setValue("AngleSnap", _angleSnap);
229     {
230         SGroup group(settings, "Indicators");
231         settings.setValue("DPI", _dpiIndicator);
232         for(int i = 0; i < I_COUNT; i++){
233             SGroup group(settings, QString::number(i));
```

```
234             settings.setValue("RGB0", iColor[i][0].name(QColor::HexArgb));
235             settings.setValue("RGB1", iColor[i][1].name(QColor::HexArgb));
236             if(i == LIGHT)
237                 settings.setValue("RGB2", light100Color.name(QColor::HexArgb));
238             else if(i == MUTE)
239                 settings.setValue("RGB2", muteNAColor.name(QColor::HexArgb));
240             settings.setValue("Enable", iEnable[i]);
241             if(i <= HW_IMAX)
242                 settings.setValue("Hardware", (int)hwIType[i]);
243         }
244     }
245 }
```

Here is the call graph for this function:



Here is the caller graph for this function:

12.34.4.38 void KbPerf::setIndicator (indicator *index*, const QColor & *color1*, const QColor & *color2*, const QColor & *color3* = `QColor()`, bool *software_enable* = `true`, i_hw *hardware_enable* = NORMAL)

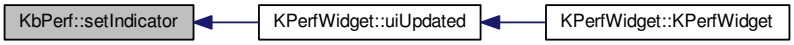
Definition at line 356 of file kbperf.cpp.

References _needsSave, _needsUpdate, HW_IMAX, hwIType, I_COUNT, iColor, iEnable, LIGHT, light100Color, MUTE, muteNAColor, NONE, and NORMAL.

Referenced by KPerfWidget::uiUpdated().

```
356                                                     {
357     if(index < 0 || index >= I_COUNT)
358         return;
359     iColor[index][0] = color1;
360     iColor[index][1] = color2;
361     if(index == LIGHT)
362         light100Color = color3;
363     else if(index == MUTE)
364         muteNAColor = color3;
365     iEnable[index] = software_enable;
366     if(hardware_enable == NONE)
367         hardware_enable = NORMAL;
368     if(index <= HW_IMAX)
369         hwIType[index] = hardware_enable;
370     _needsUpdate = _needsSave = true;
371 }
```

Here is the caller graph for this function:



12.34.4.39 void KbPerf::setNeedsUpdate ( ) [inline]

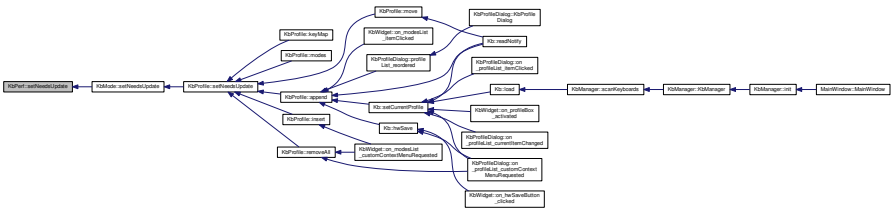
Definition at line 116 of file kbperf.h.

References _needsUpdate.

Referenced by KbMode::setNeedsUpdate().

```
116 { _needsUpdate = true; }
```

Here is the caller graph for this function:



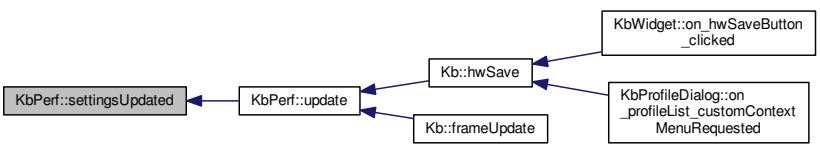
12.34.4.40 void KbPerf::settingsUpdated ( ) [signal]

Definition at line 140 of file moc_kbperf.cpp.

Referenced by update().

```
141 {
142     QMetaObject::activate(this, &staticMetaObject, 1, Q_NULLPTR);
143 }
```

Here is the caller graph for this function:



12.34.4.41 QPoint KbPerf::sniperDpi ( ) const [inline]

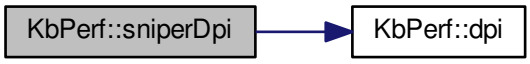
Definition at line 52 of file kbperf.h.

References dpi(), and SNIPER.

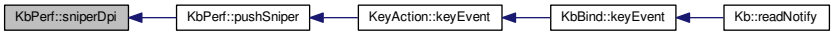
Referenced by pushSniper().

```
52 { return dpi(SNIPER); }
```

Here is the call graph for this function:



Here is the caller graph for this function:



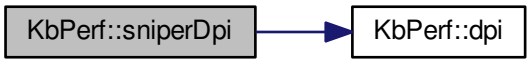
12.34.4.42 void KbPerf::sniperDpi (const QPoint & *newValue*) [inline]

Definition at line 53 of file kbperf.h.

References dpi(), and SNIPER.

```
53 { dpi(SNIPER, newValue); }
```

Here is the call graph for this function:



12.34.4.43 void KbPerf::sniperDpi (int *newValue*) [inline]

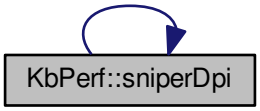
Definition at line 56 of file kbperf.h.

References sniperDpi().

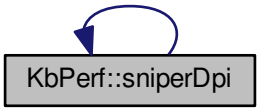
Referenced by sniperDpi().

```
56 { sniperDpi(QPoint(newValue, newValue)); }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.34.4.44 void KbPerf::update (QFile & *cmd,* int *notifyNumber,* bool *force,* bool *saveCustomDpi*)

Definition at line 385 of file kbperf.cpp.

References _angleSnap, _liftHeight, _needsUpdate, DPI_COUNT, dpiBaseIdx, dpiColor(), dpiCurX, dpiCurY, dpiOn, dpiX, dpiY, HW_I_COUNT, hwIType, OFF, ON, pushedDpis, and settingsUpdated().

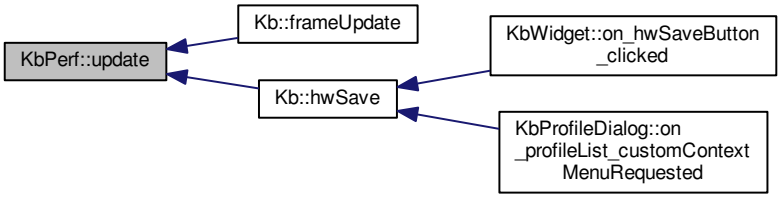
Referenced by Kb::frameUpdate(), and Kb::hwSave().

```
385                                                                                 {
386     if(!force && !_needsUpdate)
387         return;
388     emit settingsUpdated();
389     _needsUpdate = false;
390     // Save DPI stage 0 (sniper)
391     // If the mouse is set to a custom DPI, save it in stage 0
392     int stage = pushedDpis.isEmpty() ? dpiBaseIdx : 0;
393     if(!pushedDpis.isEmpty() && saveCustomDpi) {
394         cmd.write(QString("dpi 0:%1,%2").arg(dpiCurX).arg(dpiCurY).toLatin1());
395     } else {
396         // Otherwise, save stage 0 normally
397         cmd.write(QString("dpi 0:%1,%2").arg(dpiX[0]).arg(dpiY[0]).toLatin1());
398     }
399     // Save stages 1 - 5
400     for(int i = 1; i < DPI_COUNT; i++){
401         if(!dpiOn[i] && stage != i)
402             cmd.write(QString(" %1:off").arg(i).toLatin1());
403         else
404         cmd.write(QString(" %1:%2,%3").arg(i).arg(dpiX[i]).arg(dpiY[i]).toLatin1());
405     }
406     // Save stage selection, lift height, and angle snap
407     cmd.write(QString(" dpisel %1 lift %2 snap %3").arg(stage).arg(_liftHeight).arg(
    _angleSnap ? "on" : "off").toLatin1());
408     // Save DPI colors
409     cmd.write(" rgb");
410     for(int i = 0; i < DPI_COUNT; i++){
```

```
411         QColor color = dpiColor(i);
412         cmd.write(" dpi");
413         char output[9];
414         snprintf(output, sizeof(output), "%1d:%02x%02x%02x", i, color.red(), color.green(), color.blue());
415         cmd.write(output);
416     }
417     // Enable indicator notifications
418     cmd.write(QString("\n@%1 inotify all").arg(notifyNumber).toLatin1());
419     // Set indicator state
420     const char* iNames[HW_I_COUNT] = { "num", "caps", "scroll" };
421     for(int i = 0; i < HW_I_COUNT; i++){
422         if(hwIType[i] == ON)
423             cmd.write(" ion ");
424         else if(hwIType[i] == OFF)
425             cmd.write(" ioff ");
426         else
427             cmd.write(" iauto ");
428         cmd.write(iNames[i]);
429     }
430 }
```

Here is the call graph for this function:

Here is the caller graph for this function:



12.34.5 Field Documentation

12.34.5.1 bool KbPerf::_angleSnap [private]

Definition at line 164 of file kbperf.h.

Referenced by angleSnap(), load(), operator=(), save(), and update().

12.34.5.2 bool KbPerf::_dpiIndicator [private]

Definition at line 160 of file kbperf.h.

Referenced by applyIndicators(), dpiIndicator(), load(), operator=(), and save().

12.34.5.3 float KbPerf::_iOpacity [private]

Definition at line 155 of file kbperf.h.

Referenced by applyIndicators(), iOpacity(), lightIndicator(), load(), and operator=().

12.34.5.4 height KbPerf::_liftHeight [private]

Definition at line 163 of file kbperf.h.

Referenced by liftHeight(), load(), operator=(), save(), and update().

12.34.5.5 bool KbPerf::_needsSave [private]

Definition at line 166 of file kbperf.h.

Referenced by _curDpi(), angleSnap(), baseDpiIdx(), dpi(), dpiColor(), dpiEnabled(), dpiIndicator(), iOpacity(), liftHeight(), load(), needsSave(), operator=(), popDpi(), save(), and setIndicator().

12.34.5.6 bool KbPerf::_needsUpdate [private]

Definition at line 166 of file kbperf.h.

Referenced by _curDpi(), angleSnap(), baseDpiIdx(), dpi(), dpiColor(), dpiEnabled(), liftHeight(), operator=(), popDpi(), setIndicator(), setNeedsUpdate(), and update().

12.34.5.7 const int KbPerf::DPI_COUNT = 6 [static]

Definition at line 46 of file kbperf.h.

Referenced by dpi(), dpiCycleDown(), dpiCycleUp(), dpiUp(), KbPerf(), KeyAction::keyEvent(), load(), operator=(), Kb::readNotify(), save(), RebindWidget::setBind(), and update().

12.34.5.8 const int KbPerf::DPI_MAX = 12000 [static]

Definition at line 47 of file kbperf.h.

12.34.5.9 const int KbPerf::DPI_MIN = 100 [static]

Definition at line 47 of file kbperf.h.

Referenced by dpiExp(), and dpiLog().

12.34.5.10 int KbPerf::dpiBaseIdx [private]

Definition at line 142 of file kbperf.h.

Referenced by baseDpiIdx(), dpi(), KbPerf(), load(), operator=(), popDpi(), save(), and update().

12.34.5.11 QColor KbPerf::dpiClr[DPI_COUNT+1] [private]

Definition at line 143 of file kbperf.h.

Referenced by applyIndicators(), dpiColor(), KbPerf(), load(), operator=(), and save().

12.34.5.12 int KbPerf::dpiCurX [private]

Definition at line 138 of file kbperf.h.

Referenced by _curDpi(), KbPerf(), operator=(), and update().

12.34.5.13 int KbPerf::dpiCurY [private]

Definition at line 138 of file kbperf.h.

Referenced by _curDpi(), KbPerf(), operator=(), and update().

12.34.5.14 bool KbPerf::dpiOn[DPI_COUNT] [private]

Definition at line 144 of file kbperf.h.

Referenced by dpiCycleDown(), dpiCycleUp(), dpiDown(), dpiEnabled(), dpiUp(), KbPerf(), load(), operator=(), save(), and update().

12.34.5.15 int KbPerf::dpiX[DPI_COUNT] [private]

Definition at line 135 of file kbperf.h.

Referenced by dpi(), KbPerf(), load(), operator=(), save(), and update().

12.34.5.16 int KbPerf::dpiY[DPI_COUNT] [private]

Definition at line 136 of file kbperf.h.

Referenced by dpi(), KbPerf(), load(), operator=(), save(), and update().

12.34.5.17 const int KbPerf::HW_I_COUNT = (int)HW_IMAX + 1 [static]

Definition at line 107 of file kbperf.h.

Referenced by KbPerf(), and update().

12.34.5.18 i_hw KbPerf::hwlType[HW_I_COUNT] [private]

Definition at line 159 of file kbperf.h.

Referenced by getIndicator(), KbPerf(), load(), operator=(), save(), setIndicator(), and update().

12.34.5.19 const int KbPerf::I_COUNT = (int)MUTE + 1 [static]

Definition at line 107 of file kbperf.h.

Referenced by getIndicator(), KbPerf(), load(), operator=(), save(), and setIndicator().

12.34.5.20 QColor KbPerf::iColor[I_COUNT][2] [private]

Definition at line 156 of file kbperf.h.

Referenced by applyIndicators(), getIndicator(), KbPerf(), load(), operator=(), save(), and setIndicator().

12.34.5.21 bool KbPerf::iEnable[I_COUNT] [private]

Definition at line 158 of file kbperf.h.

Referenced by applyIndicators(), getIndicator(), KbPerf(), load(), operator=(), save(), and setIndicator().

12.34.5.22 QColor KbPerf::light100Color [private]

Definition at line 157 of file kbperf.h.

Referenced by applyIndicators(), getIndicator(), KbPerf(), load(), operator=(), save(), and setIndicator().

12.34.5.23 QColor KbPerf::muteNAColor [private]

Definition at line 157 of file kbperf.h.

Referenced by applyIndicators(), getIndicator(), KbPerf(), load(), operator=(), save(), and setIndicator().

12.34.5.24 const int KbPerf::OTHER = DPI_COUNT [static]

Definition at line 84 of file kbperf.h.

Referenced by applyIndicators(), MPerfWidget::colorChanged(), load(), save(), and MPerfWidget::setPerf().

12.34.5.25 QMap<quint64, QPoint> KbPerf::pushedDpis [private]

Definition at line 148 of file kbperf.h.

Referenced by baseDpiIdx(), dpi(), load(), popDpi(), pushDpi(), and update().

12.34.5.26 uint KbPerf::runningPushIdx [private]

Definition at line 149 of file kbperf.h.

Referenced by load(), operator=(), and pushDpi().

12.34.5.27 const int KbPerf::SNIPER = 0 [static]

Definition at line 46 of file kbperf.h.

Referenced by dpiCycleDown(), dpiCycleUp(), dpiDown(), and sniperDpi().

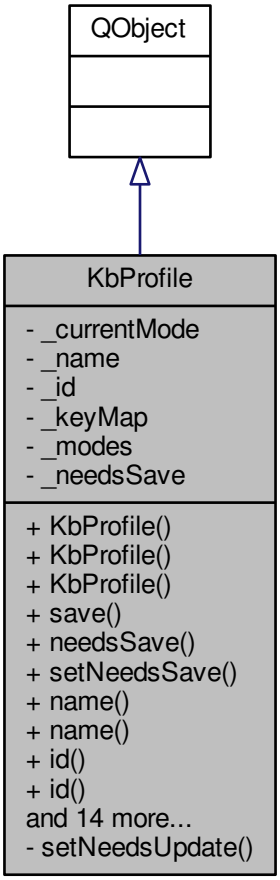
The documentation for this class was generated from the following files:

- src/ckb/kbperf.h
- src/ckb/kbperf.cpp
- src/ckb/release/moc_kbperf.cpp

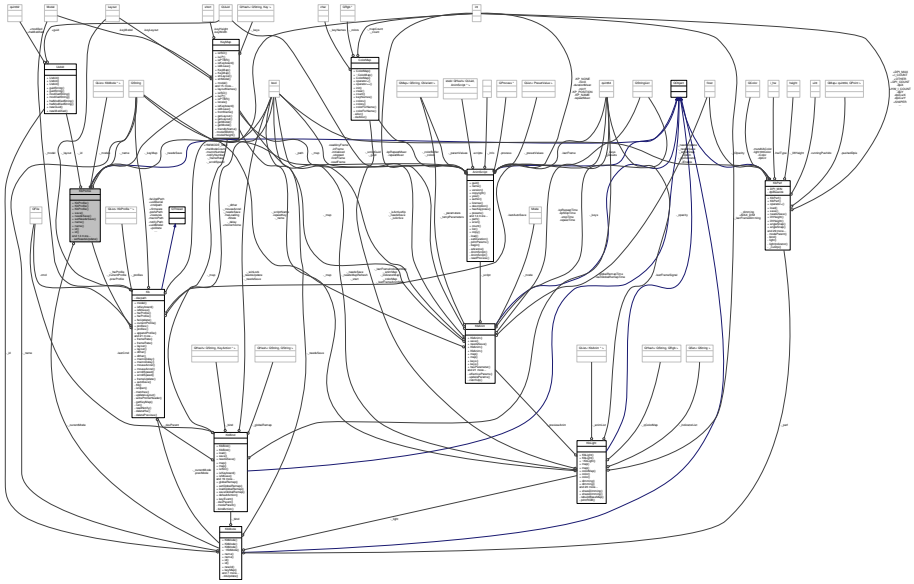
12.35 KbProfile Class Reference

`#include <src/ckb/kbprofile.h>`

Inheritance diagram for KbProfile:



Collaboration diagram for KbProfile:



Public Types

- typedef QList< KbMode * > ModeList

Public Member Functions

- KbProfile (Kb *parent, const KeyMap &keyMap, const KbProfile &other)
- KbProfile (Kb *parent, const KeyMap &keyMap, const QString &guid="", const QString &modified="")
- KbProfile (Kb *parent, const KeyMap &keyMap, CkbSettings &settings, const QString &guid)
- void save (CkbSettings &settings)
- bool needsSave () const
- void setNeedsSave ()
- QString name () const
- void name (const QString &newName)
- UsbId & id ()
- void id (const UsbId &newId)
- void newId ()
- const KeyMap & keyMap () const
- void keyMap (const KeyMap &newKeyMap)
- const ModeList & modes () const
- void modes (const QList< KbMode * > &newModes)
- void append (KbMode *newMode)
- void insert (int index, KbMode *newMode)
- void removeAll (KbMode *mode)
- void move (int from, int to)
- int modeCount () const
- int indexOf (KbMode *mode) const
- KbMode * find (const QUuid &id)
- KbMode * currentMode () const
- void currentMode (KbMode *newCurrentMode)

Private Member Functions

- void setNeedsUpdate ()

Private Attributes

- KbMode * _currentMode
- QString _name
- UsbId _id
- KeyMap _keyMap
- ModeList _modes
- bool _needsSave

12.35.1 Detailed Description

Definition at line 11 of file kbprofile.h.

12.35.2 Member Typedef Documentation

12.35.2.1 typedef QList<KbMode*> KbProfile::ModeList

Definition at line 41 of file kbprofile.h.

12.35.3 Constructor & Destructor Documentation

12.35.3.1 KbProfile::KbProfile (Kb * *parent*, const KeyMap & *keyMap*, const KbProfile & *other*) [explicit]

Definition at line 4 of file kbprofile.cpp.

References _currentMode, _modes, and modes().

```
4                                                                                   :
5     QObject(parent), _currentMode(0), _name(other._name),
      _id(other._id), _keyMap(keyMap), _needsSave(true)
6 {
7     foreach(KbMode* mode, other.modes()){
8         KbMode* newMode = new KbMode(parent, keyMap, *mode);
9         if(!_currentMode || mode == other._currentMode)
10            _currentMode = newMode;
11        _modes.append(newMode);
12    }
13 }
```

Here is the call graph for this function:

12.35.3.2 KbProfile::KbProfile (Kb * *parent*, const KeyMap & *keyMap*, const QString & *guid* = "", const QString & *modified* = "") [explicit]

Definition at line 15 of file kbprofile.cpp.

References _id, and UsbId::guid.

```
15                                                                                                  :
16     QObject(parent), _currentMode(0), _id(guid, modified.toUInt(0, 16)),
       _keyMap(keyMap), _needsSave(true)
17 {
18     if(_id.guid.isNull())
19         _id.guid = QUuid::createUuid();
20 }
```

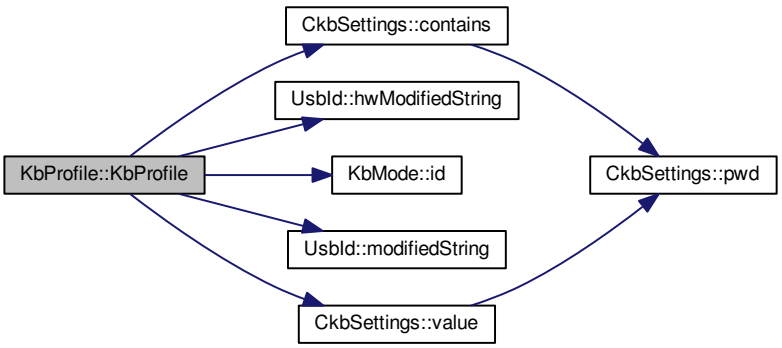
12.35.3.3 KbProfile::KbProfile (Kb * *parent*, const KeyMap & *keyMap*, CkbSettings & *settings*, const QString & *guid*) [explicit]

Definition at line 22 of file kbprofile.cpp.

References _currentMode, _id, _keyMap, _modes, _name, CkbSettings::contains(), UsbId::guid, UsbId::hwModified, UsbId::hwModifiedString(), KbMode::id(), UsbId::modified, UsbId::modifiedString(), and CkbSettings::value().

```
22                                                                                                  :
23     QObject(parent), _currentMode(0), _id(guid, 0),
       _keyMap(keyMap), _needsSave(false)
24 {
25     // Load data from preferences
26     SGroup group(settings, guid);
27     _name = settings.value("Name").toString().trimmed();
28     if(_name == "")
29         _name = "Unnamed";
30     _id.modifiedString(settings.value("Modified").toString());
31     if(settings.contains("HwModified"))
32         _id.hwModifiedString(settings.value("HwModified").toString());
33     else
34         _id.hwModified = _id.modified;
35     QUuid current = settings.value("CurrentMode").toString().trimmed();
36     // Load modes
37     uint count = settings.value("ModeCount").toUInt();
38     for(uint i = 0; i < count; i++){
39         SGroup group(settings, QString::number(i));
40         KbMode* mode = new KbMode(parent, _keyMap, settings);
41         _modes.append(mode);
42         // Set currentMode to the mode matching the current GUID, or the first mode in case it's not found
43         if(current == mode->id().guid || !_currentMode)
44             _currentMode = mode;
45     }
46 }
```

Here is the call graph for this function:



12.35.4 Member Function Documentation

12.35.4.1 void KbProfile::append (KbMode * *newMode*) [inline]

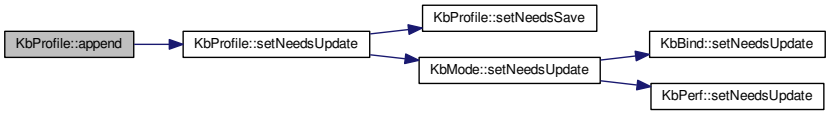
Definition at line 44 of file kbprofile.h.

References _modes, and setNeedsUpdate().

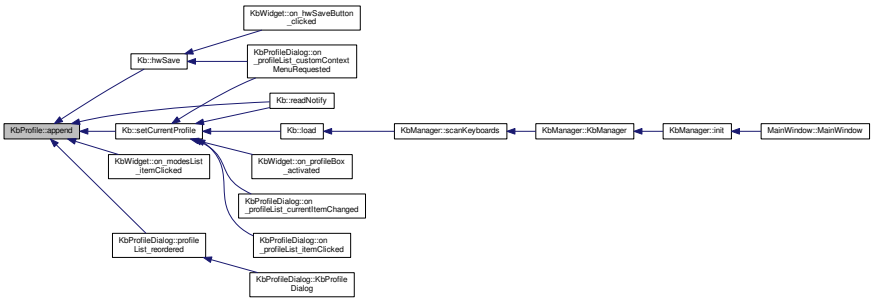
Referenced by Kb::hwSave(), KbWidget::on_modesList_itemClicked(), KbProfileDialog::profileList_reordered(), Kb::readNotify(), and Kb::setCurrentProfile().

```
44 { setNeedsUpdate(); _modes.append(newMode); }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.35.4.2 KbMode* KbProfile::currentMode ( ) const [inline]

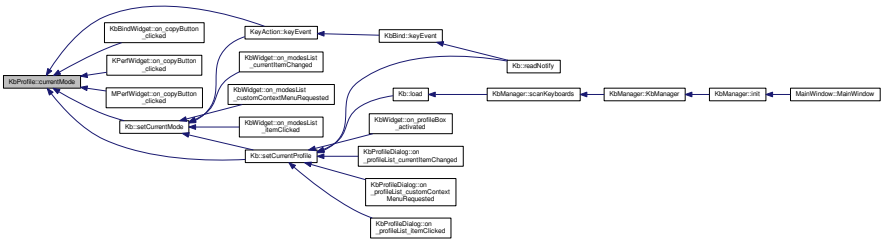
Definition at line 54 of file kbprofile.h.

References _currentMode.

Referenced by KeyAction::keyEvent(), KbBindWidget::on_copyButton_clicked(), KPerfWidget::on_copyButton_clicked(), MPerfWidget::on_copyButton_clicked(), Kb::setCurrentMode(), and Kb::setCurrentProfile().

```
54 { return _currentMode; }
```

Here is the caller graph for this function:



12.35.4.3 void KbProfile::currentMode (KbMode * *newCurrentMode*) [inline]

Definition at line 55 of file kbprofile.h.

References _currentMode, and _needsSave.

```
55 { _needsSave = true; _currentMode = newCurrentMode; }
```

12.35.4.4 KbMode* KbProfile::find (const QUuid & *id*) [inline]

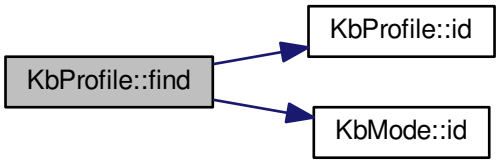
Definition at line 51 of file kbprofile.h.

References _modes, UsbId::guid, id(), and KbMode::id().

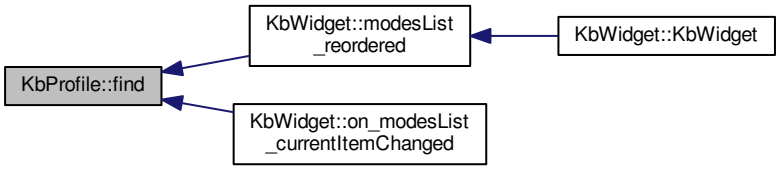
Referenced by KbWidget::modesList_reordered(), and KbWidget::on_modesList_currentItemChanged().

```
51 { foreach(KbMode* mode, _modes) { if(mode->id().guid == id) return mode; } return 0; }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.35.4.5 UsbId& KbProfile::id ( ) [inline]

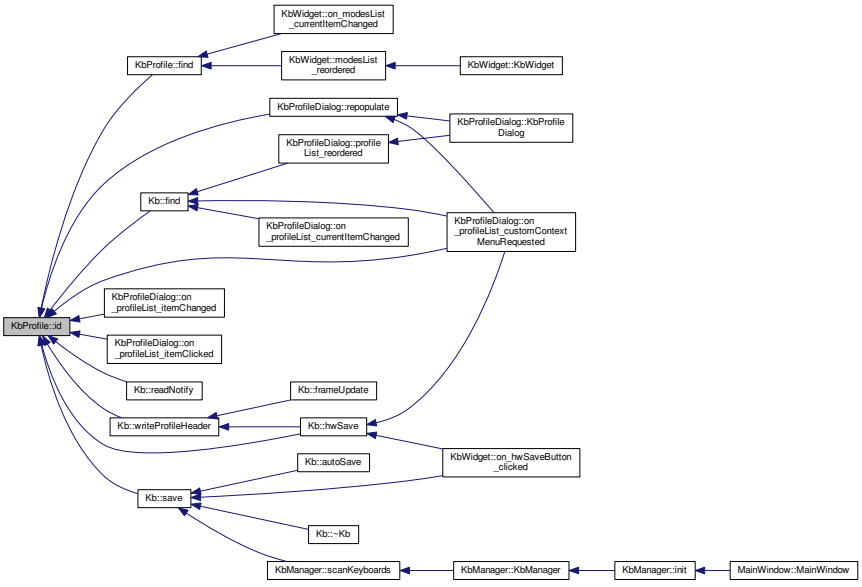
Definition at line 30 of file kbprofile.h.

References _id.

Referenced by find(), Kb::find(), Kb::hwSave(), KbProfileDialog::on_profileList_customContextMenuRequested(), KbProfileDialog::on_profileList_itemChanged(), KbProfileDialog::on_profileList_itemClicked(), Kb::readNotify(), KbProfileDialog::repopulate(), Kb::save(), and Kb::writeProfileHeader().

```
30 { return _id; }
```

Here is the caller graph for this function:



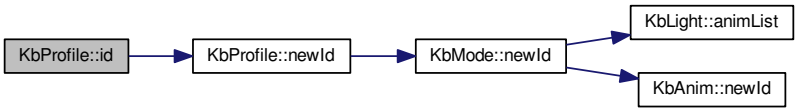
12.35.4.6 void KbProfile::id (const UsbId & *newId*) [inline]

Definition at line 31 of file kbprofile.h.

References _id, _needsSave, and newId().

```
31 { _needsSave = true; _id = newId; }
```

Here is the call graph for this function:



12.35.4.7 int KbProfile::indexOf (KbMode * *mode*) const [inline]

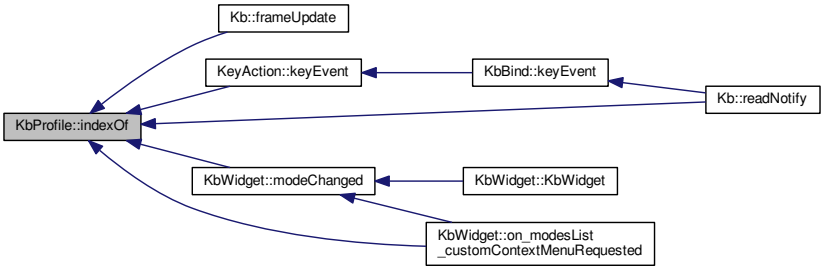
Definition at line 50 of file kbprofile.h.

References _modes.

Referenced by Kb::frameUpdate(), KeyAction::keyEvent(), KbWidget::modeChanged(), KbWidget::on_modesList_customContextMenuRequested(), and Kb::readNotify().

```
50 { return _modes.indexOf(mode); }
```

Here is the caller graph for this function:



12.35.4.8 void KbProfile::insert (int *index*, KbMode * *newMode*) [inline]

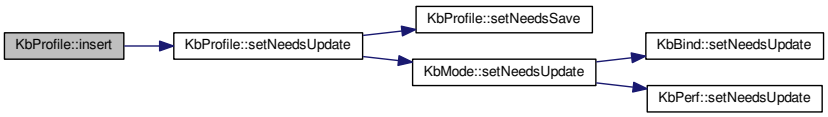
Definition at line 45 of file kbprofile.h.

References _modes, and setNeedsUpdate().

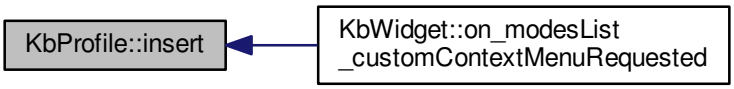
Referenced by KbWidget::on_modesList_customContextMenuRequested().

```
45 { setNeedsUpdate(); _modes.insert(index, newMode); }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.35.4.9 const KeyMap& KbProfile::keyMap ( ) const [inline]

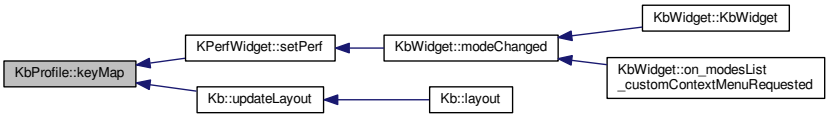
Definition at line 37 of file kbprofile.h.

References _keyMap.

Referenced by KPerfWidget::setPerf(), and Kb::updateLayout().

```
37 { return _keyMap; }
```

Here is the caller graph for this function:



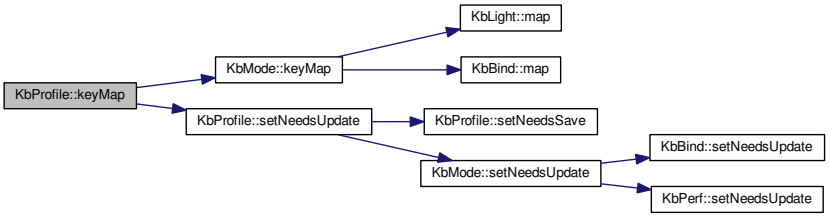
12.35.4.10 void KbProfile::keyMap (const KeyMap & *newKeyMap*)

Definition at line 85 of file kbprofile.cpp.

References _keyMap, _modes, KbMode::keyMap(), and setNeedsUpdate().

```
85                                              {
86     _keyMap = newKeyMap;
87     foreach(KbMode* mode, _modes)
88         mode->keyMap(newKeyMap);
89     setNeedsUpdate();
90 }
```

Here is the call graph for this function:



12.35.4.11 int KbProfile::modeCount ( ) const [inline]

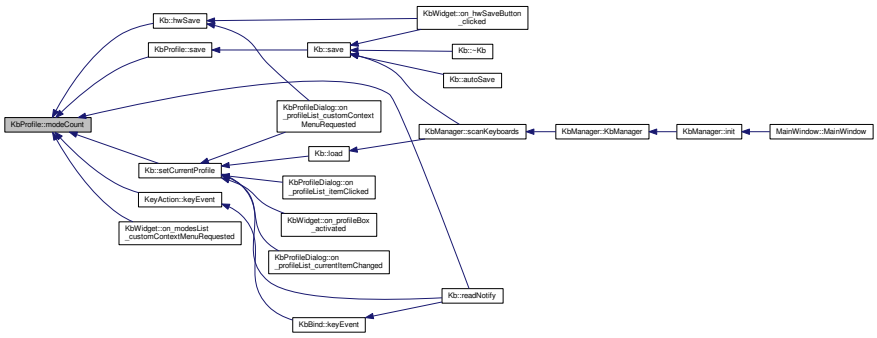
Definition at line 49 of file kbprofile.h.

References _modes.

Referenced by Kb::hwSave(), KeyAction::keyEvent(), KbWidget::on_modesList_customContextMenuRequested(), Kb::readNotify(), save(), and Kb::setCurrentProfile().

```
49 { return _modes.count(); }
```

Here is the caller graph for this function:



12.35.4.12 const ModeList& KbProfile::modes ( ) const [inline]

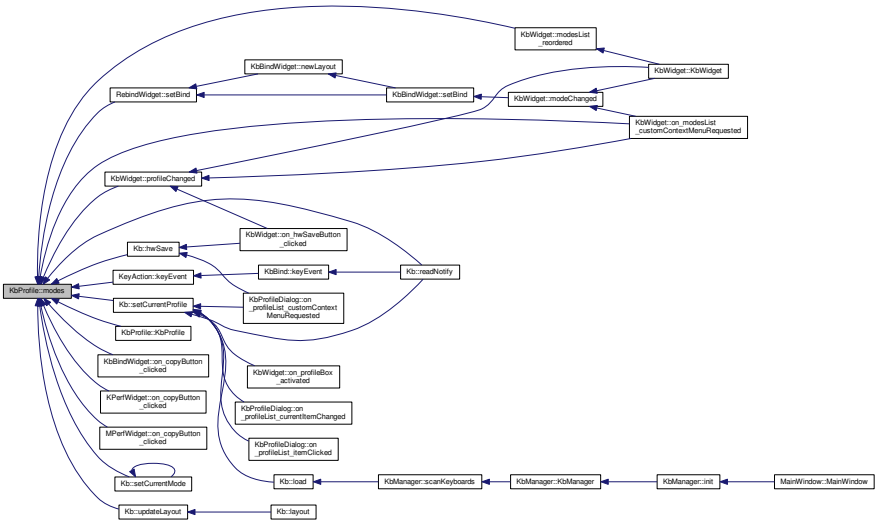
Definition at line 42 of file kbprofile.h.

References _modes.

Referenced by Kb::hwSave(), KbProfile(), KeyAction::keyEvent(), KbWidget::modesList_reordered(), KbBindWidget::on_copyButton_clicked(), KPerfWidget::on_copyButton_clicked(), MPerfWidget::on_copyButton_clicked(), KbWidget::on_modesList_customContextMenuRequested(), KbWidget::profileChanged(), Kb::readNotify(), RebindWidget::setBind(), Kb::setCurrentMode(), Kb::setCurrentProfile(), and Kb::updateLayout().

```
42 { return _modes; }
```

Here is the caller graph for this function:



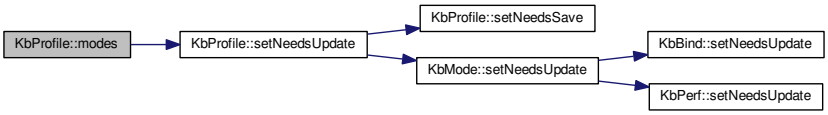
12.35.4.13 void KbProfile::modes (const QList< KbMode * > & *newModes*) [inline]

Definition at line 43 of file kbprofile.h.

References _modes, and setNeedsUpdate().

```
43 { setNeedsUpdate(); _modes = newModes; }
```

Here is the call graph for this function:



12.35.4.14 void KbProfile::move (int *from*, int *to*) [inline]

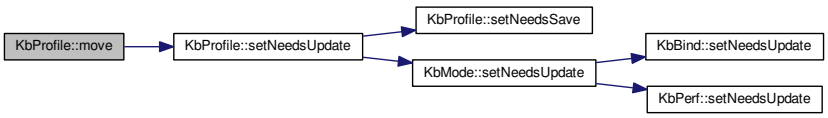
Definition at line 47 of file kbprofile.h.

References _modes, and setNeedsUpdate().

Referenced by Kb::readNotify().

```
47 { setNeedsUpdate(); _modes.move(from, to); }
```

Here is the call graph for this function:



Here is the caller graph for this function:

12.35.4.15 QString KbProfile::name ( ) const [inline]

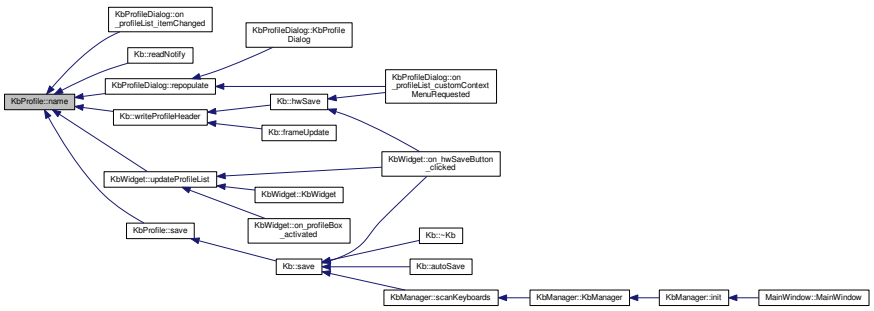
Definition at line 28 of file kbprofile.h.

References _name.

Referenced by KbProfileDialog::on_profileList_itemChanged(), Kb::readNotify(), KbProfileDialog::repopulate(), save(), KbWidget::updateProfileList(), and Kb::writeProfileHeader().

```
28 { return _name; }
```

Here is the caller graph for this function:



12.35.4.16 void KbProfile::name (const QString & *newName*) [inline]

Definition at line 29 of file kbprofile.h.

References _name, and _needsSave.

```
29 { _needsSave = true; _name = newName.trimmed(); if(_name == "")
      _name = "Unnamed"; }
```

12.35.4.17 bool KbProfile::needsSave ( ) const

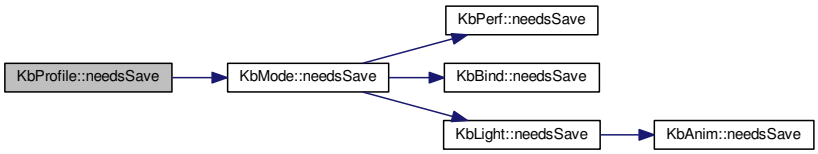
Definition at line 68 of file kbprofile.cpp.

References _modes, _needsSave, and KbMode::needsSave().

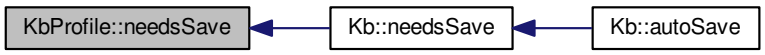
Referenced by Kb::needsSave().

```
68                                  {
69     if(_needsSave)
70         return true;
71     foreach(KbMode* mode, _modes){
72         if(mode->needsSave())
73             return true;
74     }
75     return false;
76 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.35.4.18 void KbProfile::newId ( )

Definition at line 78 of file kbprofile.cpp.

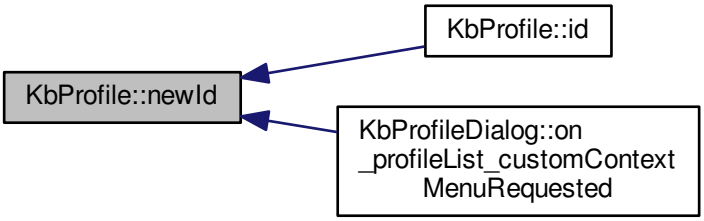
References _id, _modes, _needsSave, and KbMode::newId().

Referenced by id(), and KbProfileDialog::on_profileList_customContextMenuRequested().

```
78                         {
79     _needsSave = true;
80     _id = UsbId();
81     foreach(KbMode* mode, _modes)
82         mode->newId();
83 }
```

Here is the call graph for this function:

Here is the caller graph for this function:



12.35.4.19 void KbProfile::removeAll (KbMode * *mode*) [inline]

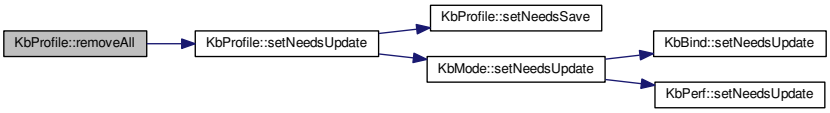
Definition at line 46 of file kbprofile.h.

References _modes, and setNeedsUpdate().

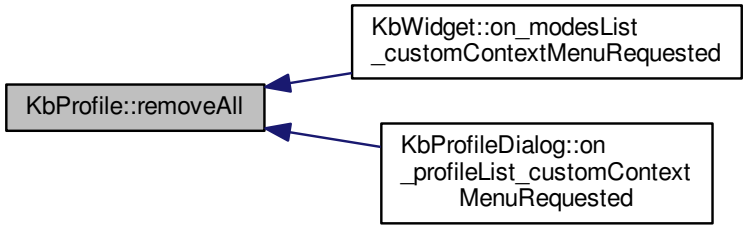
Referenced by KbWidget::on_modesList_customContextMenuRequested(), and KbProfileDialog::on_profileList_-customContextMenuRequested().

```
46 { setNeedsUpdate(); _modes.removeAll(mode); }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.35.4.20 void KbProfile::save (CkbSettings & *settings*)

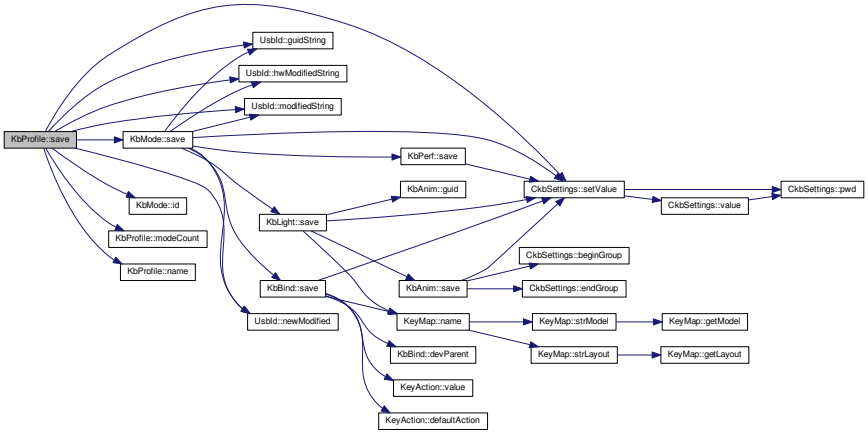
Definition at line 48 of file kbprofile.cpp.

References _currentMode, _id, _modes, _needsSave, UsbId::guidString(), UsbId::hwModifiedString(), KbMode::id(), modeCount(), UsbId::modifiedString(), name(), UsbId::newModified(), KbMode::save(), and CkbSettings::setValue().

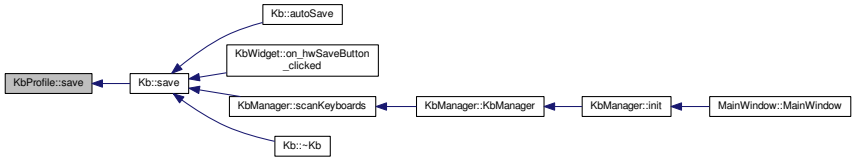
Referenced by Kb::save().

```
48                                          {
49     _needsSave = false;
50     _id.newModified();
51     // Save data to preferences
52     SGroup group(settings, id().guidString());
53     settings.setValue("Name", name());
54     settings.setValue("Modified", _id.modifiedString());
55     settings.setValue("HwModified", _id.hwModifiedString());
56     if(_currentMode)
57         settings.setValue("CurrentMode", _currentMode->id().
     guidString());
58     // Save modes
59     uint count = modeCount();
60     settings.setValue("ModeCount", count);
61     for(uint i = 0; i < count; i++){
62         SGroup group(settings, QString::number(i));
63         KbMode* mode = _modes.at(i);
64         mode->save(settings);
65     }
66 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.35.4.21 void KbProfile::setNeedsSave ( ) [inline]

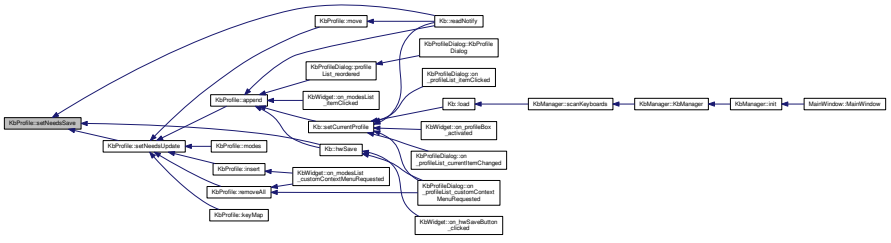
Definition at line 25 of file kbprofile.h.

References _needsSave.

Referenced by Kb::hwSave(), Kb::readNotify(), and setNeedsUpdate().

```
25 { _needsSave = true; }
```

Here is the caller graph for this function:



12.35.4.22 void KbProfile::setNeedsUpdate ( ) [inline], [private]

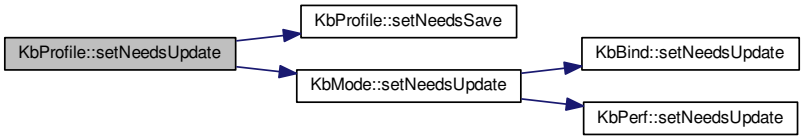
Definition at line 66 of file kbprofile.h.

References _modes, setNeedsSave(), and KbMode::setNeedsUpdate().

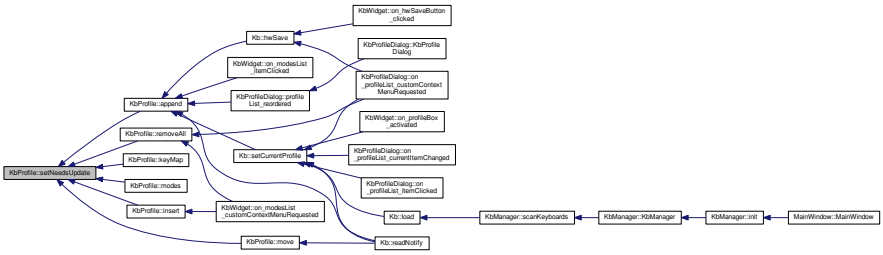
Referenced by append(), insert(), keyMap(), modes(), move(), and removeAll().

```
66 { setNeedsSave(); foreach(KbMode* mode, _modes){ mode->
      setNeedsUpdate(); } }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.35.5 Field Documentation

12.35.5.1 KbMode* KbProfile::_currentMode [private]

Definition at line 58 of file kbprofile.h.

Referenced by currentMode(), KbProfile(), and save().

12.35.5.2 UsbId KbProfile::_id [private]

Definition at line 60 of file kbprofile.h.

Referenced by id(), KbProfile(), newId(), and save().

12.35.5.3 KeyMap KbProfile::_keyMap [private]

Definition at line 61 of file kbprofile.h.

Referenced by KbProfile(), and keyMap().

12.35.5.4 ModeList KbProfile::_modes [private]

Definition at line 62 of file kbprofile.h.

Referenced by append(), find(), indexOf(), insert(), KbProfile(), keyMap(), modeCount(), modes(), move(), needsSave(), newId(), removeAll(), save(), and setNeedsUpdate().

12.35.5.5 QString KbProfile::_name [private]

Definition at line 59 of file kbprofile.h.

Referenced by KbProfile(), and name().

12.35.5.6 bool KbProfile::_needsSave [private]

Definition at line 63 of file kbprofile.h.

Referenced by currentMode(), id(), name(), needsSave(), newId(), save(), and setNeedsSave().

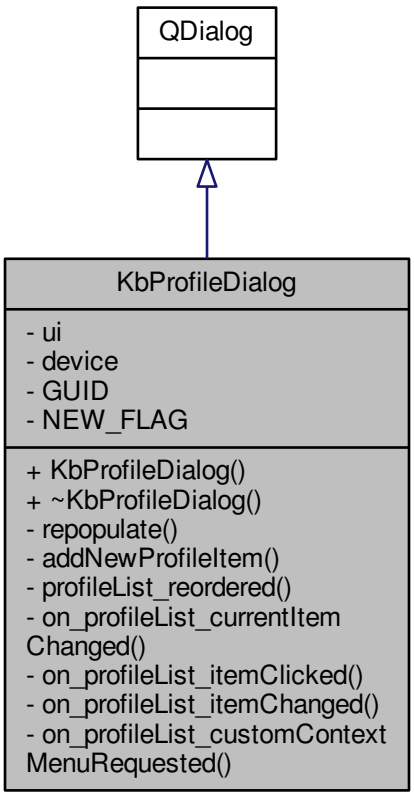
The documentation for this class was generated from the following files:

- src/ckb/kbprofile.h
- src/ckb/kbprofile.cpp

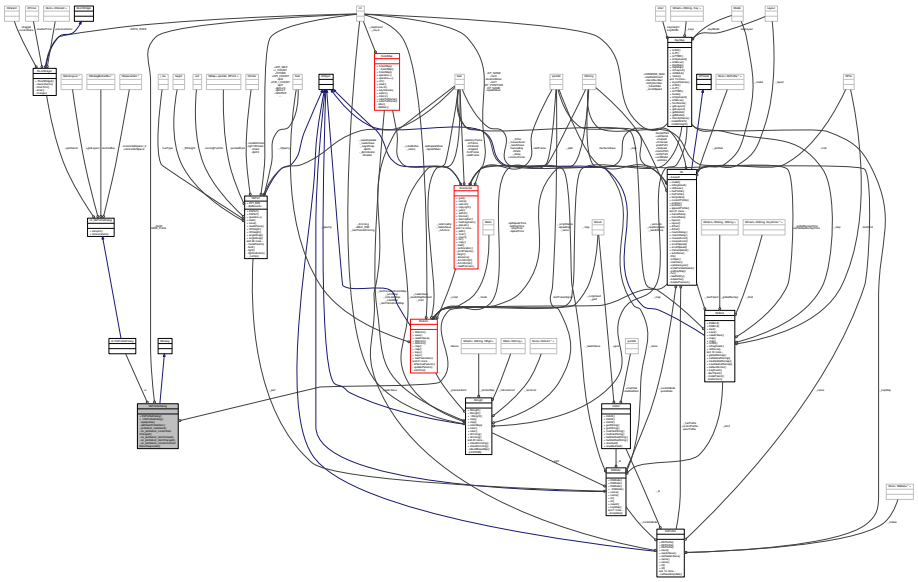
12.36 KbProfileDialog Class Reference

`#include <src/ckb/kbprofiledialog.h>`

Inheritance diagram for KbProfileDialog:



Collaboration diagram for KbProfileDialog:



Public Member Functions

- KbProfileDialog (KbWidget *parent=0)
- ~KbProfileDialog ()

Private Slots

- void profileList_reordered ()
- void on_profileList_currentItemChanged (QListWidgetItem *current, QListWidgetItem *previous)
- void on_profileList_itemClicked (QListWidgetItem *item)
- void on_profileList_itemChanged (QListWidgetItem *item)
- void on_profileList_customContextMenuRequested (const QPoint &pos)

Private Member Functions

- void repopulate ()
- void addNewProfileItem ()

Private Attributes

- Ui::KbProfileDialog * ui
- Kb * device

Static Private Attributes

- static const int GUID = Qt::UserRole
- static const int NEW_FLAG = Qt::UserRole + 1

12.36.1 Detailed Description

Definition at line 11 of file kbprofiledialog.h.

12.36.2 Constructor & Destructor Documentation

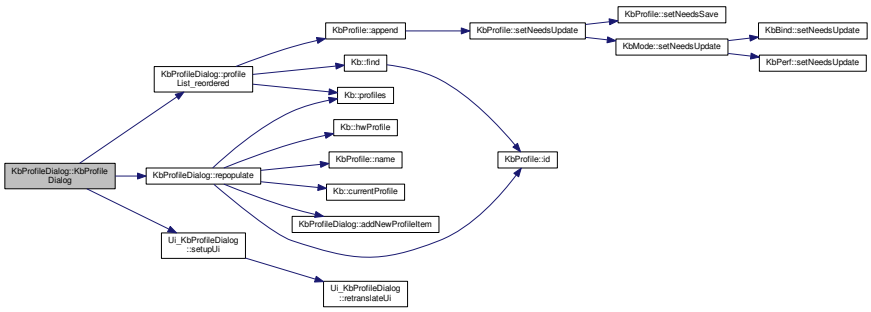
12.36.2.1 KbProfileDialog::KbProfileDialog (KbWidget * *parent* = 0) [explicit]

Definition at line 5 of file kbprofiledialog.cpp.

References Ui_KbProfileDialog::profileList, profileList_reordered(), repopulate(), Ui_KbProfileDialog::setupUi(), and ui.

```
5                                                  :
6     QDialog(parent),
7     ui(new Ui::KbProfileDialog), device(parent->
      device)
8 {
9     ui->setupUi(this);
10     connect(ui->profileList, SIGNAL(orderChanged()), this, SLOT(
      profileList_reordered()));
11
12     // Populate profile list
13     repopulate();
14 }
```

Here is the call graph for this function:



12.36.2.2 KbProfileDialog::~KbProfileDialog ( )

Definition at line 16 of file kbprofiledialog.cpp.

References ui.

```
16                                      {
17     delete ui;
18 }
```

12.36.3 Member Function Documentation

12.36.3.1 void KbProfileDialog::addNewProfileItem ( ) [private]

Definition at line 66 of file kbprofiledialog.cpp.

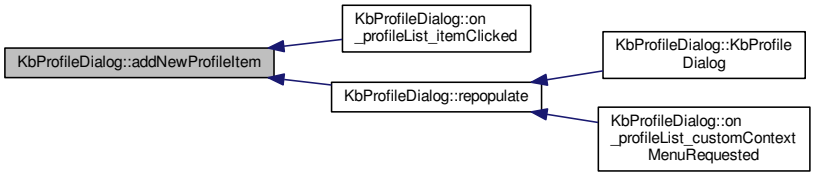
References NEW_FLAG, Ui_KbProfileDialog::profileList, and ui.

Referenced by on_profileList_itemClicked(), and repopulate().

```
66                                        {
67     // Add an item for creating a new profile. Make it editable but not dragable.
68     QListWidgetItem* item = new QListWidgetItem("New profile...", ui->
      profileList);
69     item->setFlags((item->flags() | Qt::ItemIsEditable) & ~Qt::ItemIsDragEnabled & ~Qt::ItemIsDropEnabled);
70     item->setData(NEW_FLAG, 1);
```

```
71     QFont font = item->font();
72     font.setItalic(true);
73     item->setFont(font);
74     item->setIcon(QIcon(":/img/icon_plus.png"));
75     ui->profileList->addItem(item);
76 }
```

Here is the caller graph for this function:



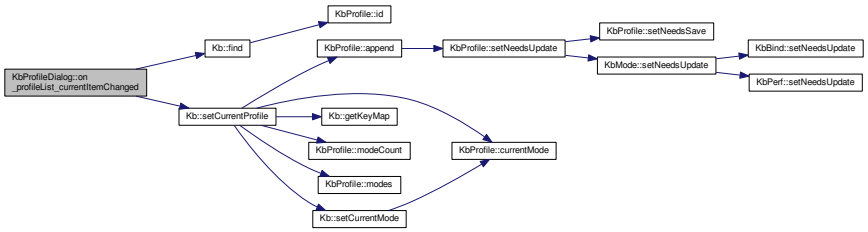
12.36.3.2 void KbProfileDialog::on_profileList_currentItemChanged (QListWidgetItem * *current*, QListWidgetItem * *previous*) [private], [slot]

Definition at line 39 of file kbprofiledialog.cpp.

References device, Kb::find(), GUID, and Kb::setCurrentProfile().

```
39                                                                                                  {
40     if(!current)
41         return;
42     KbProfile* profile = device->find(current->data(GUID).toUuid());
43     if(!profile)
44         return;
45     device->setCurrentProfile(profile);
46 }
```

Here is the call graph for this function:



12.36.3.3 void KbProfileDialog::on_profileList_customContextMenuRequested (const QPoint & *pos*) [private], [slot]

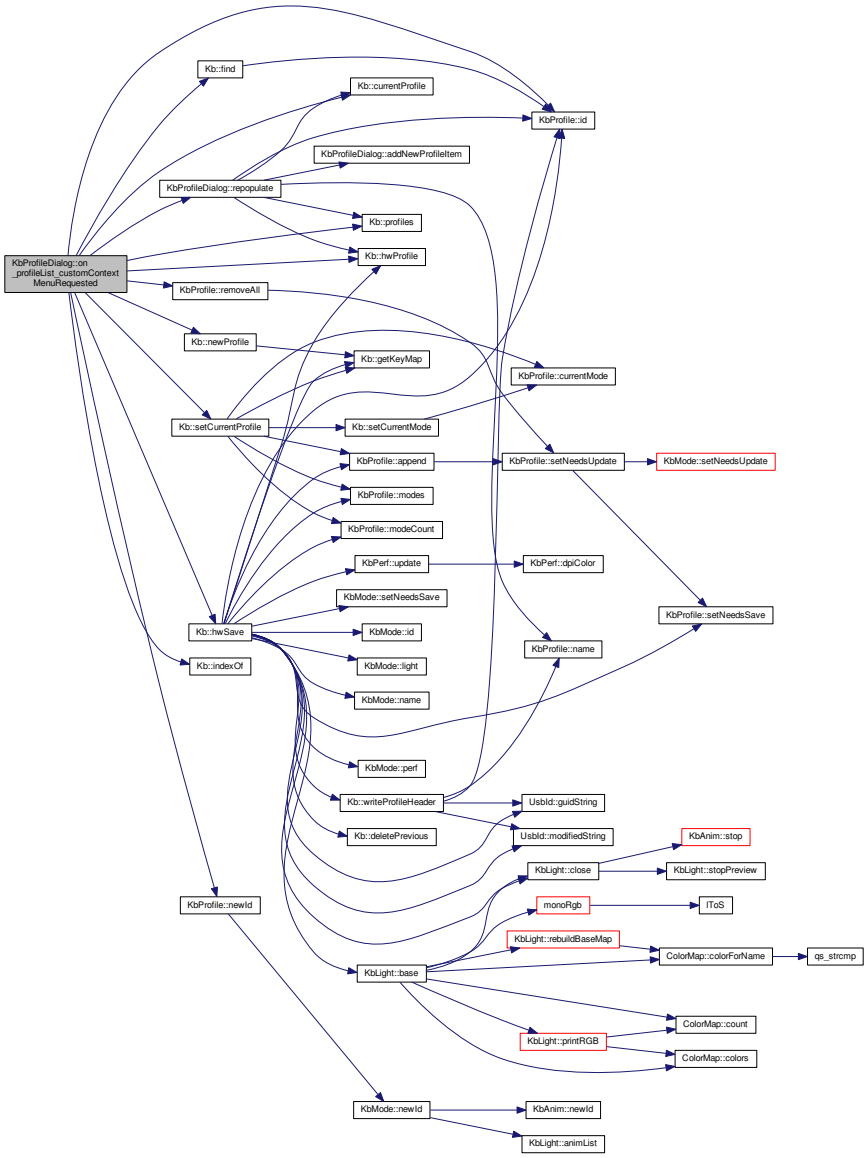
Definition at line 109 of file kbprofiledialog.cpp.

References Kb::currentProfile(), device, Kb::find(), UsbId::guid, GUID, Kb::hwProfile(), Kb::hwSave(), KbProfile::id(), Kb::indexOf(), NEW_FLAG, KbProfile::newId(), Kb::newProfile(), Ui_KbProfileDialog::profileList, Kb::profiles(), KbProfile::removeAll(), repopulate(), Kb::setCurrentProfile(), and ui.

```
109                                                                                                 {
```

```
110     QListWidgetItem* item = ui->profileList->itemAt(pos);
111     KbProfile* currentProfile = device->currentProfile();
112     if(!item || !currentProfile || item->data(GUID).toUuid() != currentProfile->
    id().guid)
113         return;
114     int index = device->indexOf(currentProfile);
115     QList<KbProfile*> profiles = device->profiles();
116
117     QMenu menu(this);
118     QAction* rename = new QAction("Rename...", this);
119     QAction* duplicate = new QAction("Duplicate", this);
120     QAction* del = new QAction("Delete", this);
121     bool canDelete = (profiles.count() > 1);
122     if(!canDelete)
123         // Can't delete the last profile on the device
124         del->setEnabled(false);
125     QAction* hwsave = new QAction("Save to Hardware", this);
126     QAction* moveup = new QAction("Move Up", this);
127     if(index == 0)
128         moveup->setEnabled(false);
129     QAction* movedown = new QAction("Move Down", this);
130     if(index >= profiles.count() - 1)
131         movedown->setEnabled(false);
132     menu.addAction(rename);
133     menu.addAction(duplicate);
134     menu.addAction(del);
135     menu.addSeparator();
136     menu.addAction(hwsave);
137     menu.addSeparator();
138     menu.addAction(moveup);
139     menu.addAction(movedown);
140     QAction* result = menu.exec(QCursor::pos());
141     if(result == rename){
142         ui->profileList->editItem(item);
143     } else if(result == duplicate){
144         KbProfile* newProfile = device->newProfile(currentProfile);
145         newProfile->newId();
146         profiles.insert(index + 1, newProfile);
147         device->profiles(profiles);
148         device->setCurrentProfile(newProfile);
149     } else if(result == del){
150         if(!canDelete)
151             return;
152         profiles.removeAll(currentProfile);
153         device->profiles(profiles);
154         currentProfile->deleteLater();
155         // Select next profile
156         if(index < profiles.count())
157             device->setCurrentProfile(profiles.at(index));
158         else
159             device->setCurrentProfile(profiles.last());
160     } else if(result == hwsave){
161         device->hwSave();
162         // Refresh item icons
163         int count = ui->profileList->count();
164         for(int i = 0; i < count; i++){
165             QListWidgetItem* item = ui->profileList->item(i);
166             if(item->data(NEW_FLAG).toUInt() == 1)
167                 continue;
168             KbProfile* profile = device->find(item->data(GUID).toUuid());
169             item->setIcon(QIcon((profile == device->hwProfile()) ? "
    :/img/icon_profile_hardware.png" : ":/img/icon_profile.png"));
170         }
171     } else if(result == moveup){
172         profiles.removeAll(currentProfile);
173         profiles.insert(index - 1, currentProfile);
174         device->profiles(profiles);
175     } else if(result == movedown){
176         profiles.removeAll(currentProfile);
177         profiles.insert(index + 1, currentProfile);
178         device->profiles(profiles);
179     }
180     repopulate();
181 }
```

Here is the call graph for this function:



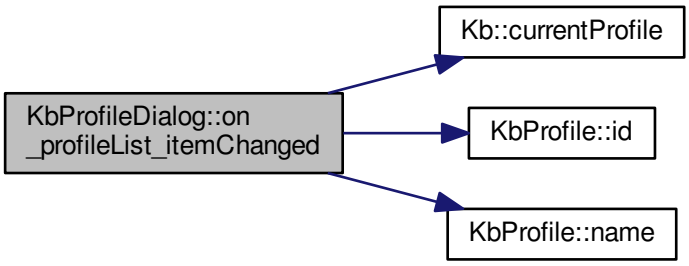
12.36.3.4 void KbProfileDialog::on_profileList_itemChanged (QListWidgetItem * *item*) [private], [slot]

Definition at line 100 of file kbprofiledialog.cpp.

References Kb::currentProfile(), device, UsbId::guid, GUID, KbProfile::id(), and KbProfile::name().

```
100                                                                 {
101     KbProfile* currentProfile = device->currentProfile();
102     if(!item || !currentProfile || item->data(GUID).toUuid() != currentProfile->
      id().guid)
103         return;
104     currentProfile->name(item->text());
105     // Set the text to the actual name (trimmed, "" replaced with "Unnamed")
106     item->setText(currentProfile->name());
107 }
```

Here is the call graph for this function:



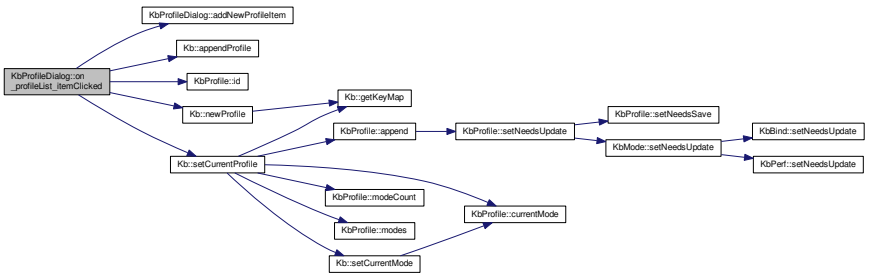
12.36.3.5 void KbProfileDialog::on_profileList_itemClicked (QListWidgetItem * *item*) [private], [slot]

Definition at line 78 of file kbprofiledialog.cpp.

References addNewProfileItem(), Kb::appendProfile(), device, UsbId::guid, GUID, KbProfile::id(), NEW_FLAG, Kb::newProfile(), Ui_KbProfileDialog::profileList, Kb::setCurrentProfile(), and ui.

```
78                                                                  {
79     QUuid guid = item->data(GUID).toUuid();
80     if(guid.isNull() && item->data(NEW_FLAG).toInt() == 1){
81         // New profile
82         item->setText("");
83         ui->profileList->editItem(item);
84         item->setFlags(item->flags() | Qt::ItemIsEditable | Qt::ItemIsDragEnabled | Qt::ItemIsDropEnabled);
85         QFont font = item->font();
86         font.setItalic(false);
87         item->setFont(font);
88         item->setIcon(QIcon(":/img/icon_profile.png"));
89         // Add the new profile and assign it to this item
90         KbProfile* newProfile = device->newProfile();
91         device->appendProfile(newProfile);
92         item->setData(GUID, newProfile->id().guid);
93         item->setData(NEW_FLAG, 0);
94         device->setCurrentProfile(newProfile);
95         // Create another "new profile" item to replace this one
96         addNewProfileItem();
97     }
98 }
```

Here is the call graph for this function:



12.36.3.6 void KbProfileDialog::profileList_reordered ( ) [private],[slot]

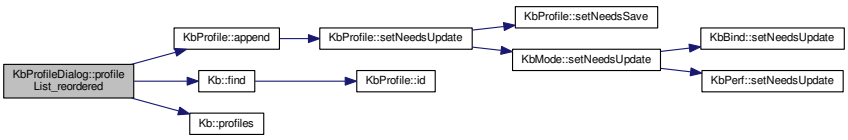
Definition at line 20 of file kbprofiledialog.cpp.

References KbProfile::append(), device, Kb::find(), GUID, Ui_KbProfileDialog::profileList, Kb::profiles(), and ui.

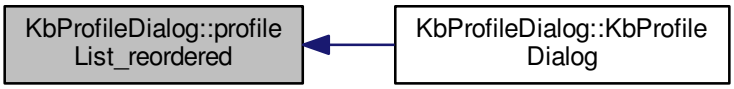
Referenced by KbProfileDialog().

```
20                                          {
21     // Rebuild profile list from items
22     QList<KbProfile*> newProfiles;
23     int count = ui->profileList->count();
24     for(int i = 0; i < count; i++){
25         QListWidgetItem* item = ui->profileList->item(i);
26         KbProfile* profile = device->find(item->data(GUID).toUuid());
27         if(profile && !newProfiles.contains(profile))
28             newProfiles.append(profile);
29         item->setFlags(item->flags() | Qt::ItemIsEditable);
30     }
31     // Add any missing profiles at the end of the list
32     foreach(KbProfile* profile, device->profiles()){
33         if(!newProfiles.contains(profile))
34             newProfiles.append(profile);
35     }
36     device->profiles(newProfiles);
37 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.36.3.7 void KbProfileDialog::repopulate ( ) [private]

Definition at line 48 of file kbprofiledialog.cpp.

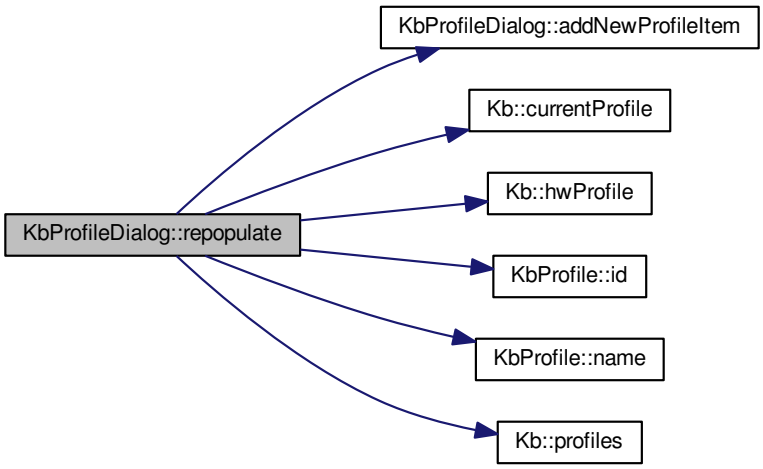
References addNewProfileItem(), Kb::currentProfile(), device, UsbId::guid, GUID, Kb::hwProfile(), KbProfile::id(), KbProfile::name(), Ui_KbProfileDialog::profileList, Kb::profiles(), and ui.

Referenced by KbProfileDialog(), and on_profileList_customContextMenuRequested().

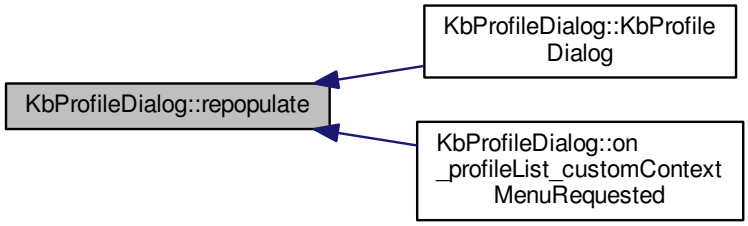
```
48                                          {
49     ui->profileList->clear();
50     QListWidgetItem* current = 0;
51     foreach(KbProfile* profile, device->profiles()){
52         QListWidgetItem* item = new QListWidgetItem(QIcon((profile == device->
      hwProfile()) ? ":/img/icon_profile_hardware.png" : ":/img/icon_profile.png"), profile->
```

```
      name(), ui->profileList);
53          item->setData(GUID, profile->id().guid);
54          item->setFlags(item->flags() | Qt::ItemIsEditable);
55          if(profile == device->currentProfile()){
56              item->setSelected(true);
57              current = item;
58          }
59          ui->profileList->addItem(item);
60      }
61      if(current)
62          ui->profileList->setCurrentItem(current);
63      addNewProfileItem();
64  }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.36.4 Field Documentation

12.36.4.1 Kb* KbProfileDialog::device [private]

Definition at line 29 of file kbprofiledialog.h.

Referenced by on_profileList_currentItemChanged(), on_profileList_customContextMenuRequested(), on_profileList_itemChanged(), on_profileList_itemClicked(), profileList_reordered(), and repopulate().

12.36.4.2 const int KbProfileDialog::GUID = Qt::UserRole [static], [private]

Definition at line 30 of file kbprofiledialog.h.

Referenced by on_profileList_currentItemChanged(), on_profileList_customContextMenuRequested(), on_profileList_itemChanged(), on_profileList_itemClicked(), profileList_reordered(), and repopulate().

12.36.4.3 const int KbProfileDialog::NEW_FLAG = Qt::UserRole + 1 [static], [private]

Definition at line 31 of file kbprofiledialog.h.

Referenced by addNewProfileItem(), on_profileList_customContextMenuRequested(), and on_profileList_itemClicked().

12.36.4.4 Ui::KbProfileDialog* KbProfileDialog::ui [private]

Definition at line 27 of file kbprofiledialog.h.

Referenced by addNewProfileItem(), KbProfileDialog(), on_profileList_customContextMenuRequested(), on_profileList_itemClicked(), profileList_reordered(), repopulate(), and ~KbProfileDialog().

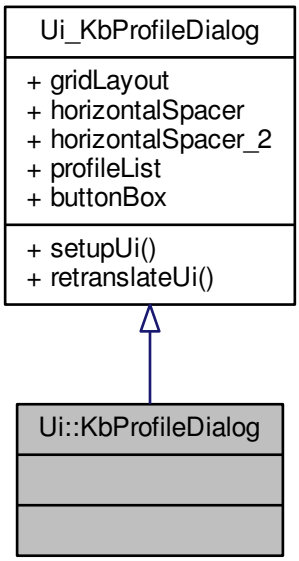
The documentation for this class was generated from the following files:

- src/ckb/kbprofiledialog.h
- src/ckb/kbprofiledialog.cpp

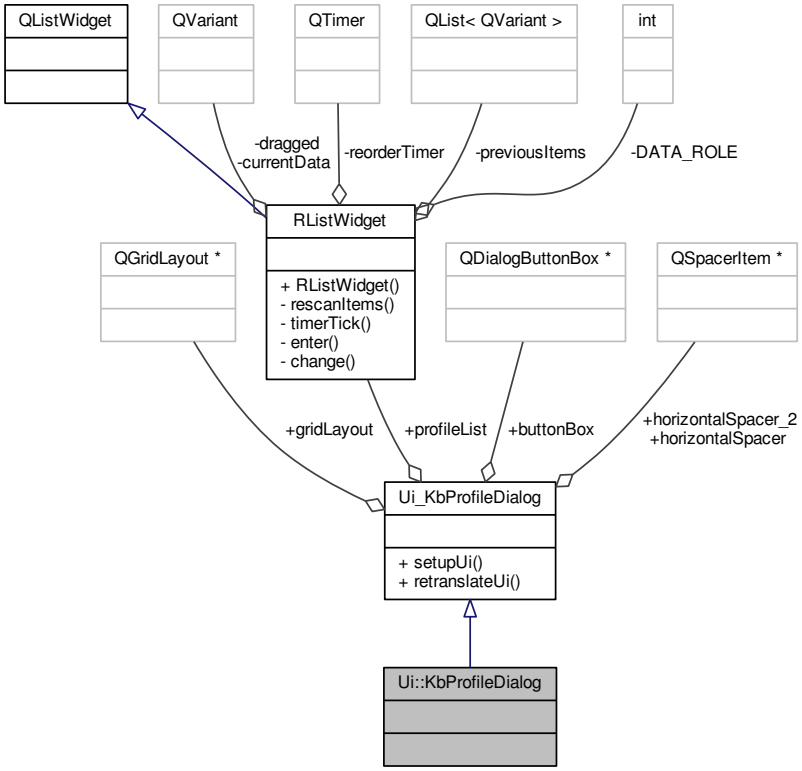
12.37 Ui::KbProfileDialog Class Reference

#include <src/ckb/ui_kbprofiledialog.h>

Inheritance diagram for Ui::KbProfileDialog:



Collaboration diagram for Ui::KbProfileDialog:



Public Member Functions

- void setupUi (QDialog *KbProfileDialog)
- void retranslateUi (QDialog *KbProfileDialog)

Data Fields

- QGridLayout * gridLayout
- QSpacerItem * horizontalSpacer
- QSpacerItem * horizontalSpacer_2
- RListWidget * profileList
- QDialogButtonBox * buttonBox

12.37.1 Detailed Description

Definition at line 81 of file ui_kbprofiledialog.h.

12.37.2 Member Function Documentation

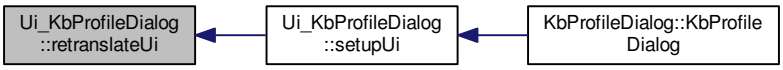
12.37.2.1 void Ui_KbProfileDialog::retranslateUi (QDialog * *KbProfileDialog*) [inline], [inherited]

Definition at line 73 of file ui_kbprofiledialog.h.

Referenced by Ui_KbProfileDialog::setupUi().

```
74     {
75         KbProfileDialog->setWindowTitle(QApplication::translate("KbProfileDialog", "Keyboard Profiles",
       Q_NULLPTR));
76     } // retranslateUi
```

Here is the caller graph for this function:



12.37.2.2 void Ui_KbProfileDialog::setupUi (QDialog * *KbProfileDialog*) [inline],[inherited]

Definition at line 34 of file ui_kbprofiledialog.h.

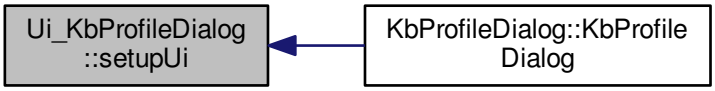
References Ui_KbProfileDialog::buttonBox, Ui_KbProfileDialog::gridLayout, Ui_KbProfileDialog::horizontalSpacer, Ui_KbProfileDialog::horizontalSpacer_2, Ui_KbProfileDialog::profileList, and Ui_KbProfileDialog::retranslateUi().

Referenced by KbProfileDialog::KbProfileDialog().

```
35     {
36         if (KbProfileDialog->objectName().isEmpty())
37             KbProfileDialog->setObjectName(QStringLiteral("KbProfileDialog"));
38         KbProfileDialog->resize(400, 300);
39         KbProfileDialog->setFocusPolicy(Qt::ClickFocus);
40         gridLayout = new QGridLayout(KbProfileDialog);
41         gridLayout->setObjectName(QStringLiteral("gridLayout"));
42         horizontalSpacer = new QSpacerItem(40, 20, QSizePolicy::Expanding,
       QSizePolicy::Minimum);
43
44         gridLayout->addItem(horizontalSpacer, 0, 0, 1, 1);
45
46         horizontalSpacer_2 = new QSpacerItem(40, 20, QSizePolicy::Expanding,
       QSizePolicy::Minimum);
47
48         gridLayout->addItem(horizontalSpacer_2, 0, 2, 1, 1);
49
50         profileList = new RListWidget(KbProfileDialog);
51         profileList->setObjectName(QStringLiteral("profileList"));
52         profileList->setContextMenuPolicy(Qt::CustomContextMenu);
53         profileList->setFrameShadow(QFrame::Plain);
54         profileList->setIconSize(QSize(24, 24));
55
56         gridLayout->addWidget(profileList, 0, 1, 1, 1);
57
58         buttonBox = new QDialogButtonBox(KbProfileDialog);
59         buttonBox->setObjectName(QStringLiteral("buttonBox"));
60         buttonBox->setOrientation(Qt::Horizontal);
61         buttonBox->setStandardButtons(QDialogButtonBox::Ok);
62
63         gridLayout->addWidget(buttonBox, 1, 0, 1, 3);
64
65
66         retranslateUi(KbProfileDialog);
67         QObject::connect(buttonBox, SIGNAL(accepted()), KbProfileDialog, SLOT(accept()));
68         QObject::connect(buttonBox, SIGNAL(rejected()), KbProfileDialog, SLOT(reject()));
69
70         QMetaObject::connectSlotsByName(KbProfileDialog);
71     } // setupUi
```

Here is the call graph for this function:

Here is the caller graph for this function:



12.37.3 Field Documentation

12.37.3.1 QDialogButtonBox* Ui_KbProfileDialog::buttonBox [inherited]

Definition at line 32 of file ui_kbprofiledialog.h.

Referenced by Ui_KbProfileDialog::setupUi().

12.37.3.2 QGridLayout* Ui_KbProfileDialog::gridLayout [inherited]

Definition at line 28 of file ui_kbprofiledialog.h.

Referenced by Ui_KbProfileDialog::setupUi().

12.37.3.3 QSpacerItem* Ui_KbProfileDialog::horizontalSpacer [inherited]

Definition at line 29 of file ui_kbprofiledialog.h.

Referenced by Ui_KbProfileDialog::setupUi().

12.37.3.4 QSpacerItem* Ui_KbProfileDialog::horizontalSpacer_2 [inherited]

Definition at line 30 of file ui_kbprofiledialog.h.

Referenced by Ui_KbProfileDialog::setupUi().

12.37.3.5 RListWidget* Ui_KbProfileDialog::profileList [inherited]

Definition at line 31 of file ui_kbprofiledialog.h.

Referenced by KbProfileDialog::addNewProfileItem(), KbProfileDialog::KbProfileDialog(), KbProfileDialog::on_profileList_customContextMenuRequested(), KbProfileDialog::on_profileList_itemClicked(), KbProfileDialog::profileList_reordered(), KbProfileDialog::repopulate(), and Ui_KbProfileDialog::setupUi().

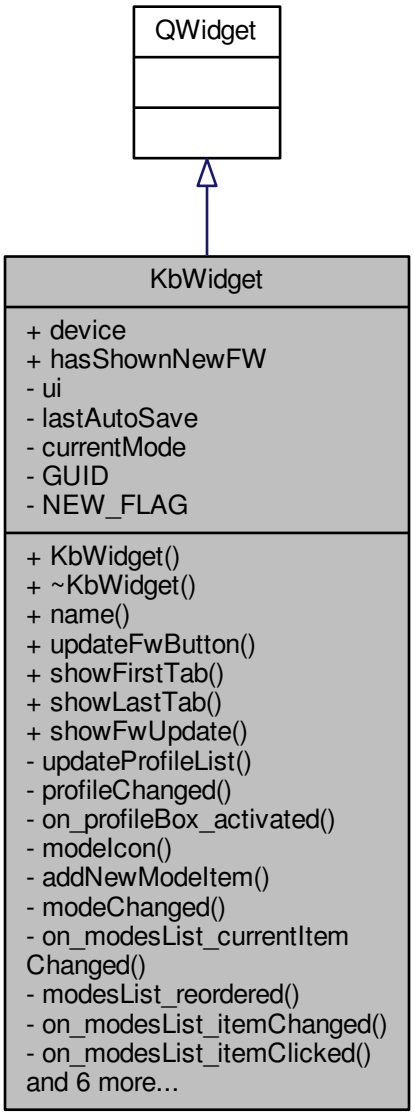
The documentation for this class was generated from the following file:

- src/ckb/ui_kbprofiledialog.h

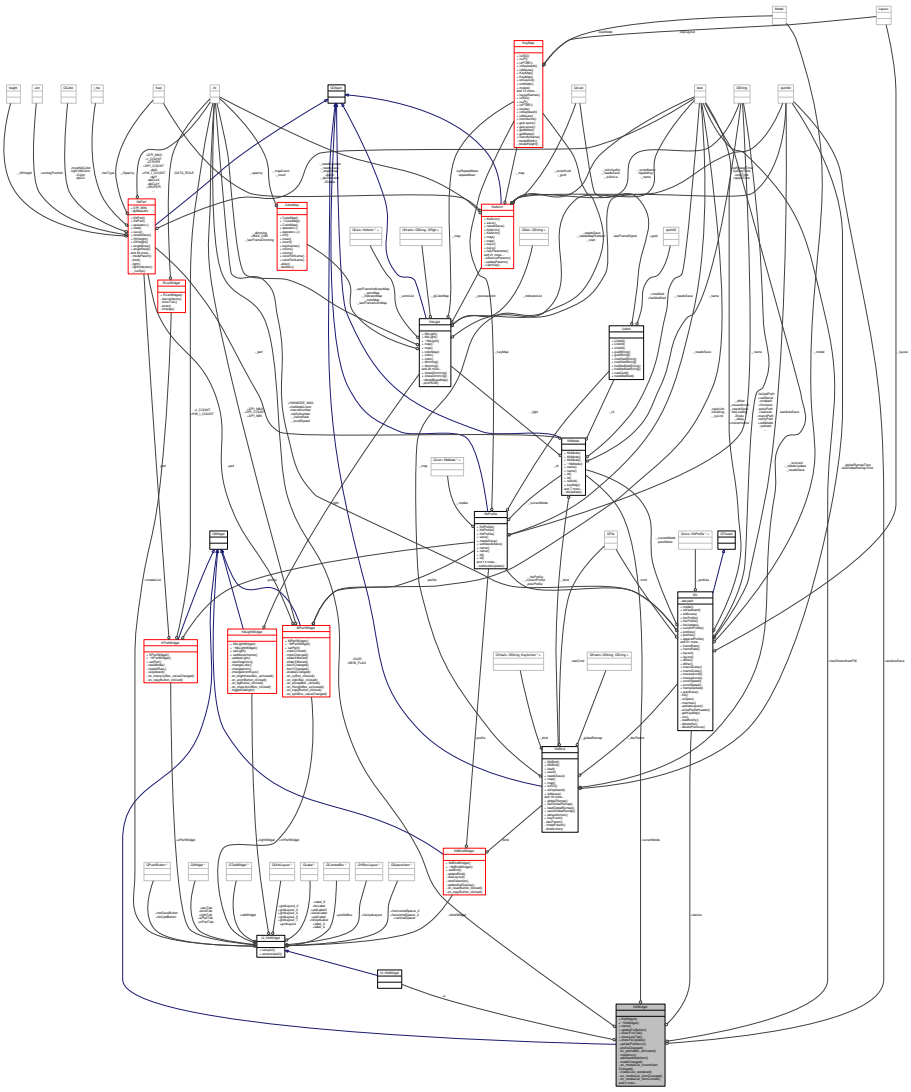
12.38 KbWidget Class Reference

`#include <src/ckb/kbwidget.h>`

Inheritance diagram for KbWidget:



Collaboration diagram for KbWidget:



Public Slots

- void showFirstTab ()
- void showLastTab ()
- void showFwUpdate ()

Public Member Functions

- KbWidget (QWidget *parent, Kb *_device)
- ~KbWidget ()
- QString name () const
- void updateFwButton ()

Data Fields

- Kb * device
- bool hasShownNewFW

Private Slots

- void updateProfileList ()
- void profileChanged ()
- void on_profileBox_activated (int index)
- QIcon modeIcon (int i)
- void addNewModeItem ()
- void modeChanged (bool spontaneous=true)
- void on_modesList_currentItemChanged (QListWidgetItem ∗current, QListWidgetItem ∗previous)
- void modesList_reordered ()
- void on_modesList_itemChanged (QListWidgetItem ∗item)
- void on_modesList_itemClicked (QListWidgetItem ∗item)
- void on_modesList_customContextMenuRequested (const QPoint &pos)

  KbWidget::on_modesList_customContextMenuRequested.
- void devUpdate ()
- void modeUpdate ()
- void on_hwSaveButton_clicked ()
- void on_tabWidget_currentChanged (int index)
- void on_fwUpdButton_clicked ()

Private Attributes

- Ui::KbWidget ∗ ui
- quint64 lastAutoSave
- KbMode ∗ currentMode

Static Private Attributes

- static const int GUID = Qt::UserRole
- static const int NEW_FLAG = Qt::UserRole + 1

12.38.1 Detailed Description

Definition at line 15 of file kbwidget.h.

12.38.2 Constructor & Destructor Documentation

12.38.2.1 KbWidget::KbWidget (QWidget ∗ *parent,* Kb ∗ *_device*) [explicit]

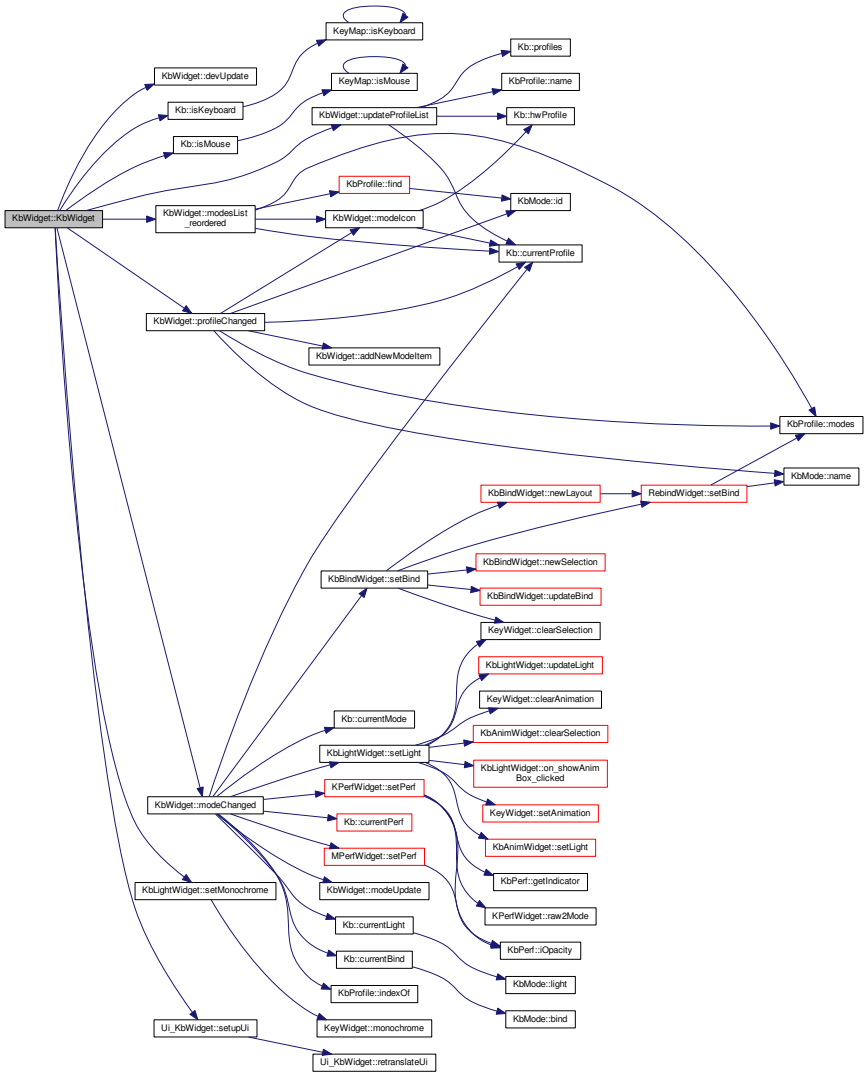
Definition at line 15 of file kbwidget.cpp.

References device, devUpdate(), Kb::features, Ui_KbWidget::fwUpdButton, Ui_KbWidget::fwUpdLabel, Ui_KbWidget::fwUpdLayout, Kb::isKeyboard(), Kb::isMouse(), Ui_KbWidget::kPerfTab, Ui_KbWidget::lightTab, Ui_KbWidget::lightWidget, modeChanged(), Ui_KbWidget::modesList, modesList_reordered(), Kb::monochrome, Ui_KbWidget::mPerfTab, Ui_KbWidget::pollLabel, Ui_KbWidget::pollLabel2, profileChanged(), KbLightWidget::setMonochrome(), Ui_KbWidget::setupUi(), Ui_KbWidget::tabWidget, ui, and updateProfileList().

```
15                                          :
16     QWidget(parent),
17     device(_device), hasShownNewFW(false),
18     ui(new Ui::KbWidget),
19     currentMode(0)
20 {
21     ui->setupUi(this);
22     connect(ui->modesList, SIGNAL(orderChanged()), this, SLOT(
       modesList_reordered()));
23
```

```
24      connect(device, SIGNAL(infoUpdated()), this, SLOT(devUpdate()));
25      connect(device, SIGNAL(profileAdded()), this, SLOT(updateProfileList()));
26      connect(device, SIGNAL(profileChanged()), this, SLOT(
    profileChanged()));
27      connect(device, SIGNAL(profileRenamed()), this, SLOT(updateProfileList()));
28      connect(device, SIGNAL(modeRenamed()), this, SLOT(profileChanged()));
29      connect(device, SIGNAL(modeRenamed()), this, SLOT(modeChanged()));
30      connect(device, SIGNAL(modeChanged(bool)), this, SLOT(
    modeChanged(bool)));
31
32      // Remove the Lighting and Performance tabs from non-RGB keyboards
33      if(!device->features.contains("rgb")){
34          ui->tabWidget->removeTab(ui->tabWidget->indexOf(ui->
    lightTab));
35          ui->tabWidget->removeTab(ui->tabWidget->indexOf(ui->
    kPerfTab));
36          ui->tabWidget->removeTab(ui->tabWidget->indexOf(ui->
    mPerfTab));
37      } else {
38          // Remove mouse Performance tab from non-mice
39          if(!device->isMouse())
40              ui->tabWidget->removeTab(ui->tabWidget->indexOf(
    ui->mPerfTab));
41          // Remove keyboard Performance tab from non-keyboards
42          if(!device->isKeyboard())
43              ui->tabWidget->removeTab(ui->tabWidget->indexOf(
    ui->kPerfTab));
44      }
45      // Hide poll rate and FW update as appropriate
46      if(!device->features.contains("pollrate")){
47          ui->pollLabel->hide();
48          ui->pollLabel2->hide();
49      }
50      if(!device->features.contains("fwupdate")){
51          ui->fwUpdButton->hide();
52          ui->fwUpdLabel->hide();
53          ui->fwUpdLayout->removeItem(ui->fwUpdLayout->itemAt(1));
54      }
55      // Set monochrome mode according to hardware
56      if(device->monochrome)
57          ui->lightWidget->setMonochrome();
58  }
```

Here is the call graph for this function:



12.38.2.2 KbWidget::~KbWidget ( )

Definition at line 60 of file kbwidget.cpp.

References ui.

```
60                      {
61     delete ui;
62 }
```

12.38.3 Member Function Documentation

12.38.3.1 void KbWidget::addNewModelItem ( ) [private], [slot]

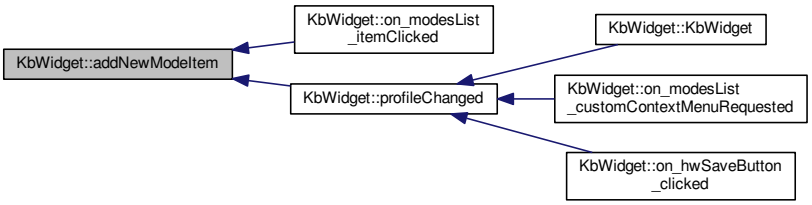
Definition at line 136 of file kbwidget.cpp.

References Ui_KbWidget::modesList, NEW_FLAG, and ui.

Referenced by on_modesList_itemClicked(), and profileChanged().

```
136                             {
137     // Add an item for creating a new mode. Make it editable but not dragable.
138     QListWidgetItem* item = new QListWidgetItem("New mode...", ui->modesList);
139     item->setFlags((item->flags() | Qt::ItemIsEditable) & ~Qt::ItemIsDragEnabled & ~Qt::ItemIsDropEnabled);
140     item->setData(NEW_FLAG, 1);
141     QFont font = item->font();
142     font.setItalic(true);
143     item->setFont(font);
144     item->setIcon(QIcon(":/img/icon_plus.png"));
145     ui->modesList->addItem(item);
146 }
```

Here is the caller graph for this function:



12.38.3.2 void KbWidget::devUpdate ( ) [private],[slot]

Definition at line 300 of file kbwidget.cpp.

References device, Kb::firmware, Ui_KbWidget::fwLabel, Ui_KbWidget::pollLabel, Kb::pollrate, Ui_KbWidget::serialLabel, ui, and Kb::usbSerial.

Referenced by KbWidget().

```
300                         {
301     // Update device tab
302     ui->serialLabel->setText(device->usbSerial);
303     ui->fwLabel->setText(device->firmware);
304     ui->pollLabel->setText(device->pollrate);
305 }
```

Here is the caller graph for this function:

12.38.3.3 void KbWidget::modeChanged (bool *spontaneous* = true) [private],[slot]

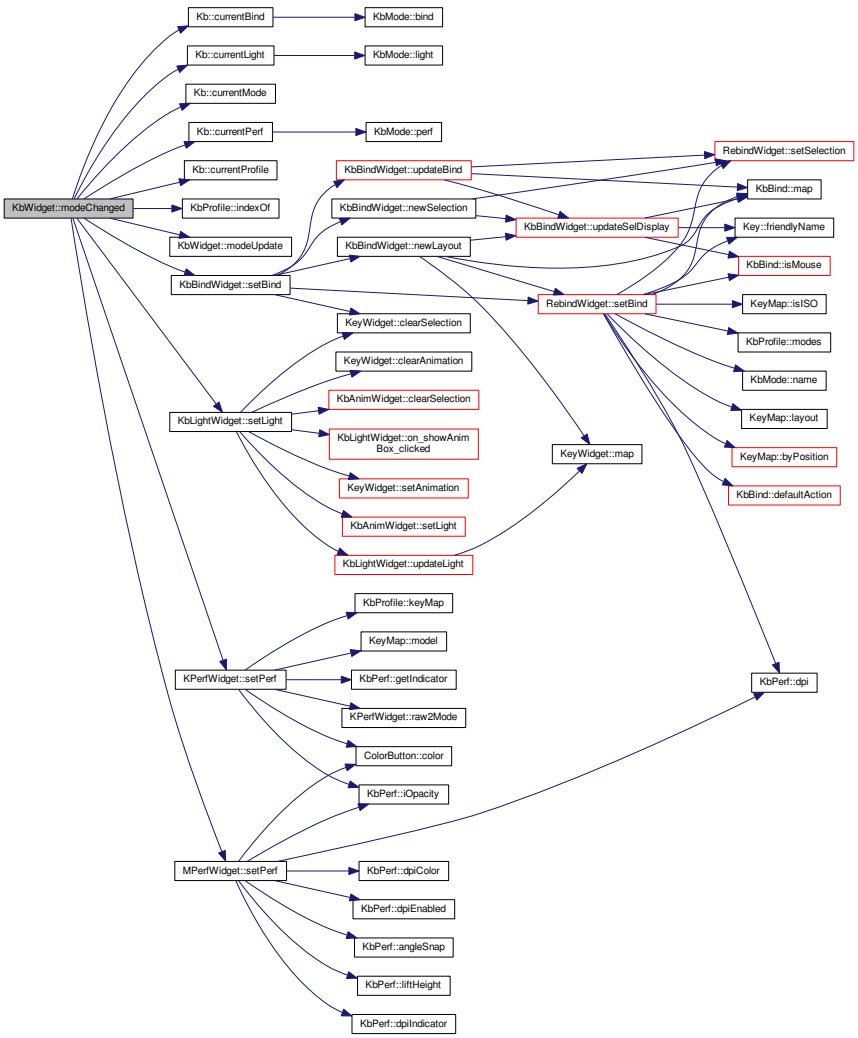
Definition at line 148 of file kbwidget.cpp.

References Ui_KbWidget::bindWidget, Kb::currentBind(), Kb::currentLight(), currentMode, Kb::currentMode(), Kb::currentPerf(), Kb::currentProfile(), device, KbProfile::indexOf(), Ui_KbWidget::kPerfWidget, Ui_KbWidget::lightWidget, Ui_KbWidget::modesList, modeUpdate(), Ui_KbWidget::mPerfWidget, KbBindWidget::setBind(), KbLightWidget::setLight(), KPerfWidget::setPerf(), MPerfWidget::setPerf(), and ui.

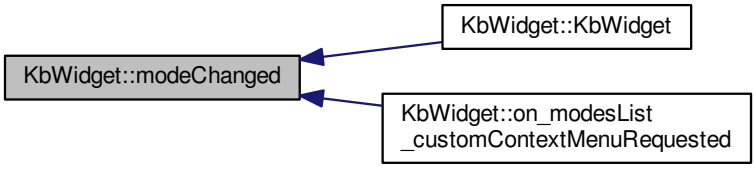
Referenced by KbWidget(), and on_modesList_customContextMenuRequested().

```
148                                         {
149     int index = device->currentProfile()->indexOf(
      device->currentMode());
150     if(index < 0)
151         return;
152     // Update tabs
153     ui->lightWidget->setLight(device->currentLight());
154     ui->bindWidget->setBind(device->currentBind(),
      device->currentProfile());
155     ui->kPerfWidget->setPerf(device->currentPerf(),
      device->currentProfile());
156     ui->mPerfWidget->setPerf(device->currentPerf(),
      device->currentProfile());
157     // Update selection
158     if(spontaneous)
159         ui->modesList->setCurrentRow(index);
160     // Connect signals
161     if(currentMode)
162         disconnect(currentMode, SIGNAL(updated()), this, SLOT(
      modeUpdate()));
163     currentMode = device->currentMode();
164     connect(currentMode, SIGNAL(updated()), this, SLOT(modeUpdate()));
165     modeUpdate();
166 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.38.3.4 QIcon KbWidget::modeIcon (int *i*) [private], [slot]

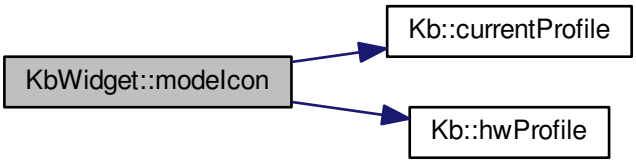
Definition at line 127 of file kbwidget.cpp.

References Kb::currentProfile(), device, Kb::hwModeCount, and Kb::hwProfile().

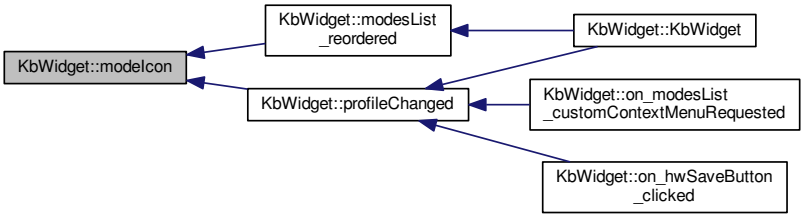
Referenced by modesList_reordered(), and profileChanged().

```
127                             {
128     KbProfile* currentProfile = device->currentProfile(), *hwProfile =
      device->hwProfile();
129     int hwModeCount = device->hwModeCount;
130     if(i >= hwModeCount)
131         return QIcon(":/img/icon_mode.png");
132     else
133         return QIcon(QString(currentProfile == hwProfile ? ":/img/icon_mode%1_hardware.png" : "
      :/img/icon_mode%1.png").arg(i + 1));
134 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.38.3.5 void KbWidget::modesList_reordered ( ) [private],[slot]

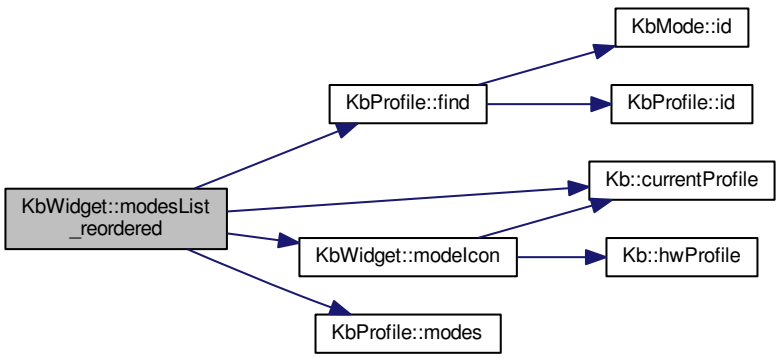
Definition at line 177 of file kbwidget.cpp.

References Kb::currentProfile(), device, KbProfile::find(), GUID, modelcon(), KbProfile::modes(), Ui_KbWidget-::modesList, NEW_FLAG, and ui.

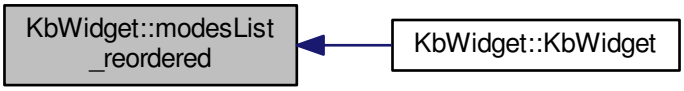
Referenced by KbWidget().

```
177                                 {
178     KbProfile* currentProfile = device->currentProfile();
179     // Rebuild mode list from items
180     QList<KbMode*> newModes;
181     int count = ui->modesList->count();
182     for(int i = 0; i < count; i++){
183         QListWidgetItem* item = ui->modesList->item(i);
184         KbMode* mode = currentProfile->find(item->data(GUID).toUuid());
185         if(mode && !newModes.contains(mode))
186             newModes.append(mode);
187         if(item->data(NEW_FLAG).toInt() != 1)
188             item->setIcon(modeIcon(i));
189         item->setFlags(item->flags() | Qt::ItemIsEditable);
190     }
191     // Add any missing modes at the end of the list
192     foreach(KbMode* mode, currentProfile->modes()){
193         if(!newModes.contains(mode))
194             newModes.append(mode);
195     }
196     currentProfile->modes(newModes);
197 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.38.3.6 void KbWidget::modeUpdate ( ) [private],[slot]

Definition at line 307 of file kbwidget.cpp.

Referenced by modeChanged().

```
307                             {
308 }
```

Here is the caller graph for this function:

12.38.3.7 QString KbWidget::name ( ) const [inline]

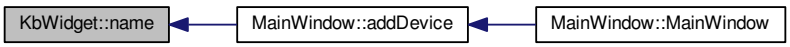
Definition at line 25 of file kbwidget.h.

References device, and Kb::usbModel.

Referenced by MainWindow::addDevice().

```
25 { return device->usbModel; }
```

Here is the caller graph for this function:



12.38.3.8 void KbWidget::on_fwUpdButton_clicked ( ) [private],[slot]

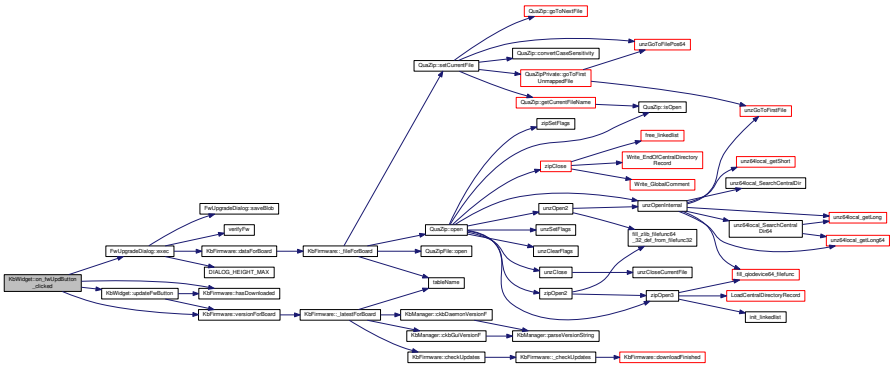
Definition at line 339 of file kbwidget.cpp.

References device, FwUpgradeDialog::exec(), Kb::features, Kb::firmware, Ui_KbWidget::fwUpdButton, KbFirmware::hasDownloaded(), ui, updateFwButton(), and KbFirmware::versionForBoard().

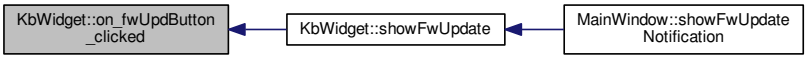
Referenced by showFwUpdate().

```
339                                     {
340     // If alt is pressed, ignore upgrades and go straight to the manual prompt
341     if(!(qApp->keyboardModifiers() & Qt::AltModifier)){
342         // Check version numbers
343         if(!KbFirmware::hasDownloaded()){
344             ui->fwUpdButton->setText("Checking...");
345             ui->fwUpdButton->setEnabled(false);
346         }
347         float newVersion = KbFirmware::versionForBoard(
    device->features, true);
348         float oldVersion = device->firmware.toFloat();
349         ui->fwUpdButton->setEnabled(true);
350         updateFwButton();
351         if(newVersion == -1.f){
352             QMessageBox::information(this, "Firmware update", "<center>There is a new firmware available
     for this device.<br />However, it requires a newer version of ckb.<br />Please upgrade ckb and try
     again.</center>");
353             return;
354         } else if(newVersion == 0.f){
355             if(QMessageBox::question(this, "Firmware update", "<center>There was a problem getting the
     status for this device.<br />Would you like to select a file manually?</center>") != QMessageBox::Yes)
356                 return;
357             // "Yes" -> fall through to browse file
358         } else if(newVersion <= oldVersion){
359             if(QMessageBox::question(this, "Firmware update", "<center>Your firmware is already up to
     date.<br />Would you like to select a file manually?</center>") != QMessageBox::Yes)
360                 return;
361             // "Yes" -> fall through to browse file
362         } else {
363             // Automatic upgrade. Fetch file from web.
364             // FwUpgradeDialog can't be parented to KbWidget because KbWidget may be deleted before the
     dialog exits
365             FwUpgradeDialog dialog(parentWidget(), newVersion, "",
    device);
366             dialog.exec();
367             return;
368         }
369     }
370     // Browse for file
371     QString path = QFileDialog::getOpenFileName(this, "Select firmware file",
    QStandardPaths::writableLocation(QStandardPaths::DownloadLocation), "Firmware blobs (*.bin)");
372     if(path.isEmpty())
373         return;
374     QFile file(path);
375     if(!file.open(QIODevice::ReadOnly)){
376         QMessageBox::warning(parentWidget(), "Error", "<center>File could not be read.</center>");
377         return;
378     }
379     QByteArray blob = file.readAll();
380     FwUpgradeDialog dialog(parentWidget(), 0.f, blob, device);
381     dialog.exec();
382 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



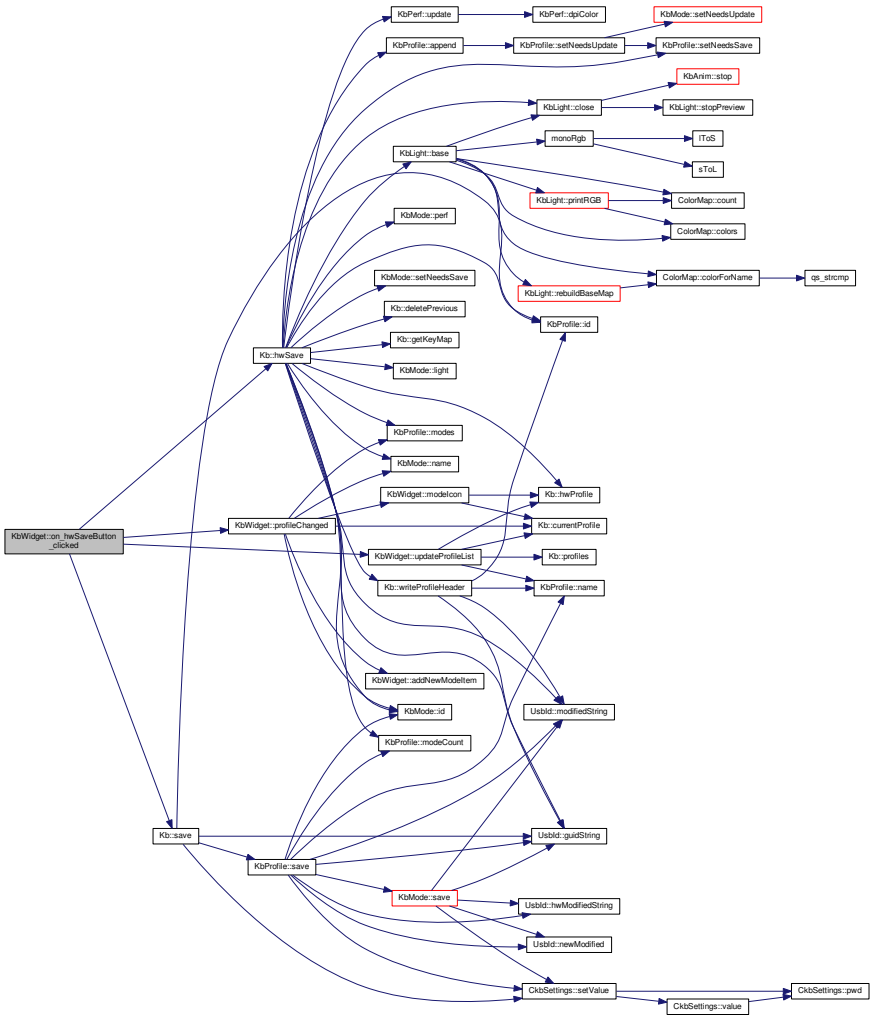
12.38.3.9 void KbWidget::on_hwSaveButton_clicked ( ) [private], [slot]

Definition at line 310 of file kbwidget.cpp.

References device, Kb::hwSave(), profileChanged(), Kb::save(), and updateProfileList().

```
310                                    {
311     device->save();
312     device->hwSave();
313     updateProfileList();
314     profileChanged();
315 }
```

Here is the call graph for this function:



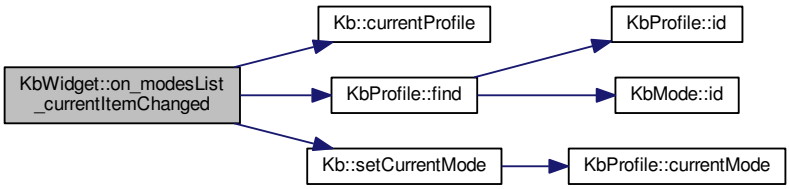
12.38.3.10 void KbWidget::on_modesList_currentItemChanged (QListWidgetItem * *current*, QListWidgetItem * *previous*) [private], [slot]

Definition at line 168 of file kbwidget.cpp.

References Kb::currentProfile(), device, KbProfile::find(), GUID, and Kb::setCurrentMode().

```
168                                                                                                 {
169     if(!current)
170         return;
171     KbMode* mode = device->currentProfile()->find(current->data(
    GUID).toUuid());
172     if(!mode)
173         return;
174     device->setCurrentMode(mode, false);
175 }
```

Here is the call graph for this function:



12.38.3.11 void KbWidget::on_modesList_customContextMenuRequested (const QPoint & *pos*) [private], [slot]

Parameters

pos	Opens on right click in the profiles list a context sensitive menue at position pos.

When clicking on a command it is located and executed.

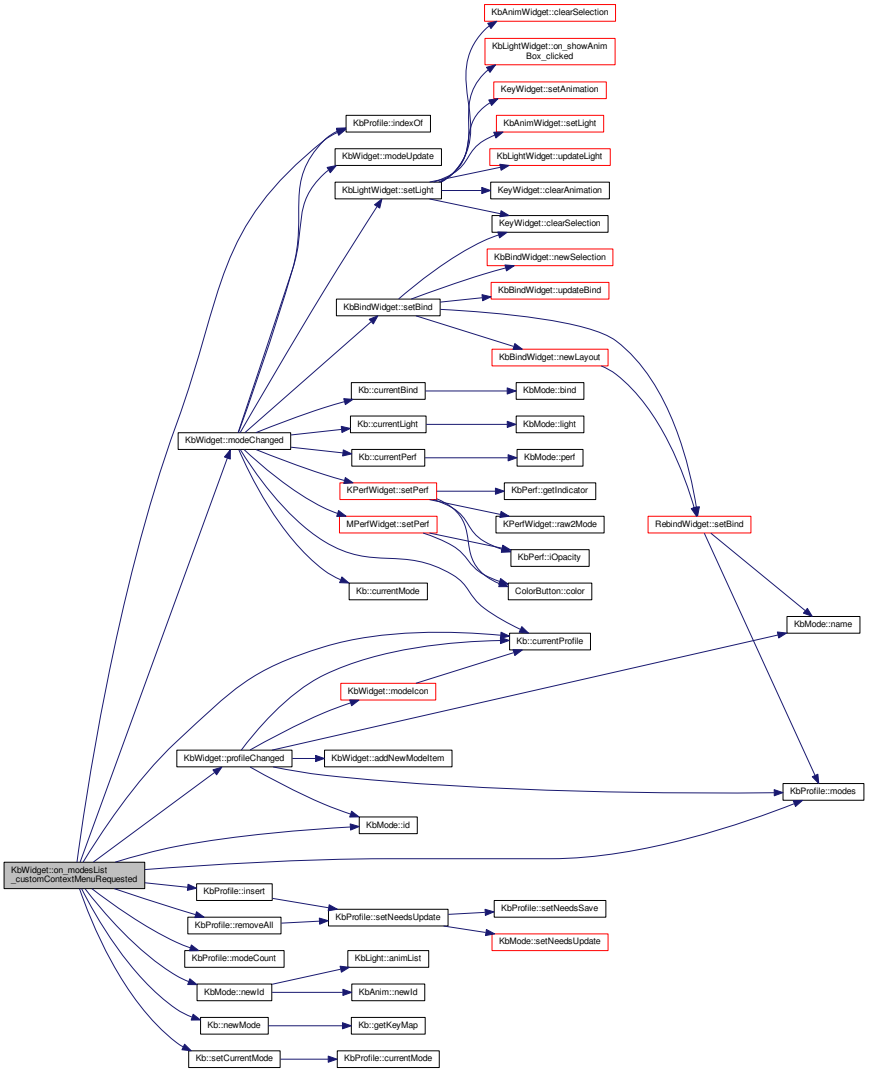
Definition at line 236 of file kbwidget.cpp.

References currentMode, Kb::currentProfile(), device, UsbId::guid, GUID, Kb::hwModeCount, KbMode::id(), KbProfile::indexOf(), KbProfile::insert(), modeChanged(), KbProfile::modeCount(), KbProfile::modes(), Ui_KbWidget::modesList, KbMode::newId(), Kb::newMode(), profileChanged(), KbProfile::removeAll(), Kb::setCurrentMode(), and ui.

```
236                                                                    {
237     QListWidgetItem* item = ui->modesList->itemAt(pos);
238     if(!item || !currentMode || item->data(GUID).toUuid() !=
    currentMode->id().guid)
239         return;
240     KbProfile* currentProfile = device->currentProfile();
241     int index = currentProfile->indexOf(currentMode);
242
243     QMenu menu(this);
244     QAction* rename = new QAction("Rename...", this);
245     QAction* duplicate = new QAction("Duplicate", this);
246     QAction* del = new QAction("Delete", this);
247     bool canDelete = (device->currentProfile()->modeCount() >
    device->hwModeCount);
248     if(!canDelete)
249         // Can't delete modes if they're required by hardware
250         del->setEnabled(false);
251     QAction* moveup = new QAction("Move Up", this);
252     if(index == 0)
253         moveup->setEnabled(false);
254     QAction* movedown = new QAction("Move Down", this);
255     if(index >= currentProfile->modeCount() - 1)
256         movedown->setEnabled(false);
257     menu.addAction(rename);
258     menu.addAction(duplicate);
259     menu.addAction(del);
260     menu.addSeparator();
261     menu.addAction(moveup);
262     menu.addAction(movedown);
263     QAction* result = menu.exec(QCursor::pos());
264     if(result == rename){
265         ui->modesList->editItem(item);
266     } else if(result == duplicate){
267         KbMode* newMode = device->newMode(currentMode);
268         newMode->newId();
269         currentProfile->insert(index + 1, newMode);
270         // Update UI
271         profileChanged();
272         device->setCurrentMode(newMode);
273     } else if(result == del){
274         if(!canDelete)
275             return;
276         currentProfile->removeAll(currentMode);
277         currentMode->deleteLater();
```

```
278         currentMode = 0;
279         // Select next mode
280         profileChanged();
281         if(index < currentProfile->modeCount())
282             device->setCurrentMode(currentProfile->modes()[index]);
283         else
284             device->setCurrentMode(currentProfile->modes().last());
285     } else if(result == moveup){
286         currentProfile->removeAll(currentMode);
287         currentProfile->insert(index - 1, currentMode);
288         // Update UI
289         profileChanged();
290         modeChanged(true);
291     } else if(result == movedown){
292         currentProfile->removeAll(currentMode);
293         currentProfile->insert(index + 1, currentMode);
294         // Update UI
295         profileChanged();
296         modeChanged(true);
297     }
298 }
```

Here is the call graph for this function:



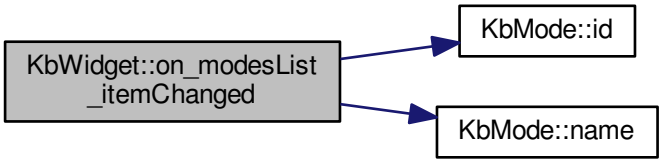
12.38.3.12 void KbWidget::on_modesList_itemChanged (QListWidgetItem * *item*) [private],[slot]

Definition at line 199 of file kbwidget.cpp.

References currentMode, UsbId::guid, GUID, KbMode::id(), and KbMode::name().

```
199                                                         {
200     if(!item || !currentMode || item->data(GUID).toUuid() !=
      currentMode->id().guid)
201         return;
202     currentMode->name(item->text());
203     // Set the text to the actual name (trimmed, "" replaced with "Unnamed")
204     item->setText(currentMode->name());
205 }
```

Here is the call graph for this function:



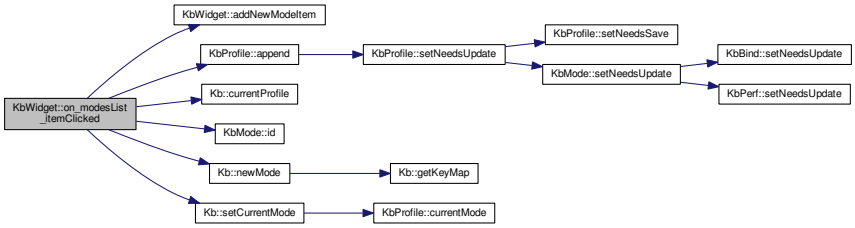
12.38.3.13 void KbWidget::on_modesList_itemClicked (QListWidgetItem * *item*) [private],[slot]

Definition at line 207 of file kbwidget.cpp.

References addNewModeItem(), KbProfile::append(), Kb::currentProfile(), device, UsbId::guid, GUID, KbMode::id(), Ui_KbWidget::modesList, NEW_FLAG, Kb::newMode(), Kb::setCurrentMode(), and ui.

```
207                                                         {
208     QUuid guid = item->data(GUID).toUuid();
209     if(guid.isNull() && item->data(NEW_FLAG).toInt() == 1){
210         // "New mode" item. Clear text and start editing
211         item->setText("");
212         ui->modesList->editItem(item);
213         item->setFlags(item->flags() | Qt::ItemIsEditable | Qt::ItemIsDragEnabled | Qt::ItemIsDropEnabled);
214         QFont font = item->font();
215         font.setItalic(false);
216         item->setFont(font);
217         item->setIcon(QIcon(":/img/icon_mode.png"));
218         // Add the new mode and assign it to this item
219         KbMode* newMode = device->newMode();
220         device->currentProfile()->append(newMode);
221         item->setData(GUID, newMode->id().guid);
222         item->setData(NEW_FLAG, 0);
223         device->setCurrentMode(newMode, false);
224         // Create another "new mode" item to replace this one
225         addNewModeItem();
226     }
227 }
```

Here is the call graph for this function:



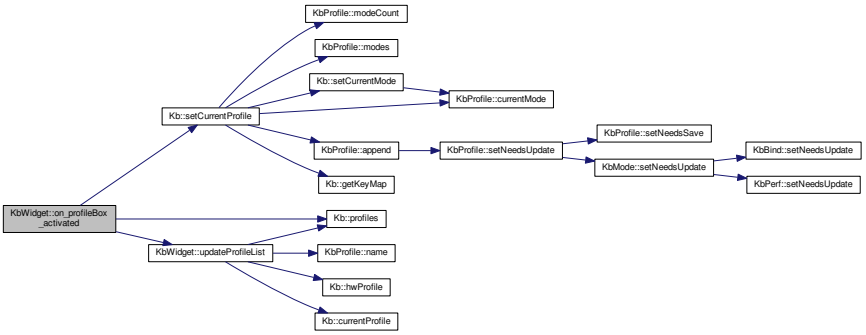
12.38.3.14 void KbWidget::on_profileBox_activated (int *index*) [private], [slot]

Definition at line 113 of file kbwidget.cpp.

References device, Kb::profiles(), Kb::setCurrentProfile(), and updateProfileList().

```
113                                                 {
114     if(index < 0)
115         return;
116     if(index >= device->profiles().count()){
117         // "Manage profiles" option
118         KbProfileDialog dialog(this);
119         dialog.exec();
120         updateProfileList();
121         return;
122     }
123     device->setCurrentProfile(device->profiles()[index]);
124     // Device will emit profileChanged() and modeChanged() signals to update UI
125 }
```

Here is the call graph for this function:



12.38.3.15 void KbWidget::on_tabWidget_currentChanged (int *index*) [private], [slot]

Definition at line 317 of file kbwidget.cpp.

References device, Ui_KbWidget::tabWidget, ui, and updateFwButton().

```
317                                                     {
318     if(!device)
319         return;
320     if(index == ui->tabWidget->count() - 1){
321         // Device tab
```

```
322         updateFwButton();
323     }
324 }
```

Here is the call graph for this function:

12.38.3.16 void KbWidget::profileChanged ( ) [private], [slot]

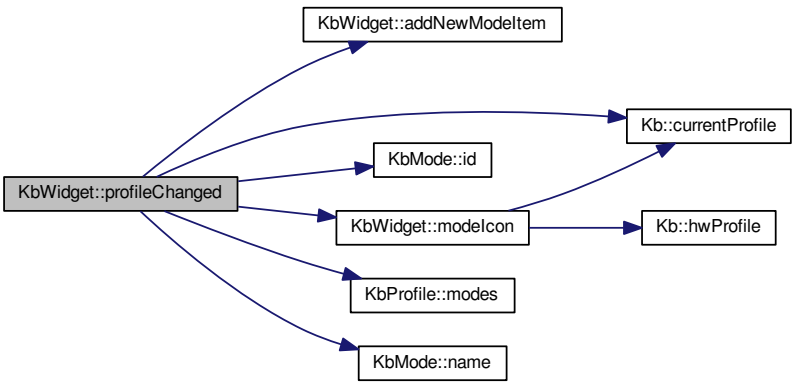
Definition at line 91 of file kbwidget.cpp.

References addNewModeItem(), currentMode, Kb::currentProfile(), device, UsbId::guid, GUID, KbMode::id(), modeIcon(), KbProfile::modes(), Ui_KbWidget::modesList, KbMode::name(), and ui.

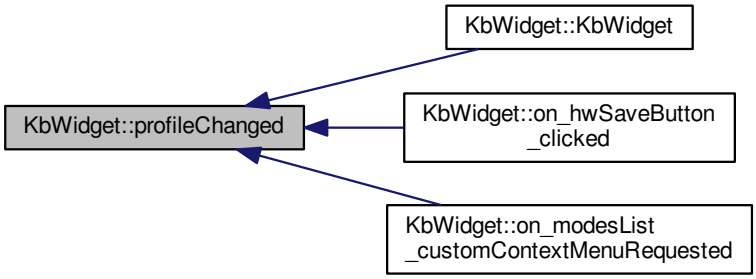
Referenced by KbWidget(), on_hwSaveButton_clicked(), and on_modesList_customContextMenuRequested().

```
91                          {
92     // Rebuild mode list
93     ui->modesList->clear();
94     int i = 0;
95     QListWidgetItem* current = 0;
96     foreach(KbMode* mode, device->currentProfile()->
      modes()){
97         QListWidgetItem* item = new QListWidgetItem(modeIcon(i), mode->
      name(), ui->modesList);
98         item->setData(GUID, mode->id().guid);
99         item->setFlags(item->flags() | Qt::ItemIsEditable);
100         if(mode == currentMode){
101             item->setSelected(true);
102             current = item;
103         }
104         ui->modesList->addItem(item);
105         i++;
106     }
107     if(current)
108         ui->modesList->setCurrentItem(current);
109     addNewModeItem();
110     // Wait for modeChanged() to refresh the rest of the UI
111 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.38.3.17 void KbWidget::showFirstTab ( ) [slot]

Definition at line 64 of file kbwidget.cpp.

References Ui_KbWidget::tabWidget, and ui.

```
64                          {
65     ui->tabWidget->setCurrentIndex(0);
66 }
```

12.38.3.18 void KbWidget::showFwUpdate ( ) [inline], [slot]

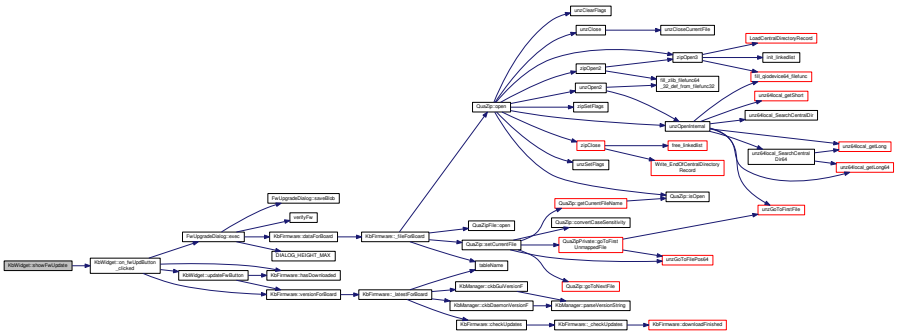
Definition at line 38 of file kbwidget.h.

References on_fwUpdButton_clicked().

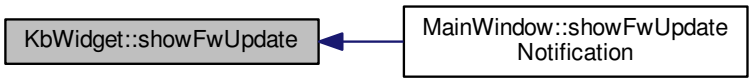
Referenced by MainWindow::showFwUpdateNotification().

```
38 { on_fwUpdButton_clicked(); }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.38.3.19 void KbWidget::showLastTab ( ) [slot]

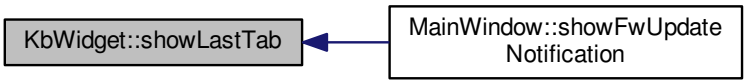
Definition at line 68 of file kbwidget.cpp.

References Ui_KbWidget::tabWidget, and ui.

Referenced by MainWindow::showFwUpdateNotification().

```
68                          {
69     ui->tabWidget->setCurrentIndex(ui->tabWidget->count() - 1);
70 }
```

Here is the caller graph for this function:



12.38.3.20 void KbWidget::updateFwButton ( )

Definition at line 326 of file kbwidget.cpp.

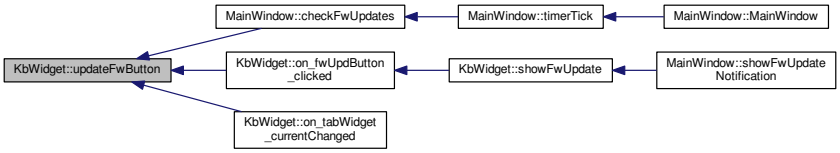
References device, Kb::features, Kb::firmware, Ui_KbWidget::fwUpdButton, KbFirmware::hasDownloaded(), ui, and KbFirmware::versionForBoard().

Referenced by MainWindow::checkFwUpdates(), on_fwUpdButton_clicked(), and on_tabWidget_currentChanged().

```
326                             {
327     if(!KbFirmware::hasDownloaded())
328         ui->fwUpdButton->setText("Check for updates");
329     else {
330         float newVersion = KbFirmware::versionForBoard(
      device->features);
331         float oldVersion = device->firmware.toFloat();
332         if(newVersion <= 0.f || newVersion <= oldVersion)
333             ui->fwUpdButton->setText("Up to date");
334         else
335             ui->fwUpdButton->setText(tr("Upgrade to v%1").arg(QString::number(newVersion, 'f',
       2)));
336     }
337 }
```

Here is the call graph for this function:

Here is the caller graph for this function:



12.38.3.21 void KbWidget::updateProfileList ( ) [private], [slot]

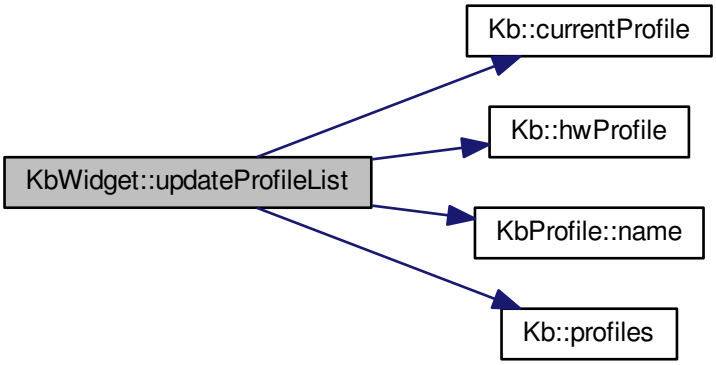
Definition at line 73 of file kbwidget.cpp.

References Kb::currentProfile(), device, Kb::hwProfile(), KbProfile::name(), Ui_KbWidget::profileBox, Kb::profiles(), and ui.

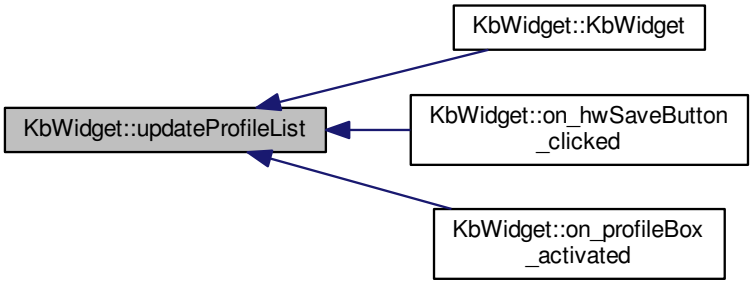
Referenced by KbWidget(), on_hwSaveButton_clicked(), and on_profileBox_activated().

```
73                                {
74     // Clear profile list and rebuild
75     KbProfile* hwProfile = device->hwProfile(), *currentProfile =
       device->currentProfile();
76     ui->profileBox->clear();
77     int i = 0;
78     foreach(KbProfile* profile, device->profiles()){
79         ui->profileBox->addItem((profile == hwProfile) ? QIcon(":/img/icon_profile_hardware.png
       ") : QIcon(":/img/icon_profile.png"),
80                                 profile->name());
81         if(profile == currentProfile)
82             ui->profileBox->setCurrentIndex(i);
83         i++;
84     }
85     ui->profileBox->addItem(QIcon(":/img/icon_blank.png"), "Manage profiles...");
86     QFont font = ui->profileBox->font();
87     font.setItalic(true);
88     ui->profileBox->setItemData(ui->profileBox->count() - 1, font, Qt::FontRole);
89 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.38.4 Field Documentation

12.38.4.1 KbMode* KbWidget::currentMode [private]

Definition at line 44 of file kbwidget.h.

Referenced by modeChanged(), on_modesList_customContextMenuRequested(), on_modesList_itemChanged(), and profileChanged().

12.38.4.2 Kb* KbWidget::device

Definition at line 24 of file kbwidget.h.

Referenced by MainWindow::addDevice(), MainWindow::checkFwUpdates(), devUpdate(), KbWidget(), modeChanged(), modeIcon(), modesList_reordered(), name(), on_fwUpdButton_clicked(), on_hwSaveButton_clicked(), on_modesList_currentItemChanged(), on_modesList_customContextMenuRequested(), on_modesList_item-

Clicked(), on_profileBox_activated(), on_tabWidget_currentChanged(), profileChanged(), MainWindow::removeDevice(), MainWindow::showFwUpdateNotification(), updateFwButton(), and updateProfileList().

12.38.4.3 const int KbWidget::GUID = Qt::UserRole [static], [private]

Definition at line 46 of file kbwidget.h.

Referenced by modesList_reordered(), on_modesList_currentItemChanged(), on_modesList_customContextMenuRequested(), on_modesList_itemChanged(), on_modesList_itemClicked(), and profileChanged().

12.38.4.4 bool KbWidget::hasShownNewFW

Definition at line 28 of file kbwidget.h.

Referenced by MainWindow::checkFwUpdates().

12.38.4.5 quint64 KbWidget::lastAutoSave [private]

Definition at line 42 of file kbwidget.h.

12.38.4.6 const int KbWidget::NEW_FLAG = Qt::UserRole + 1 [static], [private]

Definition at line 47 of file kbwidget.h.

Referenced by addNewModeItem(), modesList_reordered(), and on_modesList_itemClicked().

12.38.4.7 Ui::KbWidget* KbWidget::ui [private]

Definition at line 41 of file kbwidget.h.

Referenced by addNewModeItem(), devUpdate(), KbWidget(), modeChanged(), modesList_reordered(), on_fwUpdButton_clicked(), on_modesList_customContextMenuRequested(), on_modesList_itemClicked(), on_tabWidget_currentChanged(), profileChanged(), showFirstTab(), showLastTab(), updateFwButton(), updateProfileList(), and ~KbWidget().

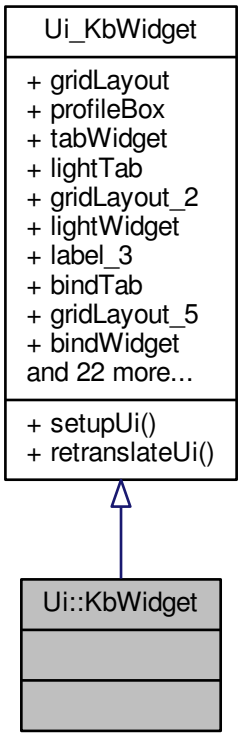
The documentation for this class was generated from the following files:

- src/ckb/kbwidget.h
- src/ckb/kbwidget.cpp

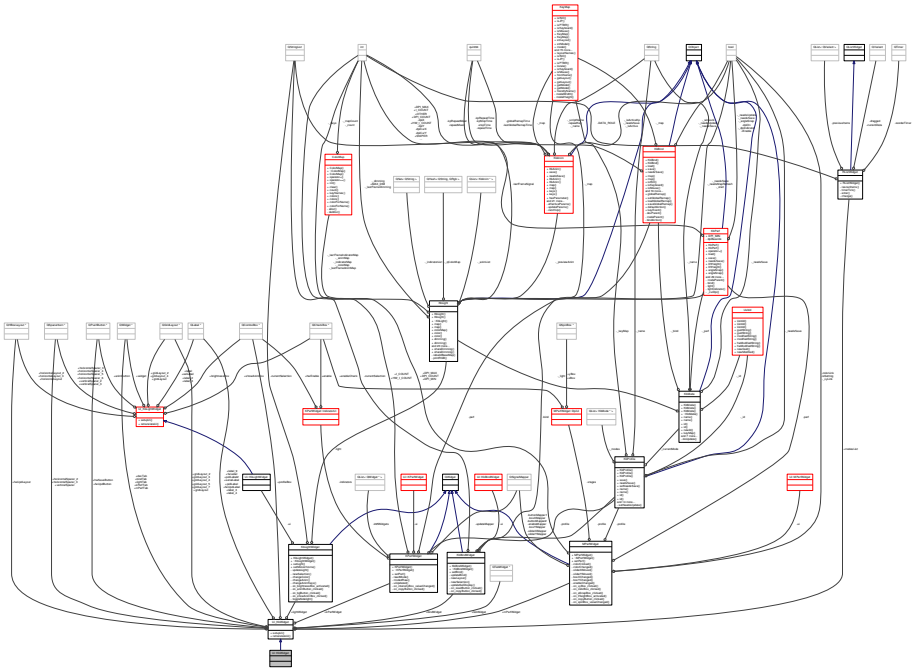
12.39 Ui::KbWidget Class Reference

`#include <src/ckb/ui_kbwidget.h>`

Inheritance diagram for Ui::KbWidget:



Collaboration diagram for Ui::KbWidget:



Public Member Functions

- void setupUi (QWidget *KbWidget)
- void retranslateUi (QWidget *KbWidget)

Data Fields

- QGridLayout * gridLayout
- QComboBox * profileBox
- QTabWidget * tabWidget
- QWidget * lightTab
- QGridLayout * gridLayout_2
- KbLightWidget * lightWidget
- QLabel * label_3
- QWidget * bindTab
- QGridLayout * gridLayout_5
- KbBindWidget * bindWidget
- QWidget * kPerfTab
- QGridLayout * gridLayout_4
- KPerfWidget * kPerfWidget
- QWidget * mPerfTab
- QGridLayout * gridLayout_7
- MPerfWidget * mPerfWidget
- QWidget * devTab
- QGridLayout * gridLayout_3
- QLabel * fwUpdLabel
- QSpacerItem * verticalSpacer
- QLabel * fwLabel
- QLabel * label_4
- QSpacerItem * horizontalSpacer_2
- QLabel * label_9
- QLabel * serialLabel
- QHBoxLayout * fwUpdLayout
- QPushButton * fwUpdButton
- QSpacerItem * horizontalSpacer_3
- QLabel * pollLabel2
- QLabel * pollLabel
- RListWidget * modesList
- QPushButton * hwSaveButton

12.39.1 Detailed Description

Definition at line 291 of file ui_kbwidget.h.

12.39.2 Member Function Documentation

12.39.2.1 void Ui_KbWidget::retranslateUi (QWidget * *KbWidget*) [inline], [inherited]

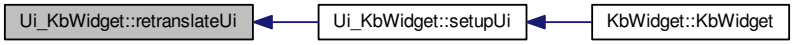
Definition at line 268 of file ui_kbwidget.h.

References Ui_KbWidget::bindTab, Ui_KbWidget::devTab, Ui_KbWidget::fwLabel, Ui_KbWidget::fwUpdButton, Ui_KbWidget::fwUpdLabel, Ui_KbWidget::hwSaveButton, Ui_KbWidget::kPerfTab, Ui_KbWidget::label_3, Ui_KbWidget::label_4, Ui_KbWidget::label_9, Ui_KbWidget::lightTab, Ui_KbWidget::mPerfTab, Ui_KbWidget::pollLabel, Ui_KbWidget::pollLabel2, Ui_KbWidget::serialLabel, and Ui_KbWidget::tabWidget.

Referenced by Ui_KbWidget::setupUi().

```
269     {
270         KbWidget->setWindowTitle(QApplication::translate("KbWidget", "Form", Q_NULLPTR));
271         label_3->setText(QApplication::translate("KbWidget", "Tip: Drag+drop items to reorder.
     Right-click for menu.", Q_NULLPTR));
272         tabWidget->setTabText(tabWidget->indexOf(lightTab),
    QApplication::translate("KbWidget", "Lighting", Q_NULLPTR));
273         tabWidget->setTabText(tabWidget->indexOf(bindTab), QApplication::translate
    ("KbWidget", "Binding", Q_NULLPTR));
274         tabWidget->setTabText(tabWidget->indexOf(kPerfTab),
    QApplication::translate("KbWidget", "Performance", Q_NULLPTR));
275         tabWidget->setTabText(tabWidget->indexOf(mPerfTab),
    QApplication::translate("KbWidget", "Performance", Q_NULLPTR));
276         fwUpdLabel->setText(QApplication::translate("KbWidget", "Status:", Q_NULLPTR));
277         fwLabel->setText(QApplication::translate("KbWidget", "N/A", Q_NULLPTR));
278         label_4->setText(QApplication::translate("KbWidget", "Serial Number:", Q_NULLPTR));
279         label_9->setText(QApplication::translate("KbWidget", "Firmware version:", Q_NULLPTR));
280         serialLabel->setText(QApplication::translate("KbWidget", "N/A", Q_NULLPTR));
281         fwUpdButton->setText(QApplication::translate("KbWidget", "Check for updates", Q_NULLPTR)
    );
282         pollLabel2->setText(QApplication::translate("KbWidget", "Poll rate:", Q_NULLPTR));
283         pollLabel->setText(QApplication::translate("KbWidget", "N/A", Q_NULLPTR));
284         tabWidget->setTabText(tabWidget->indexOf(devTab), QApplication::translate("
    KbWidget", "Device", Q_NULLPTR));
285         hwSaveButton->setText(QApplication::translate("KbWidget", "Save to hardware", Q_NULLPTR
    ));
286     } // retranslateUi
```

Here is the caller graph for this function:



12.39.2.2 void Ui_KbWidget::setupUi (QWidget * *KbWidget*) [inline], [inherited]

Definition at line 69 of file ui_kbwidget.h.

References Ui_KbWidget::bindTab, Ui_KbWidget::bindWidget, Ui_KbWidget::devTab, Ui_KbWidget::fwLabel, Ui_KbWidget::fwUpdButton, Ui_KbWidget::fwUpdLabel, Ui_KbWidget::fwUpdLayout, Ui_KbWidget::gridLayout, Ui_KbWidget::gridLayout_2, Ui_KbWidget::gridLayout_3, Ui_KbWidget::gridLayout_4, Ui_KbWidget::gridLayout_5, Ui_KbWidget::gridLayout_7, Ui_KbWidget::horizontalSpacer_2, Ui_KbWidget::horizontalSpacer_3, Ui_KbWidget::hwSaveButton, Ui_KbWidget::kPerfTab, Ui_KbWidget::kPerfWidget, Ui_KbWidget::label_3, Ui_KbWidget::label_4, Ui_KbWidget::label_9, Ui_KbWidget::lightTab, Ui_KbWidget::lightWidget, Ui_KbWidget::modesList, Ui_KbWidget::mPerfTab, Ui_KbWidget::mPerfWidget, Ui_KbWidget::pollLabel, Ui_KbWidget::pollLabel2, Ui_KbWidget::profileBox, Ui_KbWidget::retranslateUi(), Ui_KbWidget::serialLabel, Ui_KbWidget::tabWidget, and Ui_KbWidget::verticalSpacer.

Referenced by KbWidget::KbWidget().

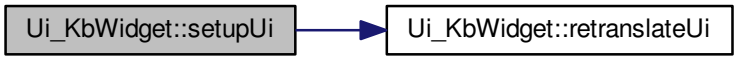
```
70     {
71         if (KbWidget->objectName().isEmpty())
72             KbWidget->setObjectName(QStringLiteral("KbWidget"));
73         KbWidget->resize(773, 539);
74         gridLayout = new QGridLayout(KbWidget);
75         gridLayout->setSpacing(9);
76         gridLayout->setObjectName(QStringLiteral("gridLayout"));
77         gridLayout->setContentsMargins(9, -1, 9, 9);
78         profileBox = new QComboBox(KbWidget);
79         profileBox->setObjectName(QStringLiteral("profileBox"));
80         QSizePolicy sizePolicy(QSizePolicy::Preferred, QSizePolicy::Fixed);
81         sizePolicy.setHorizontalStretch(18);
82         sizePolicy.setVerticalStretch(0);
83         sizePolicy.setHeightForWidth(profileBox->sizePolicy().hasHeightForWidth());
84         profileBox->setSizePolicy(sizePolicy);
85         profileBox->setIconSize(QSize(24, 24));
86
87         gridLayout->addWidget(profileBox, 0, 0, 1, 1);
88
```

```
89          tabWidget = new QTabWidget(KbWidget);
90          tabWidget->setObjectName(QStringLiteral("tabWidget"));
91          QSizePolicy sizePolicy1(QSizePolicy::Expanding, QSizePolicy::Expanding);
92          sizePolicy1.setHorizontalStretch(82);
93          sizePolicy1.setVerticalStretch(0);
94          sizePolicy1.setHeightForWidth(tabWidget->sizePolicy().hasHeightForWidth());
95          tabWidget->setSizePolicy(sizePolicy1);
96          lightTab = new QWidget();
97          lightTab->setObjectName(QStringLiteral("lightTab"));
98          gridLayout_2 = new QGridLayout(lightTab);
99          gridLayout_2->setObjectName(QStringLiteral("gridLayout_2"));
100          gridLayout_2->setHorizontalSpacing(0);
101          gridLayout_2->setVerticalSpacing(6);
102          gridLayout_2->setContentsMargins(6, 6, 6, 6);
103          lightWidget = new KbLightWidget(lightTab);
104          lightWidget->setObjectName(QStringLiteral("lightWidget"));
105          QSizePolicy sizePolicy2(QSizePolicy::Preferred, QSizePolicy::Expanding);
106          sizePolicy2.setHorizontalStretch(0);
107          sizePolicy2.setVerticalStretch(0);
108          sizePolicy2.setHeightForWidth(lightWidget->sizePolicy().hasHeightForWidth());
109          lightWidget->setSizePolicy(sizePolicy2);
110
111          gridLayout_2->addWidget(lightWidget, 0, 0, 1, 1);
112
113          label_3 = new QLabel(lightTab);
114          label_3->setObjectName(QStringLiteral("label_3"));
115          QSizePolicy sizePolicy3(QSizePolicy::Preferred, QSizePolicy::Minimum);
116          sizePolicy3.setHorizontalStretch(0);
117          sizePolicy3.setVerticalStretch(0);
118          sizePolicy3.setHeightForWidth(label_3->sizePolicy().hasHeightForWidth());
119          label_3->setSizePolicy(sizePolicy3);
120
121          gridLayout_2->addWidget(label_3, 1, 0, 1, 1);
122
123          tabWidget->addTab(lightTab, QString());
124          bindTab = new QWidget();
125          bindTab->setObjectName(QStringLiteral("bindTab"));
126          gridLayout_5 = new QGridLayout(bindTab);
127          gridLayout_5->setSpacing(0);
128          gridLayout_5->setObjectName(QStringLiteral("gridLayout_5"));
129          gridLayout_5->setContentsMargins(6, 6, 6, 6);
130          bindWidget = new KbBindWidget(bindTab);
131          bindWidget->setObjectName(QStringLiteral("bindWidget"));
132          sizePolicy2.setHeightForWidth(bindWidget->sizePolicy().hasHeightForWidth());
133          bindWidget->setSizePolicy(sizePolicy2);
134
135          gridLayout_5->addWidget(bindWidget, 0, 0, 1, 1);
136
137          tabWidget->addTab(bindTab, QString());
138          kPerfTab = new QWidget();
139          kPerfTab->setObjectName(QStringLiteral("kPerfTab"));
140          gridLayout_4 = new QGridLayout(kPerfTab);
141          gridLayout_4->setObjectName(QStringLiteral("gridLayout_4"));
142          gridLayout_4->setContentsMargins(6, 6, 6, 6);
143          kPerfWidget = new KPerfWidget(kPerfTab);
144          kPerfWidget->setObjectName(QStringLiteral("kPerfWidget"));
145
146          gridLayout_4->addWidget(kPerfWidget, 0, 0, 1, 1);
147
148          tabWidget->addTab(kPerfTab, QString());
149          mPerfTab = new QWidget();
150          mPerfTab->setObjectName(QStringLiteral("mPerfTab"));
151          gridLayout_7 = new QGridLayout(mPerfTab);
152          gridLayout_7->setObjectName(QStringLiteral("gridLayout_7"));
153          gridLayout_7->setContentsMargins(6, 6, 6, 6);
154          mPerfWidget = new MPerfWidget(mPerfTab);
155          mPerfWidget->setObjectName(QStringLiteral("mPerfWidget"));
156
157          gridLayout_7->addWidget(mPerfWidget, 0, 0, 1, 1);
158
159          tabWidget->addTab(mPerfTab, QString());
160          devTab = new QWidget();
161          devTab->setObjectName(QStringLiteral("devTab"));
162          gridLayout_3 = new QGridLayout(devTab);
163          gridLayout_3->setObjectName(QStringLiteral("gridLayout_3"));
164          gridLayout_3->setContentsMargins(6, 6, 6, 6);
165          fwUpdLabel = new QLabel(devTab);
166          fwUpdLabel->setObjectName(QStringLiteral("fwUpdLabel"));
167          fwUpdLabel->setMinimumSize(QSize(0, 34));
168
169          gridLayout_3->addWidget(fwUpdLabel, 4, 0, 1, 1);
170
171          verticalSpacer = new QSpacerItem(20, 40, QSizePolicy::Minimum, QSizePolicy::Expanding
    );
172
173          gridLayout_3->addItem(verticalSpacer, 6, 2, 1, 1);
174
```

```
175         fwLabel = new QLabel(devTab);
176         fwLabel->setObjectName(QStringLiteral("fwLabel"));
177         fwLabel->setCursor(QCursor(Qt::IBeamCursor));
178         fwLabel->setTextInteractionFlags(Qt::LinksAccessibleByMouse|Qt::TextSelectableByMouse);
179 
180         gridLayout_3->addWidget(fwLabel, 2, 2, 1, 1);
181 
182         label_4 = new QLabel(devTab);
183         label_4->setObjectName(QStringLiteral("label_4"));
184         label_4->setMinimumSize(QSize(0, 34));
185 
186         gridLayout_3->addWidget(label_4, 0, 0, 1, 1);
187 
188         horizontalSpacer_2 = new QSpacerItem(40, 1, QSizePolicy::Fixed,
    QSizePolicy::Minimum);
189 
190         gridLayout_3->addItem(horizontalSpacer_2, 2, 1, 1, 1);
191 
192         label_9 = new QLabel(devTab);
193         label_9->setObjectName(QStringLiteral("label_9"));
194         label_9->setMinimumSize(QSize(0, 34));
195 
196         gridLayout_3->addWidget(label_9, 2, 0, 1, 1);
197 
198         serialLabel = new QLabel(devTab);
199         serialLabel->setObjectName(QStringLiteral("serialLabel"));
200         serialLabel->setCursor(QCursor(Qt::IBeamCursor));
201         serialLabel->setTextInteractionFlags(Qt::LinksAccessibleByMouse|
    Qt::TextSelectableByMouse);
202 
203         gridLayout_3->addWidget(serialLabel, 0, 2, 1, 1);
204 
205         fwUpdLayout = new QHBoxLayout();
206         fwUpdLayout->setObjectName(QStringLiteral("fwUpdLayout"));
207         fwUpdButton = new QPushButton(devTab);
208         fwUpdButton->setObjectName(QStringLiteral("fwUpdButton"));
209 
210         fwUpdLayout->addWidget(fwUpdButton);
211 
212         horizontalSpacer_3 = new QSpacerItem(40, 20, QSizePolicy::Expanding,
    QSizePolicy::Minimum);
213 
214         fwUpdLayout->addItem(horizontalSpacer_3);
215 
216 
217         gridLayout_3->addLayout(fwUpdLayout, 4, 2, 1, 1);
218 
219         pollLabel2 = new QLabel(devTab);
220         pollLabel2->setObjectName(QStringLiteral("pollLabel2"));
221         pollLabel2->setMinimumSize(QSize(0, 34));
222 
223         gridLayout_3->addWidget(pollLabel2, 1, 0, 1, 1);
224 
225         pollLabel = new QLabel(devTab);
226         pollLabel->setObjectName(QStringLiteral("pollLabel"));
227 
228         gridLayout_3->addWidget(pollLabel, 1, 2, 1, 1);
229 
230         tabWidget->addTab(devTab, QString());
231 
232         gridLayout->addWidget(tabWidget, 0, 1, 3, 1);
233 
234         modesList = new RListWidget(KbWidget);
235         modesList->setObjectName(QStringLiteral("modesList"));
236         QSizePolicy sizePolicy4(QSizePolicy::Expanding, QSizePolicy::Expanding);
237         sizePolicy4.setHorizontalStretch(18);
238         sizePolicy4.setVerticalStretch(0);
239         sizePolicy4.setHeightForWidth(modesList->sizePolicy().hasHeightForWidth());
240         modesList->setSizePolicy(sizePolicy4);
241         modesList->setContextMenuPolicy(Qt::CustomContextMenu);
242         modesList->setFrameShadow(QFrame::Plain);
243         modesList->setDragDropMode(QAbstractItemView::InternalMove);
244         modesList->setIconSize(QSize(24, 24));
245         modesList->setMovement(QListView::Snap);
246 
247         gridLayout->addWidget(modesList, 1, 0, 1, 1);
248 
249         hwSaveButton = new QPushButton(KbWidget);
250         hwSaveButton->setObjectName(QStringLiteral("hwSaveButton"));
251         QSizePolicy sizePolicy5(QSizePolicy::Minimum, QSizePolicy::Fixed);
252         sizePolicy5.setHorizontalStretch(18);
253         sizePolicy5.setVerticalStretch(0);
254         sizePolicy5.setHeightForWidth(hwSaveButton->sizePolicy().hasHeightForWidth());
255         hwSaveButton->setSizePolicy(sizePolicy5);
256 
257         gridLayout->addWidget(hwSaveButton, 2, 0, 1, 1);
258 
```

```
259
260         retranslateUi(KbWidget);
261
262         tabWidget->setCurrentIndex(0);
263
264
265         QMetaObject::connectSlotsByName(KbWidget);
266     } // setupUi
```

Here is the call graph for this function:



Here is the caller graph for this function:

12.39.3 Field Documentation

12.39.3.1 QWidget* Ui_KbWidget::bindTab [inherited]

Definition at line 43 of file ui_kbwidget.h.

Referenced by Ui_KbWidget::retranslateUi(), and Ui_KbWidget::setupUi().

12.39.3.2 KbBindWidget* Ui_KbWidget::bindWidget [inherited]

Definition at line 45 of file ui_kbwidget.h.

Referenced by KbWidget::modeChanged(), and Ui_KbWidget::setupUi().

12.39.3.3 QWidget* Ui_KbWidget::devTab [inherited]

Definition at line 52 of file ui_kbwidget.h.

Referenced by Ui_KbWidget::retranslateUi(), and Ui_KbWidget::setupUi().

12.39.3.4 QLabel* Ui_KbWidget::fwLabel [inherited]

Definition at line 56 of file ui_kbwidget.h.

Referenced by KbWidget::devUpdate(), Ui_KbWidget::retranslateUi(), and Ui_KbWidget::setupUi().

12.39.3.5 QPushButton* Ui_KbWidget::fwUpdButton [inherited]

Definition at line 62 of file ui_kbwidget.h.

Referenced by KbWidget::KbWidget(), KbWidget::on_fwUpdButton_clicked(), Ui_KbWidget::retranslateUi(), Ui_KbWidget::setupUi(), and KbWidget::updateFwButton().

12.39.3.6 QLabel* Ui_KbWidget::fwUpdLabel [inherited]

Definition at line 54 of file ui_kbwidget.h.

Referenced by KbWidget::KbWidget(), Ui_KbWidget::retranslateUi(), and Ui_KbWidget::setupUi().

12.39.3.7 QHBoxLayout* Ui_KbWidget::fwUpdLayout [inherited]

Definition at line 61 of file ui_kbwidget.h.

Referenced by KbWidget::KbWidget(), and Ui_KbWidget::setupUi().

12.39.3.8 QGridLayout* Ui_KbWidget::gridLayout [inherited]

Definition at line 36 of file ui_kbwidget.h.

Referenced by Ui_KbWidget::setupUi().

12.39.3.9 QGridLayout* Ui_KbWidget::gridLayout_2 [inherited]

Definition at line 40 of file ui_kbwidget.h.

Referenced by Ui_KbWidget::setupUi().

12.39.3.10 QGridLayout* Ui_KbWidget::gridLayout_3 [inherited]

Definition at line 53 of file ui_kbwidget.h.

Referenced by Ui_KbWidget::setupUi().

12.39.3.11 QGridLayout* Ui_KbWidget::gridLayout_4 [inherited]

Definition at line 47 of file ui_kbwidget.h.

Referenced by Ui_KbWidget::setupUi().

12.39.3.12 QGridLayout* Ui_KbWidget::gridLayout_5 [inherited]

Definition at line 44 of file ui_kbwidget.h.

Referenced by Ui_KbWidget::setupUi().

12.39.3.13 QGridLayout* Ui_KbWidget::gridLayout_7 [inherited]

Definition at line 50 of file ui_kbwidget.h.

Referenced by Ui_KbWidget::setupUi().

12.39.3.14 QSpacerItem* Ui_KbWidget::horizontalSpacer_2 [inherited]

Definition at line 58 of file ui_kbwidget.h.

Referenced by Ui_KbWidget::setupUi().

12.39.3.15 QSpacerItem* Ui_KbWidget::horizontalSpacer_3 [inherited]

Definition at line 63 of file ui_kbwidget.h.

Referenced by Ui_KbWidget::setupUi().

12.39.3.16 QPushButton* Ui_KbWidget::hwSaveButton [inherited]

Definition at line 67 of file ui_kbwidget.h.

Referenced by Ui_KbWidget::retranslateUi(), and Ui_KbWidget::setupUi().

12.39.3.17 QWidget* Ui_KbWidget::kPerfTab [inherited]

Definition at line 46 of file ui_kbwidget.h.

Referenced by KbWidget::KbWidget(), Ui_KbWidget::retranslateUi(), and Ui_KbWidget::setupUi().

12.39.3.18 KPerfWidget* Ui_KbWidget::kPerfWidget [inherited]

Definition at line 48 of file ui_kbwidget.h.

Referenced by KbWidget::modeChanged(), and Ui_KbWidget::setupUi().

12.39.3.19 QLabel* Ui_KbWidget::label_3 [inherited]

Definition at line 42 of file ui_kbwidget.h.

Referenced by Ui_KbWidget::retranslateUi(), and Ui_KbWidget::setupUi().

12.39.3.20 QLabel* Ui_KbWidget::label_4 [inherited]

Definition at line 57 of file ui_kbwidget.h.

Referenced by Ui_KbWidget::retranslateUi(), and Ui_KbWidget::setupUi().

12.39.3.21 QLabel* Ui_KbWidget::label_9 [inherited]

Definition at line 59 of file ui_kbwidget.h.

Referenced by Ui_KbWidget::retranslateUi(), and Ui_KbWidget::setupUi().

12.39.3.22 QWidget* Ui_KbWidget::lightTab [inherited]

Definition at line 39 of file ui_kbwidget.h.

Referenced by KbWidget::KbWidget(), Ui_KbWidget::retranslateUi(), and Ui_KbWidget::setupUi().

12.39.3.23 KbLightWidget* Ui_KbWidget::lightWidget [inherited]

Definition at line 41 of file ui_kbwidget.h.

Referenced by KbWidget::KbWidget(), KbWidget::modeChanged(), and Ui_KbWidget::setupUi().

12.39.3.24 RListWidget* Ui_KbWidget::modesList [inherited]

Definition at line 66 of file ui_kbwidget.h.

Referenced by KbWidget::addNewModeItem(), KbWidget::KbWidget(), KbWidget::modeChanged(), KbWidget::modesList_reordered(), KbWidget::on_modesList_customContextMenuRequested(), KbWidget::on_modesList_itemClicked(), KbWidget::profileChanged(), and Ui_KbWidget::setupUi().

12.39.3.25 QWidget* Ui_KbWidget::mPerfTab [inherited]

Definition at line 49 of file ui_kbwidget.h.

Referenced by KbWidget::KbWidget(), Ui_KbWidget::retranslateUi(), and Ui_KbWidget::setupUi().

12.39.3.26 MPerfWidget* Ui_KbWidget::mPerfWidget [inherited]

Definition at line 51 of file ui_kbwidget.h.

Referenced by KbWidget::modeChanged(), and Ui_KbWidget::setupUi().

12.39.3.27 QLabel* Ui_KbWidget::pollLabel [inherited]

Definition at line 65 of file ui_kbwidget.h.

Referenced by KbWidget::devUpdate(), KbWidget::KbWidget(), Ui_KbWidget::retranslateUi(), and Ui_KbWidget::setupUi().

12.39.3.28 QLabel* Ui_KbWidget::pollLabel2 [inherited]

Definition at line 64 of file ui_kbwidget.h.

Referenced by KbWidget::KbWidget(), Ui_KbWidget::retranslateUi(), and Ui_KbWidget::setupUi().

12.39.3.29 QComboBox* Ui_KbWidget::profileBox [inherited]

Definition at line 37 of file ui_kbwidget.h.

Referenced by Ui_KbWidget::setupUi(), and KbWidget::updateProfileList().

12.39.3.30 QLabel* Ui_KbWidget::serialLabel [inherited]

Definition at line 60 of file ui_kbwidget.h.

Referenced by KbWidget::devUpdate(), Ui_KbWidget::retranslateUi(), and Ui_KbWidget::setupUi().

12.39.3.31 QTabWidget* Ui_KbWidget::tabWidget [inherited]

Definition at line 38 of file ui_kbwidget.h.

Referenced by KbWidget::KbWidget(), KbWidget::on_tabWidget_currentChanged(), Ui_KbWidget::retranslateUi(), Ui_KbWidget::setupUi(), KbWidget::showFirstTab(), and KbWidget::showLastTab().

12.39.3.32 QSpacerItem* Ui_KbWidget::verticalSpacer [inherited]

Definition at line 55 of file ui_kbwidget.h.

Referenced by Ui_KbWidget::setupUi().

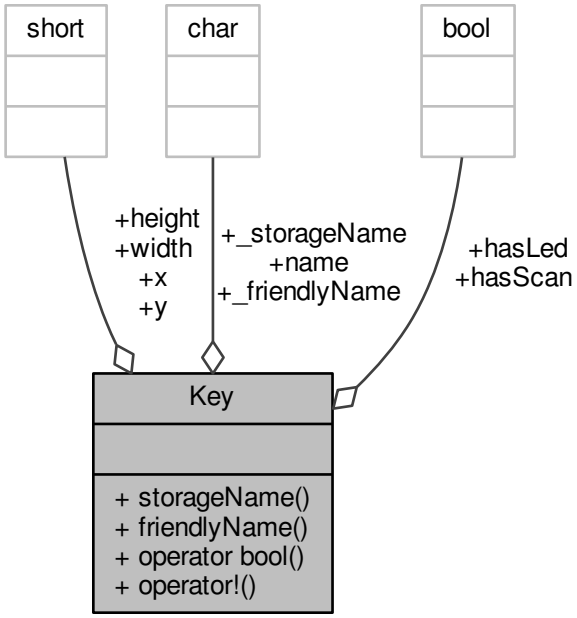
The documentation for this class was generated from the following file:

- src/ckb/ui_kbwidget.h

12.40 Key Struct Reference

`#include <src/ckb/keymap.h>`

Collaboration diagram for Key:



Public Member Functions

- const char * storageName () const
- QString friendlyName (bool os=true) const
- operator bool () const
- bool operator! () const

Data Fields

- const char * _storageName
- const char * _friendlyName
- const char * name

- short x
- short y
- short width
- short height
- bool hasLed
- bool hasScan

12.40.1 Detailed Description

Definition at line 8 of file keymap.h.

12.40.2 Member Function Documentation

12.40.2.1 QString Key::friendlyName (bool *os* = `true`) const [inline]

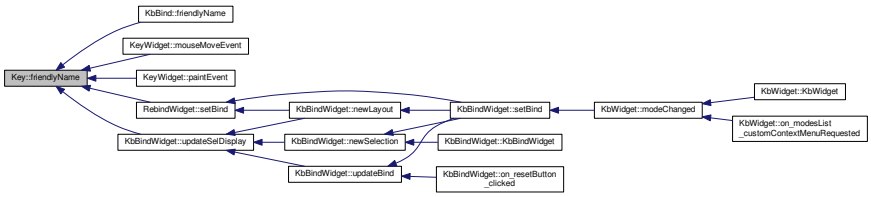
Definition at line 23 of file keymap.h.

References _friendlyName, and name.

Referenced by KbBind::friendlyName(), KeyWidget::mouseMoveEvent(), KeyWidget::paintEvent(), RebindWidget::setBind(), and KbBindWidget::updateSelDisplay().

```
23                                                      {
24         if(os){
25 #ifdef Q_OS_MACX
26             if(!strcmp(name, "lwin")) return "Left Cmd";
27             if(!strcmp(name, "rwin")) return "Right Cmd";
28             if(!strcmp(name, "lalt")) return "Left Option";
29             if(!strcmp(name, "ralt")) return "Right Option";
30             if(!strcmp(name, "prtscn")) return "F13";
31             if(!strcmp(name, "scroll")) return "F14";
32             if(!strcmp(name, "pause")) return "F15";
33             if(!strcmp(name, "ins")) return "Help";
34             if(!strcmp(name, "numlock")) return "Clear";
35 #elif defined(Q_OS_LINUX)
36             if(!strcmp(name, "lwin")) return "Left Super";
37             if(!strcmp(name, "rwin")) return "Right Super";
38 #endif
39         }
40         return _friendlyName ? _friendlyName : QString(
    name).toUpper();
41     }
```

Here is the caller graph for this function:



12.40.2.2 Key::operator bool ( ) const [inline]

Definition at line 44 of file keymap.h.

References name.

```
44 { return name != 0; }
```

12.40.2.3 bool Key::operator! ( ) const [inline]

Definition at line 45 of file keymap.h.

```
45 { return !(bool)*this; }
```

12.40.2.4 const char* Key::storageName ( ) const [inline]

Definition at line 11 of file keymap.h.

References _storageName, and name.

Referenced by KeyMap::toStorage().

```
11 { return _storageName ? _storageName : name; }
```

Here is the caller graph for this function:

12.40.3 Field Documentation

12.40.3.1 const char* Key::_friendlyName

Definition at line 13 of file keymap.h.

Referenced by _patch(), and friendlyName().

12.40.3.2 const char* Key::_storageName

Definition at line 10 of file keymap.h.

Referenced by _patch(), and storageName().

12.40.3.3 bool Key::hasLed

Definition at line 19 of file keymap.h.

Referenced by ColorMap::init(), KeyWidget::mouseMoveEvent(), KeyWidget::mousePressEvent(), KeyWidget::paintEvent(), and KeyWidget::selectAll().

12.40.3.4 bool Key::hasScan

Definition at line 20 of file keymap.h.

Referenced by KeyWidget::mouseMoveEvent(), KeyWidget::mousePressEvent(), KeyWidget::paintEvent(), and KeyWidget::selectAll().

12.40.3.5 short Key::height

Definition at line 17 of file keymap.h.

Referenced by getMap(), KeyWidget::mouseMoveEvent(), KeyWidget::mousePressEvent(), KeyWidget::paintEvent(), and patchANSI().

12.40.3.6 const char* Key::name

Definition at line 14 of file keymap.h.

Referenced by friendlyName(), ColorMap::init(), KeyWidget::mouseMoveEvent(), KeyWidget::mousePressEvent(), operator bool(), KeyWidget::paintEvent(), KeyWidget::selectAll(), and storageName().

12.40.3.7 short Key::width

Definition at line 17 of file keymap.h.

Referenced by getMap(), KeyWidget::mouseMoveEvent(), KeyWidget::mousePressEvent(), KeyWidget::paintEvent(), patchABNT2(), patchANSI(), patchJP106(), and patchJP106fn().

12.40.3.8 short Key::x

Definition at line 16 of file keymap.h.

Referenced by AnimScript::begin(), KeyMap::byPosition(), getMap(), AnimScript::keypress(), KeyWidget::mouseMoveEvent(), KeyWidget::mousePressEvent(), KeyWidget::paintEvent(), patchABNT2(), patchANSI(), patchJP106(), and patchJP106fn().

12.40.3.9 short Key::y

Definition at line 16 of file keymap.h.

Referenced by AnimScript::begin(), KeyMap::byPosition(), getMap(), AnimScript::keypress(), KeyWidget::mouseMoveEvent(), KeyWidget::mousePressEvent(), KeyWidget::paintEvent(), and patchANSI().

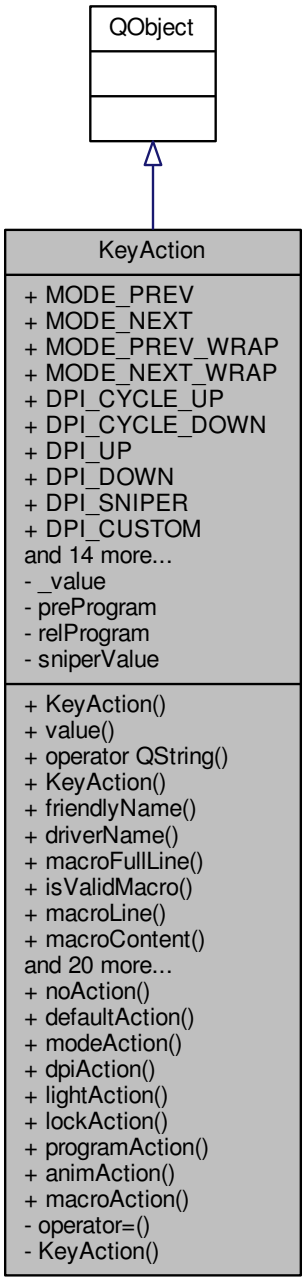
The documentation for this struct was generated from the following file:

- src/ckb/keymap.h

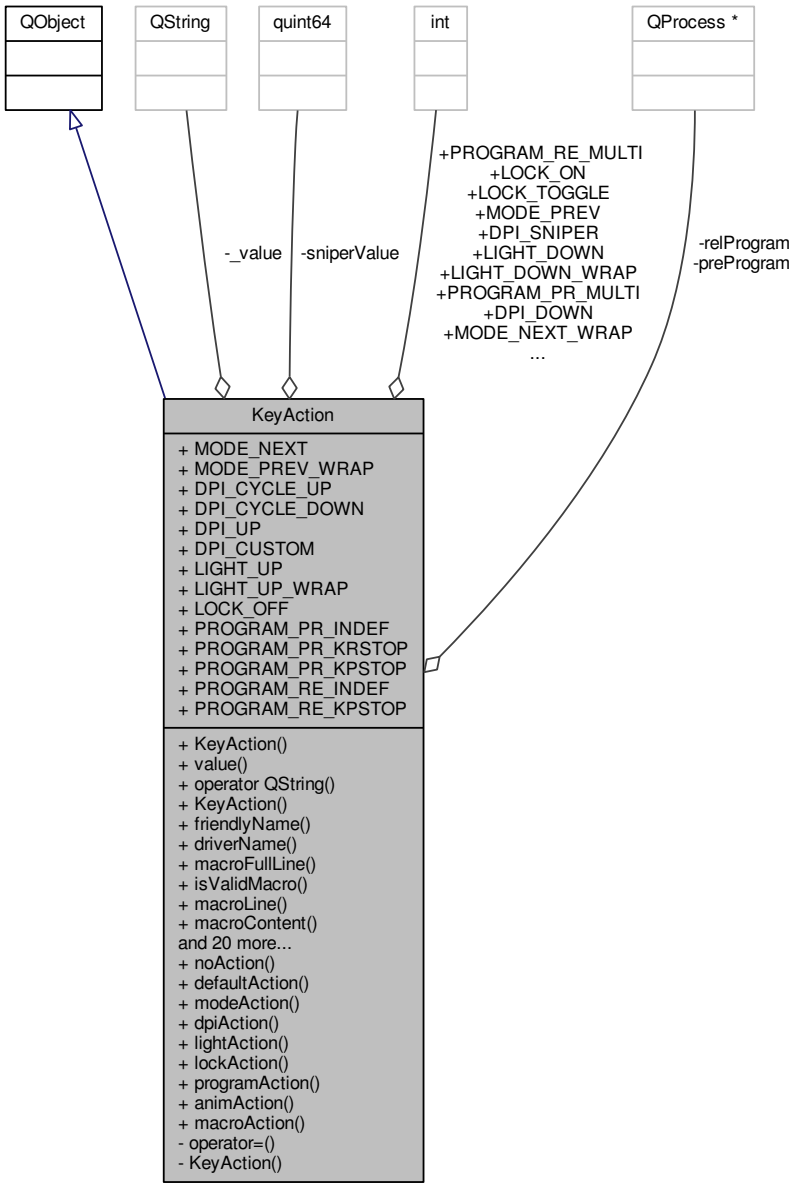
12.41 KeyAction Class Reference

`#include <src/ckb/keyaction.h>`

Inheritance diagram for KeyAction:



Collaboration diagram for KeyAction:



Public Types

- enum Type { UNBOUND, NORMAL, SPECIAL }

Public Member Functions

- KeyAction (const QString &action, QObject *parent=0)
- QString value () const
- operator QString () const
- KeyAction (QObject *parent=0)
- QString friendlyName (const KeyMap &map) const

- QString driverName () const
- QString macroFullLine () const

  macroFullLine If a macro command and a macro definition exists for the given key, returns the complete string except the leading "$" (the $ may confuse some caller).
- bool isValidMacro () const

  isValidMacro checks whether a keyAction contains a valid macro. This is done easily: If the macro action starts with $macro: and has five elements, delimited by ":", we may assume, that is a structural correct macro action. If it has 4 entries only, it is an older definition and ok also.
- QStringList macroLine () const

  macroLine returns all interresting content for a macro definition.
- QString macroContent () const

  macroContent returns the macro key definition only (the second part of the macro action).
- QString macroTiming () const

  macroTiming returns the macro key definition with original timing infos (the fifth and up to now last part of the macro action). If the implementation does not know anything about delays and has no 5th part, return first part.
- void macroDisplay ()

  Debug output for invalid macro Definitions.
- Type type () const
- bool isUnbound () const
- bool isNormal () const
- bool isSpecial () const
- bool isMedia () const
- bool isProgram () const
- bool isAnim () const
- bool isMacro () const
- bool isDPI () const
- bool isMouse () const
- QString specialInfo (int &parameter) const
- int programInfo (QString &onPress, QString &onRelease) const
- int dpiInfo (QPoint &custom) const
- QUuid animInfo (bool &onlyOnce, bool &stopOnRelease) const
- void keyEvent (KbBind *bind, bool down)
- void keyRelease (KbBind *bind)
- void adjustDisplay ()
- ~KeyAction ()

Static Public Member Functions

- static QString noAction ()
- static QString defaultAction (const QString &key, KeyMap::Model model)
- static QString modeAction (int mode)
- static QString dpiAction (int level, int customX=0, int customY=0)
- static QString lightAction (int type=LIGHT_UP_WRAP)
- static QString lockAction (int type=LOCK_TOGGLE)
- static QString programAction (const QString &onPress, const QString &onRelease, int stop)
- static QString animAction (const QUuid &guid, bool onlyOnce, bool stopOnRelease)
- static QString macroAction (QString macroDef)

  well documented in cpp file

Static Public Attributes

- static const int MODE_PREV = -2
- static const int MODE_NEXT = -1
- static const int MODE_PREV_WRAP = -4
- static const int MODE_NEXT_WRAP = -3
- static const int DPI_CYCLE_UP = -4
- static const int DPI_CYCLE_DOWN = -3
- static const int DPI_UP = -2
- static const int DPI_DOWN = -1
- static const int DPI_SNIPER = 0
- static const int DPI_CUSTOM = 6
- static const int LIGHT_UP = 0
- static const int LIGHT_DOWN = 1
- static const int LIGHT_UP_WRAP = 2
- static const int LIGHT_DOWN_WRAP = 3
- static const int LOCK_TOGGLE = 0
- static const int LOCK_ON = 1
- static const int LOCK_OFF = 2
- static const int PROGRAM_PR_MULTI = 0x04
- static const int PROGRAM_PR_INDEF = 0x00
- static const int PROGRAM_PR_KRSTOP = 0x01
- static const int PROGRAM_PR_KPSTOP = 0x02
- static const int PROGRAM_RE_MULTI = 0x40
- static const int PROGRAM_RE_INDEF = 0x00
- static const int PROGRAM_RE_KPSTOP = 0x20

Private Member Functions

- void operator= (const KeyAction &rhs)

  ccMSC: Don't copy key actions (the old one needs to be deleted first) frickler24: statement left as described, but copying is done in KbBind copy constructor
- KeyAction (const KeyAction &rhs)

Private Attributes

- QString _value
- QProcess * preProgram
- QProcess * relProgram
- quint64 sniperValue

12.41.1 Detailed Description

Definition at line 13 of file keyaction.h.

12.41.2 Member Enumeration Documentation

12.41.2.1 enum KeyAction::Type

Enumerator

UNBOUND

NORMAL

SPECIAL

Definition at line 146 of file keyaction.h.

```
146                 {
147         UNBOUND,
148         NORMAL,
149         SPECIAL,
150     };
```

12.41.3 Constructor & Destructor Documentation

12.41.3.1 KeyAction::KeyAction (const QString & *action,* QObject * *parent* = 0)

Definition at line 22 of file keyaction.cpp.

```
23     : QObject(parent), _value(action), preProgram(0),
      relProgram(0), sniperValue(0)
24 {
25 }
```

12.41.3.2 KeyAction::KeyAction (QObject * *parent* = 0) [explicit]

Definition at line 27 of file keyaction.cpp.

```
28     : QObject(parent), _value(""), preProgram(0),
      relProgram(0), sniperValue(0)
29 {
30 }
```

12.41.3.3 KeyAction::~KeyAction ( )

Definition at line 32 of file keyaction.cpp.

References preProgram, and relProgram.

```
32                      {
33     // Clean up processes
34     if(preProgram){
35         preProgram->kill();
36         delete preProgram;
37     }
38     if(relProgram){
39         relProgram->kill();
40         delete relProgram;
41     }
42 }
```

12.41.3.4 KeyAction::KeyAction (const KeyAction & *rhs*) [inline], [private]

Definition at line 187 of file keyaction.h.

```
187 : QObject() {}
```

12.41.4 Member Function Documentation

12.41.4.1 void KeyAction::adjustDisplay ( )

Definition at line 464 of file keyaction.cpp.

Referenced by keyEvent().

```
464                             {
465 #ifdef USE_LIBX11
466     // Try to get the current display from the X server
467     char* display_name = XDisplayName(NULL);
468     if(!display_name)
469         return;
470     Display* display = XOpenDisplay(display_name);
471     if(!display)
472         return;
473     char* display_string = DisplayString(display);
474     if(!display_string || strlen(display_string) == 0){
475         XCloseDisplay(display);
476         return;
477     }
478     size_t envstr_size = strlen(display_string) + 4;
479     char* envstr = new char[envstr_size];
480     strncpy(envstr, display_string, envstr_size);
481     envstr[envstr_size - 1] = 0;
482 
483     Window root_window = XRootWindow(display, DefaultScreen(display));
484     Window root_window_ret, child_window_ret, window;
485     XWindowAttributes attr;
486     int root_x, root_y, win_x, win_y;
487     unsigned int mask_ret;
488 
489     // Find the screen which currently has the mouse
490     XQueryPointer(display, root_window, &root_window_ret, &child_window_ret, &root_x, &root_y, &win_x, &
    win_y, &mask_ret);
491     if(child_window_ret == (Window)NULL)
492         window = root_window_ret;
493     else
494         window = child_window_ret;
495     XGetWindowAttributes(display, window,  &attr);
496 
497     char* ptr = strchr(envstr, ':');
498     if(ptr){
499         ptr = strchr(ptr, '.');
500         if(ptr)
501             *ptr = '\0';
502         char buf[16];
503         snprintf(buf, sizeof(buf), ".%i", XScreenNumberOfScreen(attr.screen));
504         strncat(envstr, buf, envstr_size - 1 - strlen(envstr));
505 
506         // Update environment variable
507         setenv("DISPLAY", envstr, 1);
508     }
509 
510     delete[] envstr;
511     XCloseDisplay(display);
512 #endif
513 }
```

Here is the caller graph for this function:

12.41.4.2 QString KeyAction::animAction (const QUuid & *guid,* bool *onlyOnce,* bool *stopOnRelease*) [static]

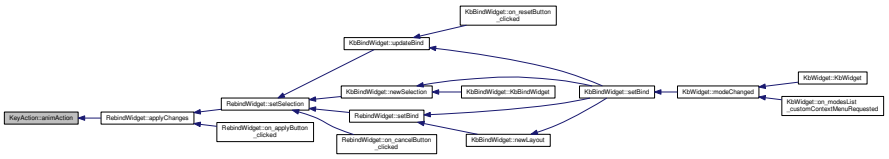
Definition at line 184 of file keyaction.cpp.

References ANIM_KRSTOP, and ANIM_ONCE.

Referenced by RebindWidget::applyChanges().

```
184                                                                                 {
185     int flags = (onlyOnce ? ANIM_ONCE : 0) | (stopOnRelease ?
    ANIM_KRSTOP : 0);
186     return "$anim:" + guid.toString() + QString("+%1").arg(flags);
187 }
```

Here is the caller graph for this function:



12.41.4.3 QUuid KeyAction::animInfo (bool & *onlyOnce,* bool & *stopOnRelease*) const

Definition at line 225 of file keyaction.cpp.

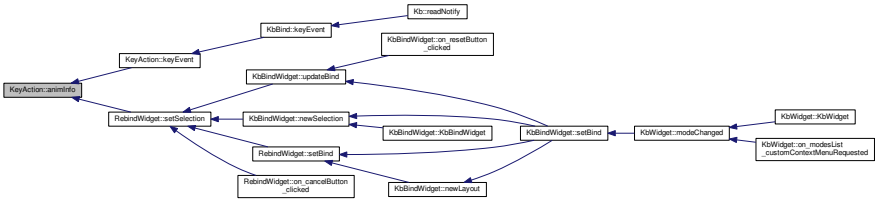
References _value, ANIM_KRSTOP, ANIM_ONCE, and isAnim().

Referenced by keyEvent(), and RebindWidget::setSelection().

```
225                                                                 {
226     if(!isAnim())
227         return QUuid();
228     QString param = _value.mid(6);
229     QStringList split = param.split("+");
230     if(split.length() < 2)
231         return QUuid();
232     QUuid id = split[0];
233     int flags = split[1].toInt();
234     onlyOnce = !!(flags & ANIM_ONCE);
235     stopOnRelease = !!(flags & ANIM_KRSTOP);
236     return id;
237 }
```

Here is the call graph for this function:

Here is the caller graph for this function:



12.41.4.4 QString KeyAction::defaultAction (const QString & *key,* KeyMap::Model *model*) [static]

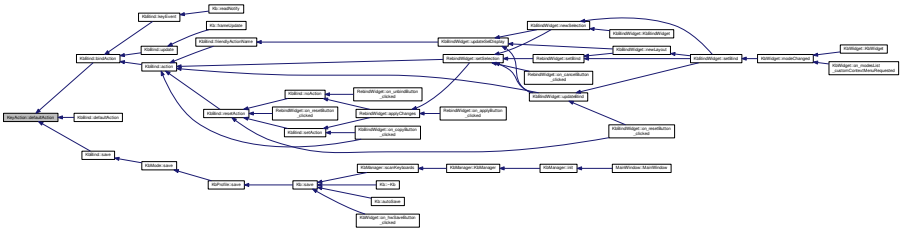
Definition at line 44 of file keyaction.cpp.

References KeyMap::GLAIVE, and KeyMap::HARPOON.

Referenced by KbBind::bindAction(), KbBind::defaultAction(), and KbBind::save().

```
44                                                                      {
45     // G1-G18 are unbound by default
46     if(key.length() >= 2 && key[0] == 'g'
47         && ((key.length() == 2 && key[1] >= '0' && key[1] <= '9')
48         || (key.length() == 3 && key[1] == '1' && key[2] >= '0' && key[2] <= '8')))
49         return "";
50     // So are thumbgrid buttons
51     if(key.startsWith("thumb"))
52         return "";
53     // TODO: default action for MR
54     if(key == "mr")
55         return "";
56     // M1-M3 switch modes
57     if(key == "m1")
58         return "$mode:0";
59     if(key == "m2")
60         return "$mode:1";
61     if(key == "m3")
62         return "$mode:2";
63     // Brightness and Win Lock are their own functions
64     if(key == "light")
65         return "$light:2";
66     if(key == "lock")
67         return "$lock:0";
68     // DPI buttons
69     if(key == "dpiup"){
70         if(model == KeyMap::HARPOON ||
71        model == KeyMap::GLAIVE){
72         return "$dpi:-4";
73         }
74     return "$dpi:-2";
75     }
76     if(key == "dpidn")
77         return "$dpi:-1";
78     if(key == "sniper")
79         return "$dpi:0";
80     // Everything else is a standard keypress
81     return key;
82 }
```

Here is the caller graph for this function:



12.41.4.5 QString KeyAction::dpiAction (int *level*, int *customX* = 0, int *customY* = 0) [static]

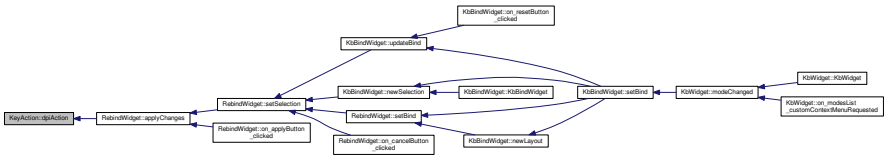
Definition at line 162 of file keyaction.cpp.

References DPI_CUSTOM.

Referenced by RebindWidget::applyChanges().

```
162                                                                     {
163     QString action = tr("$dpi:%1").arg(level);
164     if(level == DPI_CUSTOM)
165         action += tr("+%1+%2").arg(customX).arg(customY);
166     return action;
167 }
```

Here is the caller graph for this function:



12.41.4.6 int KeyAction::dpiInfo (QPoint & *custom*) const

Definition at line 211 of file keyaction.cpp.

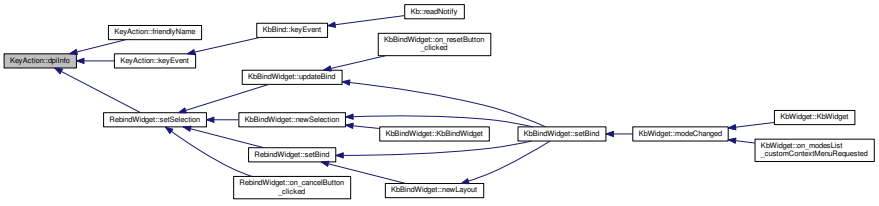
References _value, DPI_CUSTOM, and isDPI().

Referenced by friendlyName(), keyEvent(), and RebindWidget::setSelection().

```
211                                         {
212     if(!isDPI())
213         return 0;
214     QString param = _value.mid(5);
215     QStringList lxy = param.split("+");
216     int level = lxy[0].toInt();
217     if(level == DPI_CUSTOM){
218         if(lxy.length() != 3)
219             return 0;
220         custom = QPoint(lxy[1].toInt(), lxy[2].toInt());
221     }
222     return level;
223 }
```

Here is the call graph for this function:

Here is the caller graph for this function:



12.41.4.7 QString KeyAction::driverName ( ) const

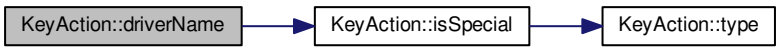
Definition at line 239 of file keyaction.cpp.

References _value, and isSpecial().

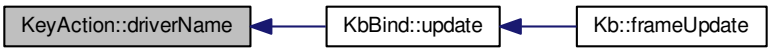
Referenced by KbBind::update().

```
239                                     {
240     if(isSpecial())
241         return "";
242     return _value;
243 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.41.4.8 QString KeyAction::friendlyName (const KeyMap & *map*) const

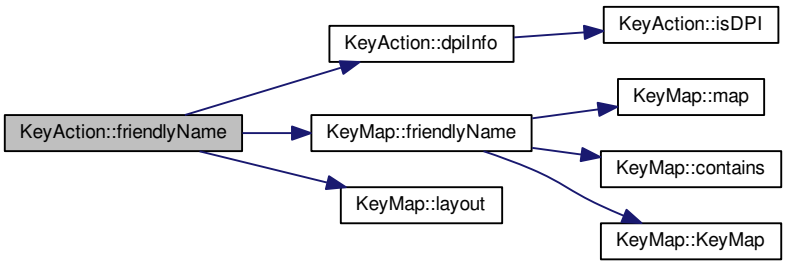
Definition at line 84 of file keyaction.cpp.

References _value, DPI_CUSTOM, DPI_CYCLE_DOWN, DPI_CYCLE_UP, DPI_DOWN, DPI_SNIPER, DPI_UP, dpiInfo(), KeyMap::friendlyName(), KeyMap::layout(), LIGHT_DOWN, LIGHT_DOWN_WRAP, LIGHT_UP, LIGHT-_UP_WRAP, LOCK_OFF, LOCK_ON, LOCK_TOGGLE, MODE_NEXT, MODE_NEXT_WRAP, MODE_PREV, and MODE_PREV_WRAP.

```
84                                                      {
85     if(_value.isEmpty())
86         return "Unbound";
87     QStringList parts = _value.split(":");
88     QString prefix = parts[0];
89     if(parts.length() < 2){
90         KeyMap::Layout layout = map.layout();
91         QString name = KeyMap::friendlyName(_value, layout);
92         if(name.isEmpty())
93             return "(Unknown)";
94         return name;
95     }
96     int suffix = parts[1].toInt();
97     if(prefix == "$mode"){
98         switch(suffix){
99         case MODE_PREV:
100         case MODE_PREV_WRAP:
101             return "Switch to previous mode";
102         case MODE_NEXT:
103         case MODE_NEXT_WRAP:
104             return "Switch to next mode";
105         default:
106             return tr("Switch to mode %1").arg(suffix + 1);
107         }
108     } else if(prefix == "$dpi"){
109         // Split off custom parameters (if any)
110         int level = parts[1].split("+")[0].toInt();
```

```
111         switch(level){
112         case DPI_CYCLE_UP:
113             return "DPI cycle up";
114         case DPI_CYCLE_DOWN:
115             return "DPI cycle down";
116         case DPI_UP:
117             return "DPI up";
118         case DPI_DOWN:
119             return "DPI down";
120         case DPI_SNIPER:
121             return "Sniper";
122         case DPI_CUSTOM:{
123             QPoint xy;
124             dpiInfo(xy);
125             return tr("DPI: %1, %2").arg(xy.x()).arg(xy.y());
126         }
127         default:
128             return tr("DPI stage %1").arg(level);
129         }
130     } else if(prefix == "$light"){
131         switch(suffix){
132         case LIGHT_UP:
133         case LIGHT_UP_WRAP:
134             return "Brightness up";
135         case LIGHT_DOWN:
136         case LIGHT_DOWN_WRAP:
137             return "Brightness down";
138         }
139     } else if(prefix == "$lock"){
140         switch(suffix){
141         case LOCK_TOGGLE:
142             return "Toggle Windows lock";
143         case LOCK_ON:
144             return "Windows lock on";
145         case LOCK_OFF:
146             return "Windows lock off";
147         }
148     } else if(prefix == "$anim"){
149         return "Start animation";
150     } else if(prefix == "$program"){
151         return "Launch program";
152     } else if(prefix == "$macro"){
153         return "Send G-key macro";
154     }
155     return "(Unknown)";
156 }
```

Here is the call graph for this function:



12.41.4.9 bool KeyAction::isAnim ( ) const [inline]

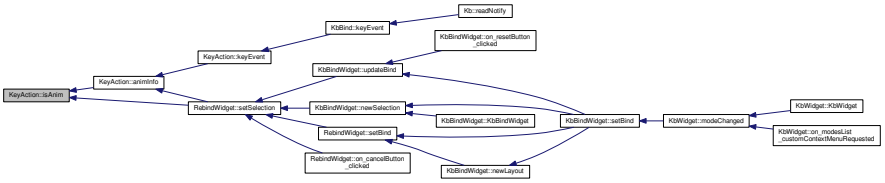
Definition at line 159 of file keyaction.h.

References _value.

Referenced by animInfo(), and RebindWidget::setSelection().

```
159 { return _value.startsWith("$anim:"); }
```

Here is the caller graph for this function:



12.41.4.10 bool KeyAction::isDPI ( ) const [inline]

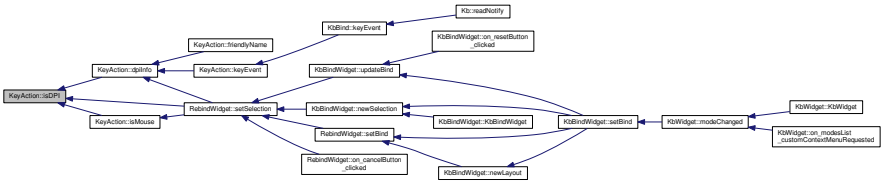
Definition at line 162 of file keyaction.h.

References _value.

Referenced by dpiInfo(), isMouse(), and RebindWidget::setSelection().

```
162 { return _value.startsWith("$dpi:"); }
```

Here is the caller graph for this function:



12.41.4.11 bool KeyAction::isMacro ( ) const [inline]

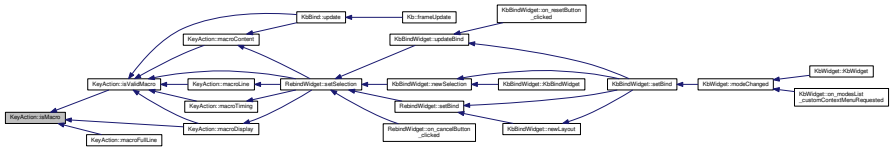
Definition at line 160 of file keyaction.h.

References _value.

Referenced by isValidMacro(), macroDisplay(), and macroFullLine().

```
160 { return _value.startsWith("$macro:"); }
```

Here is the caller graph for this function:



12.41.4.12 bool KeyAction::isMedia ( ) const [inline]

Definition at line 156 of file keyaction.h.

References _value.

```
156 { return _value == "mute" || _value == "volup" || _value == "voldn" ||
      _value == "stop" || _value == "prev" || _value == "play" ||
      _value == "next"; }
```

12.41.4.13 bool KeyAction::isMouse ( ) const [inline]

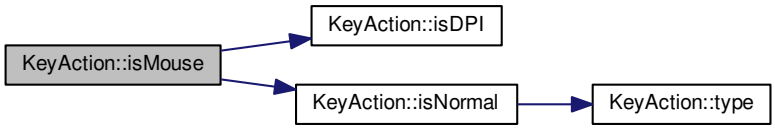
Definition at line 163 of file keyaction.h.

References _value, isDPI(), and isNormal().

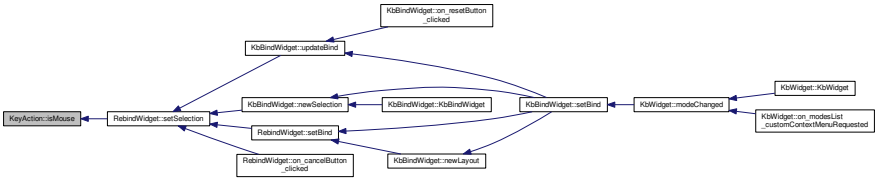
Referenced by RebindWidget::setSelection().

```
163 { return (isNormal() && (_value.startsWith("mouse") || _value.startsWith("wheel"))) ||
      isDPI(); }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.41.4.14 bool KeyAction::isNormal ( ) const [inline]

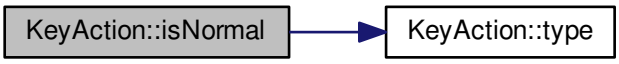
Definition at line 153 of file keyaction.h.

References NORMAL, and type().

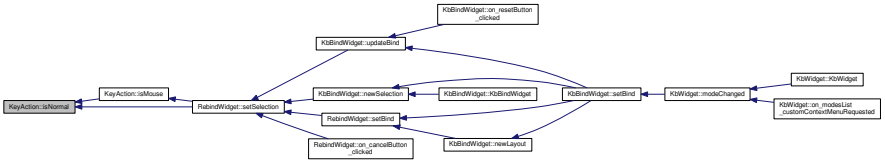
Referenced by isMouse(), and RebindWidget::setSelection().

```
153 { return type() == NORMAL; }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.41.4.15 bool KeyAction::isProgram ( ) const [inline]

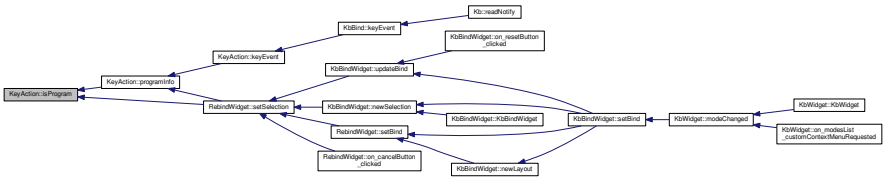
Definition at line 158 of file keyaction.h.

References _value.

Referenced by programInfo(), and RebindWidget::setSelection().

```
158 { return _value.startsWith("$program:"); }
```

Here is the caller graph for this function:



12.41.4.16 bool KeyAction::isSpecial ( ) const [inline]

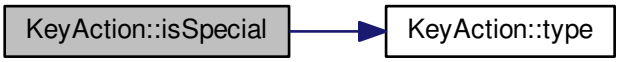
Definition at line 154 of file keyaction.h.

References SPECIAL, and type().

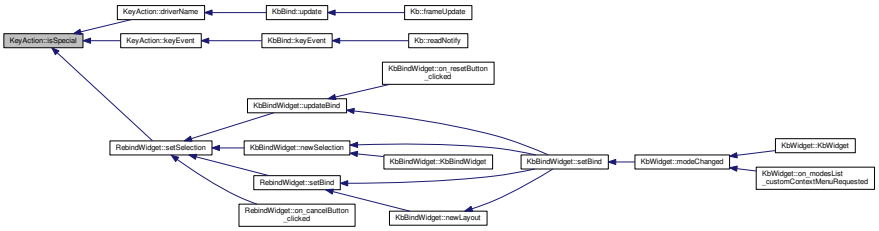
Referenced by driverName(), keyEvent(), and RebindWidget::setSelection().

```
154 { return type() == SPECIAL; }
```

Here is the call graph for this function:



Here is the caller graph for this function:



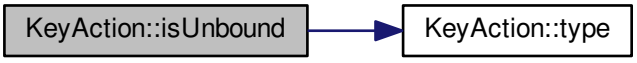
12.41.4.17 bool KeyAction::isUnbound ( ) const [inline]

Definition at line 152 of file keyaction.h.

References type(), and UNBOUND.

```
152 { return type() == UNBOUND; }
```

Here is the call graph for this function:



12.41.4.18 bool KeyAction::isValidMacro ( ) const [inline]

Returns

bool as true iff the macro definition contains all four elements.

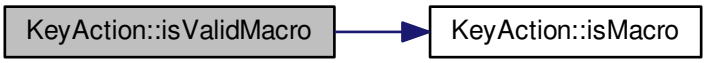
Definition at line 55 of file keyaction.h.

References _value, and isMacro().

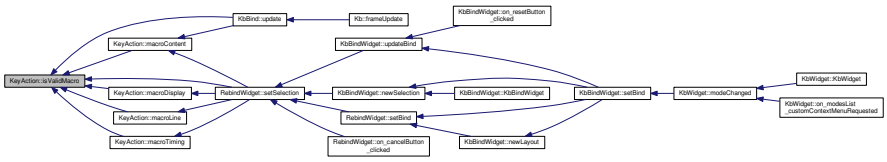
Referenced by macroContent(), macroDisplay(), macroLine(), macroTiming(), RebindWidget::setSelection(), and KbBind::update().

```
55                                      {
56          if (isMacro()) {
57              QStringList ret;
58              ret =_value.split(":");
59              return ((ret.count() >= 4) && (ret.count() <= 5));
60          } else {
61              return false;
62          }
63      }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.41.4.19 void KeyAction::keyEvent (KbBind * *bind,* bool *down*)

Definition at line 245 of file keyaction.cpp.

References _value, adjustDisplay(), animInfo(), KbPerf::baseDpiIdx(), KbProfile::currentMode(), Kb::currentProfile(), KbBind::devParent(), KbLight::dimming(), KbPerf::DPI_COUNT, DPI_CUSTOM, DPI_CYCLE_DOWN, DPI_CYCLE_UP, DPI_DOWN, DPI_SNIPER, DPI_UP, KbPerf::dpiCycleDown(), KbPerf::dpiCycleUp(), KbPerf::dpiDown(), dpiInfo(), KbPerf::dpiUp(), KbLight::findAnim(), KbProfile::indexOf(), KbAnim::isActive(), isSpecial(), KbBind::light(), LIGHT_DOWN, LIGHT_DOWN_WRAP, LIGHT_UP, LIGHT_UP_WRAP, LOCK_OFF, LOCK_ON, LOCK_TOGGLE, KbLight::MAX_DIM, MODE_NEXT, MODE_NEXT_WRAP, MODE_PREV, MODE_PREV_WRAP, KbProfile::modeCount(), KbProfile::modes(), KbBind::perf(), KbPerf::popDpi(), preProgram, PROGRAM_PR_KPSTOP, PROGRAM_PR_KRSTOP, PROGRAM_PR_MULTI, PROGRAM_RE_KPSTOP, PROGRAM_RE_MULTI, programInfo(), KbPerf::pushDpi(), KbPerf::pushSniper(), relProgram, Kb::setCurrentMode(), sniperValue, KbAnim::stop(), KbAnim::trigger(), and KbBind::winLock().

Referenced by KbBind::keyEvent().

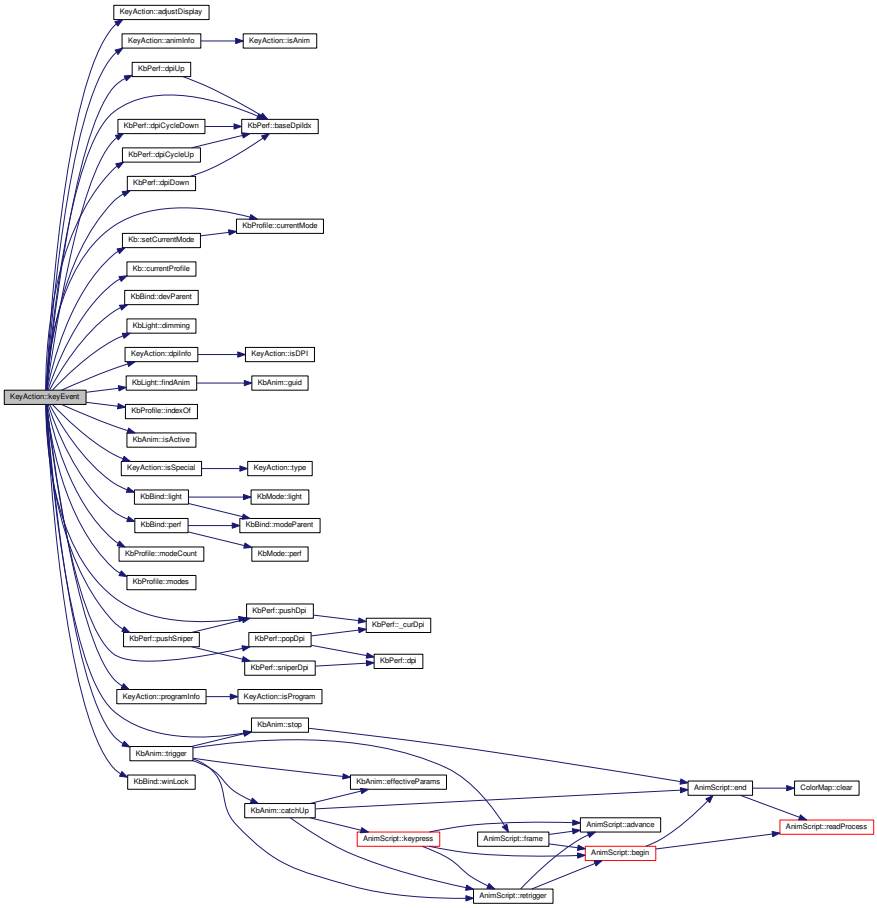
```
245                                             {
246     // No need to respond to standard actions
247     if(!isSpecial())
248         return;
249     QStringList parts = _value.split(":");
250     if(parts.length() < 2)
251         return;
252     QString prefix = parts[0];
253     int suffix = parts[1].toInt();
254     if(prefix == "$mode"){
255         if(!down)
256             return;
257         // Change mode
258         Kb* device = bind->devParent();
259         KbProfile* currentProfile = device->currentProfile();
```

```
260         int mode = currentProfile->indexOf(currentProfile->currentMode());
261         int modeCount = currentProfile->modeCount();
262         switch(suffix){
263         case MODE_PREV_WRAP:
264             mode--;
265             if(mode < 0)
266                 mode = modeCount - 1;
267             break;
268         case MODE_NEXT_WRAP:
269             mode++;
270             if(mode >= modeCount)
271                 mode = 0;
272             break;
273         case MODE_PREV:
274             mode--;
275             break;
276         case MODE_NEXT:
277             mode++;
278             break;
279         default:
280             // Absolute
281             mode = suffix;
282             break;
283         }
284         if(mode < 0 || mode >= modeCount)
285             return;
286         device->setCurrentMode(currentProfile->modes()[mode]);
287     } else if(prefix == "$dpi"){
288         KbPerf* perf = bind->perf();
289         int level = parts[1].split("+")[0].toInt();
290         switch(level){
291         case DPI_CYCLE_UP:
292             if(!down)
293                 return;
294             perf->dpiCycleUp();
295             break;
296         case DPI_CYCLE_DOWN:
297             if(!down)
298                 return;
299             perf->dpiCycleDown();
300             break;
301         case DPI_UP:
302             if(!down)
303                 return;
304             perf->dpiUp();
305             break;
306         case DPI_DOWN:
307             if(!down)
308                 return;
309             perf->dpiDown();
310             break;
311         case DPI_SNIPER:
312             if(down)
313                 sniperValue = perf->pushSniper();
314             else {
315                 perf->popDpi(sniperValue);
316                 sniperValue = 0;
317             }
318             break;
319         case DPI_CUSTOM:{
320             QPoint xy;
321             dpiInfo(xy);
322             if(xy.x() <= 0 || xy.y() <= 0)
323                 break;
324             if(down)
325                 sniperValue = perf->pushDpi(xy);
326             else {
327                 perf->popDpi(sniperValue);
328                 sniperValue = 0;
329             }
330             break;
331         }
332         default:
333             if(level < 1 || level >= KbPerf::DPI_COUNT
334                     || !down)
335                 return;
336             perf->baseDpiIdx(level);
337             break;
338         }
339     } else if(prefix == "$light"){
340         if(!down)
341             return;
342         // Change brightness
343         KbLight* light = bind->light();
344         int dim = light->dimming();
345         switch(suffix){
346         case LIGHT_UP:
```

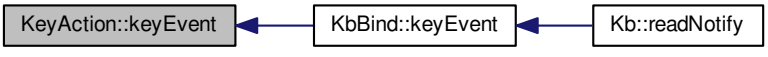
```
347             if(dim > 0)
348                 dim--;
349             break;
350         case LIGHT_DOWN:
351             if(dim < KbLight::MAX_DIM)
352                 dim++;
353             break;
354         case LIGHT_UP_WRAP:
355             dim--;
356             if(dim < 0)
357                 dim = KbLight::MAX_DIM;
358             break;
359         case LIGHT_DOWN_WRAP:
360             dim++;
361             if(dim > KbLight::MAX_DIM)
362                 dim = 0;
363             break;
364         }
365         light->dimming(dim);
366     } else if(prefix == "$lock"){
367         if(!down)
368             return;
369         // Change win lock
370         switch(suffix){
371         case LOCK_TOGGLE:
372             bind->winLock(!bind->winLock());
373             break;
374         case LOCK_ON:
375             bind->winLock(true);
376             break;
377         case LOCK_OFF:
378             bind->winLock(false);
379             break;
380         }
381     } else if(prefix == "$anim"){
382         // Start animation
383         bool onlyOnce = false, stopOnRelease = false;
384         QUuid id = animInfo(onlyOnce, stopOnRelease);
385         KbAnim* anim = bind->light()->findAnim(id);
386         if(!anim)
387             return;
388         if(down){
389             if(!onlyOnce || !anim->isActive())
390                 // If "only once" is enabled, don't start the animation when it's already running
391                 anim->trigger(QDateTime::currentMSecsSinceEpoch(), true);
392         } else if(stopOnRelease){
393             // Key released - stop animation
394             anim->stop();
395         }
396     } else if(prefix == "$program"){
397         // Launch program
398         QString onPress, onRelease;
399         int stop = programInfo(onPress, onRelease);
400         // Stop running programs based on setting
401         QProcess* process = 0;
402         if(down){
403             if(stop & PROGRAM_PR_KPSTOP){
404                 process = preProgram;
405                 if(process)
406                     process->kill();
407                 process = 0;
408             }
409             if(stop & PROGRAM_RE_KPSTOP)
410                 process = relProgram;
411         } else {
412             if(stop & PROGRAM_PR_KRSTOP)
413                 process = preProgram;
414         }
415         if(process)
416             process->kill();
417         // Launch new process if requested
418         QString& program = down ? onPress : onRelease;
419         if(program.isEmpty())
420             return;
421         // Check if the program is running already. If so, don't start it again.
422         process = down ? preProgram : relProgram;
423         if(process){
424             if(process->state() == QProcess::NotRunning)
425                 delete process;
426             else
427                 return;
428         }
429
430         // Adjust the selected display.
431         adjustDisplay();
432
433         // Start the program. Wrap it around sh to parse arguments.
```

```
434         if((down && (stop & PROGRAM_PR_MULTI))
435                 || (!down && (stop & PROGRAM_RE_MULTI))){
436             // Multiple instances allowed? Start detached process
437             QProcess::startDetached("sh", QStringList() << "-c" << program);
438         } else {
439             process = new QProcess(this);
440             process->start("sh", QStringList() << "-c" << program);
441             if(down)
442                 preProgram = process;
443             else
444                 relProgram = process;
445         }
446     } else if (prefix == "$macro") {
447         // Do nothing, because all work is done by the keyboard itself.
448         // For now, there is no reason to react on G-key press or release.
449         // If u find some reason, then here is the place for it.
450     }
451 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.41.4.20 void KeyAction::keyRelease (KbBind * *bind*)

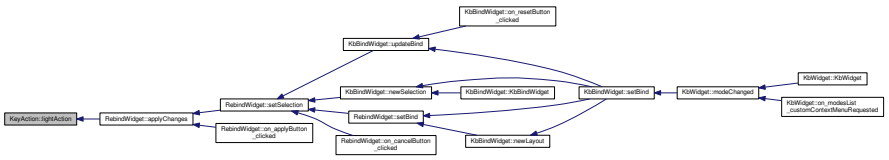
12.41.4.21 QString KeyAction::lightAction (int *type* = LIGHT_UP_WRAP) [static]

Definition at line 169 of file keyaction.cpp.

Referenced by RebindWidget::applyChanges().

```
169                                          {
170     return QString("$light:%1").arg(type);
171 }
```

Here is the caller graph for this function:



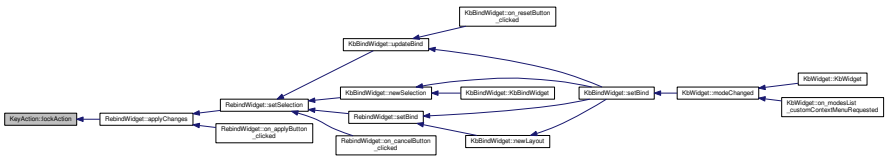
12.41.4.22 QString KeyAction::lockAction (int *type* = LOCK_TOGGLE) [static]

Definition at line 173 of file keyaction.cpp.

Referenced by RebindWidget::applyChanges().

```
173                                          {
174     return QString("$lock:%1").arg(type);
175 }
```

Here is the caller graph for this function:



12.41.4.23 QString KeyAction::macroAction (QString *macroDef*) [static]

KeyAction::macroAction is called when applying changes on a macro definition. macroAction ist called while being in the macro pane and clicking Apply with something in the Macro Text Box. It tags that input with "$macro:" for further recognition.

Parameters

macroDef	holds the String containing parts 2-5 of a complete macro definition.

Returns

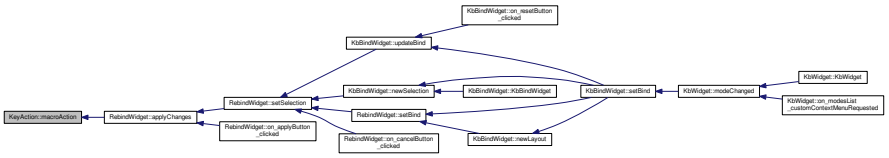
QString holding the complete G-Key macro definition (parts 1-5)

Definition at line 523 of file keyaction.cpp.

Referenced by RebindWidget::applyChanges().

```
523                                                {
524     return QString ("$macro:%1").arg(macroDef);
525 }
```

Here is the caller graph for this function:



12.41.4.24 QString KeyAction::macroContent ( ) const [inline]

Returns

QString macroContent

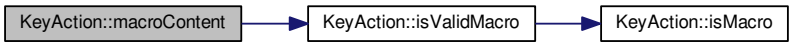
Definition at line 86 of file keyaction.h.

References _value, and isValidMacro().

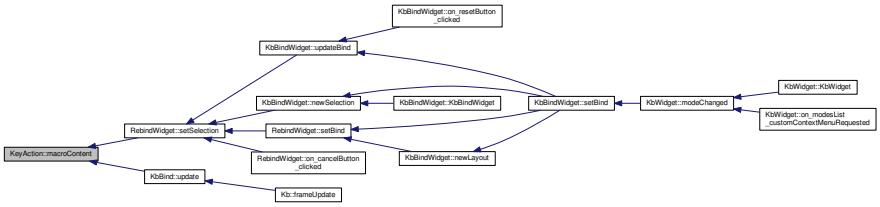
Referenced by RebindWidget::setSelection(), and KbBind::update().

```
86                                        {
87         // return isValidMacro() ? _value.split(":")[1].replace(QRegExp("=\\d+"), "") : ""; ///< Is used if
       we have ckb without delay handling
88         return isValidMacro() ? _value.split(":")[1] : "";
89     }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.41.4.25 void KeyAction::macroDisplay ( )

KeyAction::macroDisplay is just for debugging. It shows the content of the key action and some other info.

General Info on KeyAction::_value for macros: That string consists of 4 elements, all delimited by ":".

1. Macro command indicator "$macro:"
2. Macro Key Definition (coming from pteMacroBox): This sequence will program the keyboard and is hardly readable
3. Readable Macro String: This is displayed in pteMacroText
4. Readable Macro Comment:This is displayed in pteMacroComment
5. completely unreadable original macro information with timing values

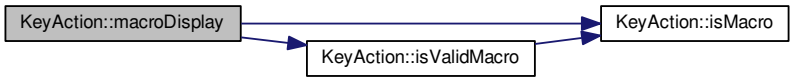
Definition at line 456 of file keyaction.cpp.

References _value, isMacro(), and isValidMacro().

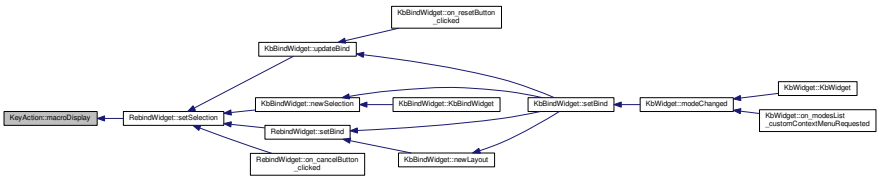
Referenced by RebindWidget::setSelection().

```
456                                {
457     qDebug() << "isMacro returns" << (isMacro() ? "true" : "false");
458     qDebug() << "isValidMacro returns" << (isValidMacro() ? "true" : "false");
459     QStringList ret =_value.split(":");
460     qDebug() << "Macro definition conains" << ret.count() << "elements";
461     qDebug() << "Macro definition is" << _value;
462 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.41.4.26 QString KeyAction::macroFullLine ( ) const [inline]

Returns

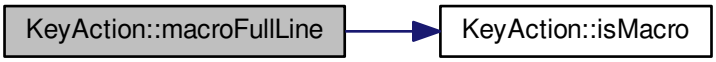
QString All 5 parts are returned in one QString. If no definition exists, return ""

Definition at line 43 of file keyaction.h.

References _value, and isMacro().

```
43                                          {
44          return isMacro() ? _value.right(_value.length()-1) : "";
45      }
```

Here is the call graph for this function:



12.41.4.27 QStringList KeyAction::macroLine ( ) const [inline]

Returns

QStringList returns the Macro Key Definition, Readble Macro String, Readable Macro Comment and the original timing information (if it exists as a 5th part) as QStringList.

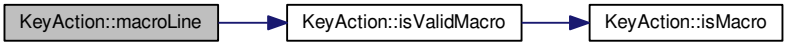
Definition at line 73 of file keyaction.h.

References _value, and isValidMacro().

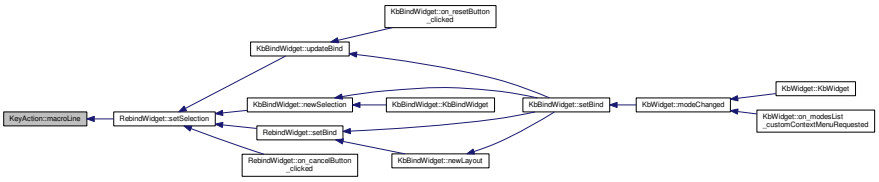
Referenced by RebindWidget::setSelection().

```
73                                          {
74          if (isValidMacro()) {
75              QStringList ret =_value.split(":");
76              ret.removeFirst();
77              return ret;
78          } else return QStringList();
79      }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.41.4.28 QString KeyAction::macroTiming ( ) const [inline]

Returns

QString macroTiming

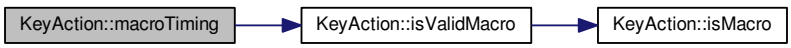
Definition at line 98 of file keyaction.h.

References _value, and isValidMacro().

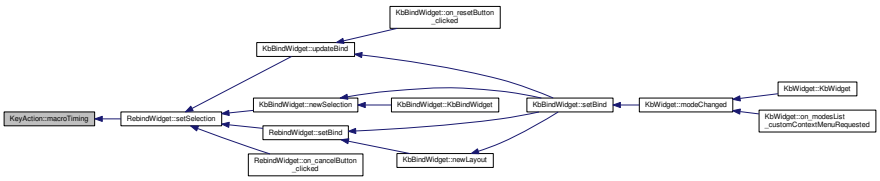
Referenced by RebindWidget::setSelection().

```
98                                      {
99          if (isValidMacro()) {
100             QStringList rval = _value.split(":");
101             return (rval.length() == 4)? rval[1] : rval[4];
102         }
103         return QString("");
104     }
```

Here is the call graph for this function:



Here is the caller graph for this function:



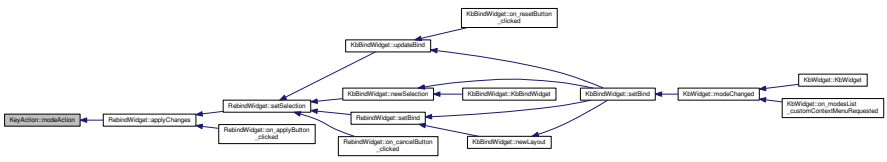
12.41.4.29 QString KeyAction::modeAction (int *mode*) [static]

Definition at line 158 of file keyaction.cpp.

Referenced by RebindWidget::applyChanges().

```
158                                         {
159     return QString("$mode:%1").arg(mode);
160 }
```

Here is the caller graph for this function:



12.41.4.30 static QString KeyAction::noAction ( ) [inline], [static]

Definition at line 25 of file keyaction.h.

```
25 { return ""; }
```

12.41.4.31 KeyAction::operator QString ( ) const [inline]

Definition at line 20 of file keyaction.h.

References _value.

```
20 { return _value; }
```

12.41.4.32 void KeyAction::operator= (const KeyAction & *rhs*) [inline], [private]

Definition at line 186 of file keyaction.h.

```
186 {}
```

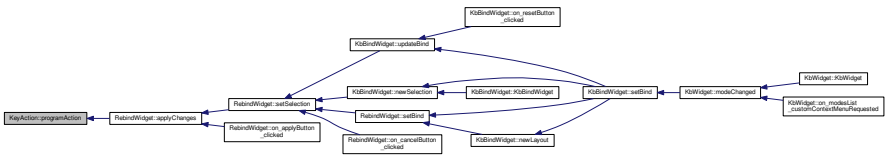
12.41.4.33 QString KeyAction::programAction (const QString & *onPress*, const QString & *onRelease*, int *stop*) [static]

Definition at line 177 of file keyaction.cpp.

Referenced by RebindWidget::applyChanges().

```
177                                                                                                 {
178     // URL-encode the commands and place them in the string (":" and "+" are both replaced, so they won't
       interfere)
179     return "$program:" + QString::fromUtf8(QUrl::toPercentEncoding(onPress.trimmed())) + "+" +
      QString::fromUtf8(QUrl::toPercentEncoding(onRelease.trimmed())) + QString("+%1").arg(stop);
180 }
```

Here is the caller graph for this function:



12.41.4.34 int KeyAction::programInfo (QString & *onPress*, QString & *onRelease*) const

Definition at line 199 of file keyaction.cpp.

References _value, and isProgram().

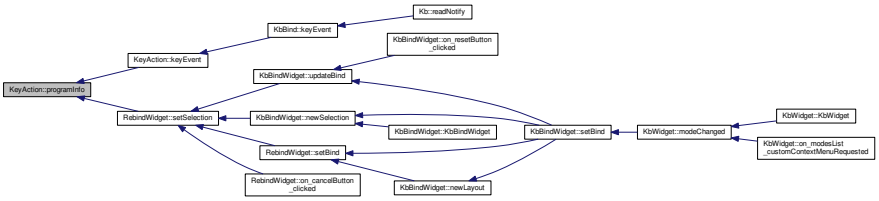
Referenced by keyEvent(), and RebindWidget::setSelection().

```
199                                                                                 {
200     if(!isProgram())
201         return 0;
202     QString param = _value.mid(9);
203     QStringList programs = param.split("+");
```

```
204     if(programs.length() != 3)
205         return 0;
206     onPress = QUrl::fromPercentEncoding(programs[0].toUtf8());
207     onRelease = QUrl::fromPercentEncoding(programs[1].toUtf8());
208     return programs[2].toInt();
209 }
```

Here is the call graph for this function:

Here is the caller graph for this function:



12.41.4.35 QString KeyAction::specialInfo (int & *parameter*) const

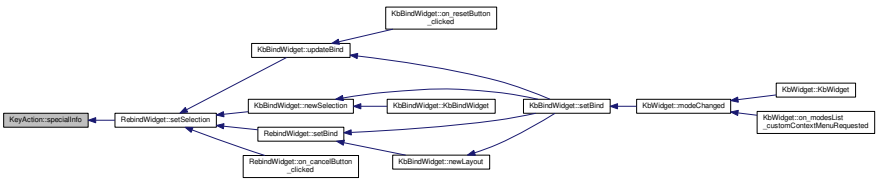
Definition at line 189 of file keyaction.cpp.

References _value.

Referenced by RebindWidget::setSelection().

```
189                                                 {
190     QStringList list = _value.split(":");
191     if(list.length() < 2){
192         parameter = INT_MIN;
193         return "";
194     }
195     parameter = list[1].toInt();
196     return list[0].replace("$", "");
197 }
```

Here is the caller graph for this function:



12.41.4.36 KeyAction::Type KeyAction::type ( ) const

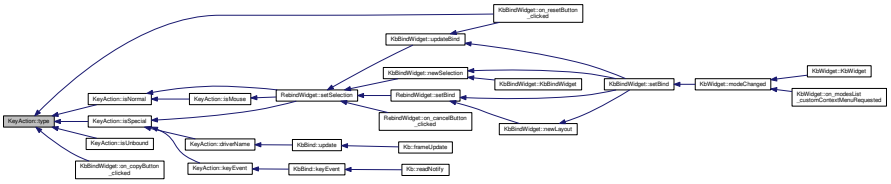
Definition at line 14 of file keyaction.cpp.

References _value, NORMAL, SPECIAL, and UNBOUND.

Referenced by isNormal(), isSpecial(), isUnbound(), KbBindWidget::on_copyButton_clicked(), and KbBindWidget::on_resetButton_clicked().

```
14                                      {
15     if(_value.isEmpty())
16         return UNBOUND;
17     if(_value.at(0) == '$')
18         return SPECIAL;
19     return NORMAL;
20 }
```

Here is the caller graph for this function:



12.41.4.37 QString KeyAction::value ( ) const [inline]

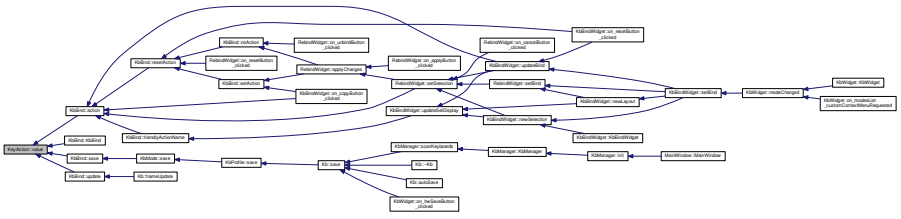
Definition at line 19 of file keyaction.h.

References _value.

Referenced by KbBind::action(), KbBind::KbBind(), KbBind::save(), and KbBind::update().

```
19 { return _value; }
```

Here is the caller graph for this function:



12.41.5 Field Documentation

12.41.5.1 QString KeyAction::_value [private]

Definition at line 189 of file keyaction.h.

Referenced by animInfo(), dpiInfo(), driverName(), friendlyName(), isAnim(), isDPI(), isMacro(), isMedia(), isMouse(), isProgram(), isValidMacro(), keyEvent(), macroContent(), macroDisplay(), macroFullLine(), macroLine(), macroTiming(), operator QString(), programInfo(), specialInfo(), type(), and value().

12.41.5.2 const int KeyAction::DPI_CUSTOM = 6 [static]

Definition at line 128 of file keyaction.h.

Referenced by dpiAction(), dpiInfo(), friendlyName(), keyEvent(), and RebindWidget::setSelection().

12.41.5.3 const int KeyAction::DPI_CYCLE_DOWN = -3 [static]

Definition at line 126 of file keyaction.h.

Referenced by friendlyName(), and keyEvent().

12.41.5.4 const int KeyAction::DPI_CYCLE_UP = -4 [static]

Definition at line 126 of file keyaction.h.

Referenced by friendlyName(), and keyEvent().

12.41.5.5 const int KeyAction::DPI_DOWN = -1 [static]

Definition at line 127 of file keyaction.h.

Referenced by friendlyName(), and keyEvent().

12.41.5.6 const int KeyAction::DPI_SNIPER = 0 [static]

Definition at line 128 of file keyaction.h.

Referenced by friendlyName(), and keyEvent().

12.41.5.7 const int KeyAction::DPI_UP = -2 [static]

Definition at line 127 of file keyaction.h.

Referenced by friendlyName(), and keyEvent().

12.41.5.8 const int KeyAction::LIGHT_DOWN = 1 [static]

Definition at line 131 of file keyaction.h.

Referenced by friendlyName(), keyEvent(), and RebindWidget::setSelection().

12.41.5.9 const int KeyAction::LIGHT_DOWN_WRAP = 3 [static]

Definition at line 132 of file keyaction.h.

Referenced by friendlyName(), keyEvent(), and RebindWidget::setSelection().

12.41.5.10 const int KeyAction::LIGHT_UP = 0 [static]

Definition at line 131 of file keyaction.h.

Referenced by friendlyName(), keyEvent(), and RebindWidget::setSelection().

12.41.5.11 const int KeyAction::LIGHT_UP_WRAP = 2 [static]

Definition at line 132 of file keyaction.h.

Referenced by friendlyName(), keyEvent(), and RebindWidget::setSelection().

12.41.5.12 const int KeyAction::LOCK_OFF = 2 [static]

Definition at line 135 of file keyaction.h.

Referenced by friendlyName(), and keyEvent().

12.41.5.13 const int KeyAction::LOCK_ON = 1 [static]

Definition at line 135 of file keyaction.h.

Referenced by friendlyName(), and keyEvent().

12.41.5.14 const int KeyAction::LOCK_TOGGLE = 0 [static]

Definition at line 135 of file keyaction.h.

Referenced by friendlyName(), and keyEvent().

12.41.5.15 const int KeyAction::MODE_NEXT = -1 [static]

Definition at line 122 of file keyaction.h.

Referenced by friendlyName(), keyEvent(), and RebindWidget::setSelection().

12.41.5.16 const int KeyAction::MODE_NEXT_WRAP = -3 [static]

Definition at line 123 of file keyaction.h.

Referenced by friendlyName(), keyEvent(), and RebindWidget::setSelection().

12.41.5.17 const int KeyAction::MODE_PREV = -2 [static]

Definition at line 122 of file keyaction.h.

Referenced by friendlyName(), keyEvent(), and RebindWidget::setSelection().

12.41.5.18 const int KeyAction::MODE_PREV_WRAP = -4 [static]

Definition at line 123 of file keyaction.h.

Referenced by friendlyName(), keyEvent(), and RebindWidget::setSelection().

12.41.5.19 QProcess* KeyAction::preProgram [private]

Definition at line 192 of file keyaction.h.

Referenced by keyEvent(), and ~KeyAction().

12.41.5.20 const int KeyAction::PROGRAM_PR_INDEF = 0x00 [static]

Definition at line 138 of file keyaction.h.

12.41.5.21 const int KeyAction::PROGRAM_PR_KPSTOP = 0x02 [static]

Definition at line 138 of file keyaction.h.

Referenced by keyEvent().

12.41.5.22 const int KeyAction::PROGRAM_PR_KRSTOP = 0x01 [static]

Definition at line 138 of file keyaction.h.

Referenced by keyEvent().

12.41.5.23 const int KeyAction::PROGRAM_PR_MULTI = 0x04 [static]

Definition at line 138 of file keyaction.h.

Referenced by RebindWidget::applyChanges(), keyEvent(), and RebindWidget::setSelection().

12.41.5.24 const int KeyAction::PROGRAM_RE_INDEF = 0x00 [static]

Definition at line 139 of file keyaction.h.

Referenced by RebindWidget::applyChanges(), and RebindWidget::setSelection().

12.41.5.25 const int KeyAction::PROGRAM_RE_KPSTOP = 0x20 [static]

Definition at line 139 of file keyaction.h.

Referenced by RebindWidget::applyChanges(), keyEvent(), and RebindWidget::setSelection().

12.41.5.26 const int KeyAction::PROGRAM_RE_MULTI = 0x40 [static]

Definition at line 139 of file keyaction.h.

Referenced by RebindWidget::applyChanges(), keyEvent(), and RebindWidget::setSelection().

12.41.5.27 QProcess* KeyAction::relProgram [private]

Definition at line 193 of file keyaction.h.

Referenced by keyEvent(), and ~KeyAction().

12.41.5.28 quint64 KeyAction::sniperValue [private]

Definition at line 196 of file keyaction.h.

Referenced by keyEvent().

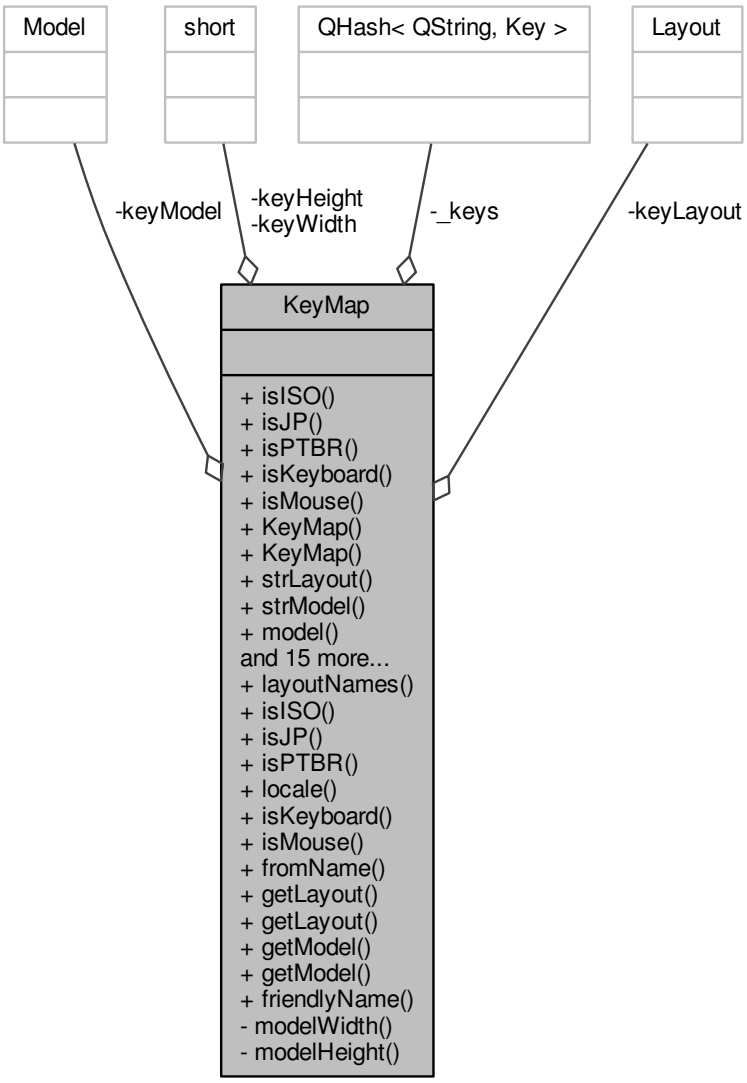
The documentation for this class was generated from the following files:

- src/ckb/keyaction.h
- src/ckb/keyaction.cpp

12.42 KeyMap Class Reference

`#include <src/ckb/keymap.h>`

Collaboration diagram for KeyMap:



Public Types

- enum Model {
  NO_MODEL = -1, K63, K65, K68,
  K70, K95, K95P, STRAFE,
  M65, SABRE, SCIMITAR, HARPOON,
  GLAIVE, _MODEL_MAX }
- enum Layout {

NO_LAYOUT = -1, DK, EU, EU_DVORAK,
GB, GB_DVORAK, US, US_DVORAK,
FR, DE, IT, JP,
NO, PL, PT_BR, MX,
ES, SE, _LAYOUT_MAX }

Public Member Functions

- bool isISO () const
- bool isJP () const
- bool isPTBR () const
- bool isKeyboard () const
- bool isMouse () const
- KeyMap ()
- KeyMap (Model _keyModel, Layout _keyLayout)
- QString strLayout () const
- QString strModel () const
- Model model () const
- Layout layout () const
- QString name () const
- uint count () const
- uint width () const
- uint height () const
- Key key (const QString &name) const
- Key operator[] (const QString &name) const
- bool contains (const QString &name) const
- const QHash< QString, Key > & map () const
- operator const QHash< QString, Key > & () const
- QStringList keys () const
- QList< Key > positions () const
- QString toStorage (const QString &name)
- QString fromStorage (const QString &storage)
- QStringList byPosition () const

Static Public Member Functions

- static QStringList layoutNames ()
- static bool isISO (Layout layout)
- static bool isJP (Layout layout)
- static bool isPTBR (Layout layout)
- static Layout locale ()
- static bool isKeyboard (Model model)
- static bool isMouse (Model model)
- static KeyMap fromName (const QString &name)
- static Layout getLayout (const QString &name)
- static QString getLayout (Layout layout)
- static Model getModel (const QString &name)
- static QString getModel (Model model)
- static QString friendlyName (const QString &key, Layout layout=US)

Static Private Member Functions

- static int modelWidth (Model model)
- static int modelHeight (Model model)

Private Attributes

- QHash< QString, Key > _keys
- short keyWidth
- short keyHeight
- Model keyModel:8
- Layout keyLayout:8

12.42.1 Detailed Description

Definition at line 49 of file keymap.h.

12.42.2 Member Enumeration Documentation

12.42.2.1 enum KeyMap::Layout

Enumerator

NO_LAYOUT
DK
EU
EU_DVORAK
GB
GB_DVORAK
US
US_DVORAK
FR
DE
IT
JP
NO
PL
PT_BR
MX
ES
SE
_LAYOUT_MAX

Definition at line 70 of file keymap.h.

```
70                 {
71         NO_LAYOUT = -1,
72         DK,                     // Danish
73         EU,                     // English (EU)
74         EU_DVORAK,
75         GB,                     // English (UK)
76         GB_DVORAK,
77         US,                     // English (US)
78         US_DVORAK,
79         FR,                     // French
80         DE,                     // German
81         IT,                     // Italian
82         JP,                     // Japanese
83         NO,                     // Norwegian
84         PL,                     // Polish (identical to US)
85         PT_BR,                  // Portuguese (Brazil)
86         MX,                     // Spanish (Mexico/Latin America)
87         ES,                     // Spanish (Spain)
88         SE,                     // Swedish
89         _LAYOUT_MAX
90     };
```

12.42.2.2 enum KeyMap::Model

Enumerator

NO_MODEL

K63

K65

K68

K70

K95

K95P

STRAFE

M65

SABRE

SCIMITAR

HARPOON

GLAIVE

_MODEL_MAX

Definition at line 51 of file keymap.h.

```
51                 {
52         NO_MODEL = -1,
53         // Keyboard models
54         K63,
55         K65,
56         K68,
57         K70,
58         K95,
59         K95P,
60         STRAFE,
61         // Mouse models
62         M65,
63         SABRE,
64         SCIMITAR,
65         HARPOON,
66         GLAIVE,
67         _MODEL_MAX
68     };
```

12.42.3 Constructor & Destructor Documentation

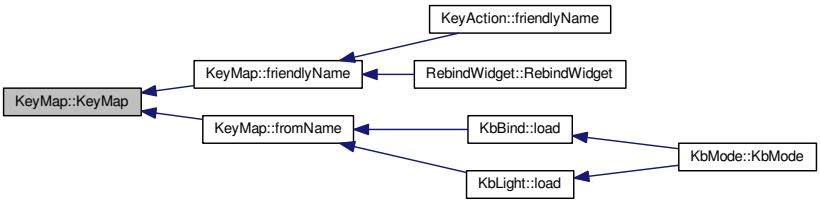
12.42.3.1 KeyMap::KeyMap ( )

Definition at line 897 of file keymap.cpp.

Referenced by friendlyName(), and fromName().

```
897                 :
898       keyWidth(0), keyHeight(0),
899       keyModel(NO_MODEL), keyLayout(NO_LAYOUT)
900 {}
```

Here is the caller graph for this function:



12.42.3.2 KeyMap::KeyMap (Model *_keyModel,* Layout *_keyLayout*)

Definition at line 891 of file keymap.cpp.

```
891                                                :
892     _keys(getMap(_keyModel, _keyLayout)),
893     keyWidth(modelWidth(_keyModel)), keyHeight(
      modelHeight(_keyModel)),
894     keyModel(_keyModel), keyLayout(_keyLayout)
895 {}
```

12.42.4 Member Function Documentation

12.42.4.1 QStringList KeyMap::byPosition ( ) const

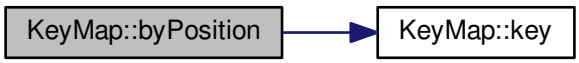
Definition at line 902 of file keymap.cpp.

References key(), Key::x, and Key::y.

Referenced by RebindWidget::setBind().

```
902                                      {
903     // Use QMaps to order the keys
904     QMap<int, QMap<int, QString> > ordered;
905     QHashIterator<QString, Key> i(*this);
906     while(i.hasNext()){
907         i.next();
908         const Key& key = i.value();
909         ordered[key.y][key.x] = i.key();
910     }
911     // Merge them into a single list
912     QStringList result;
913     QMapIterator<int, QMap<int, QString> > y(ordered);
914     while(y.hasNext()){
915         y.next();
916         QMapIterator<int, QString> x(y.value());
917         while(x.hasNext()){
918             x.next();
919             result << x.value();
920         }
921     }
922     return result;
923 }
```

Here is the call graph for this function:



Here is the caller graph for this function:

12.42.4.2 bool KeyMap::contains (const QString & *name*) const [inline]

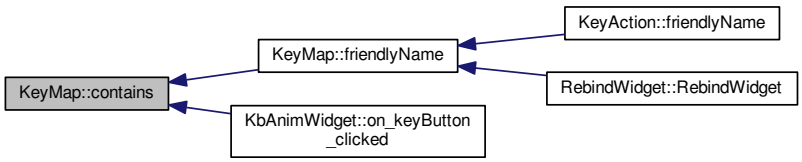
Definition at line 143 of file keymap.h.

References _keys.

Referenced by friendlyName(), and KbAnimWidget::on_keyButton_clicked().

```
143 { return _keys.contains(name); }
```

Here is the caller graph for this function:



12.42.4.3 uint KeyMap::count ( ) const [inline]

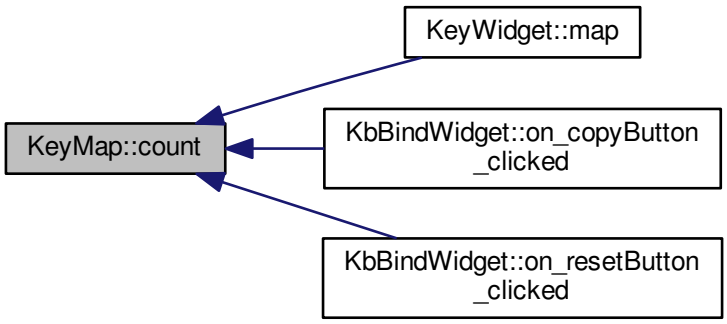
Definition at line 134 of file keymap.h.

References _keys.

Referenced by KeyWidget::map(), KbBindWidget::on_copyButton_clicked(), and KbBindWidget::on_resetButton_-clicked().

```
134 { return _keys.count(); }
```

Here is the caller graph for this function:



12.42.4.4 QString KeyMap::friendlyName (const QString & *key*, Layout *layout* = US) [static]

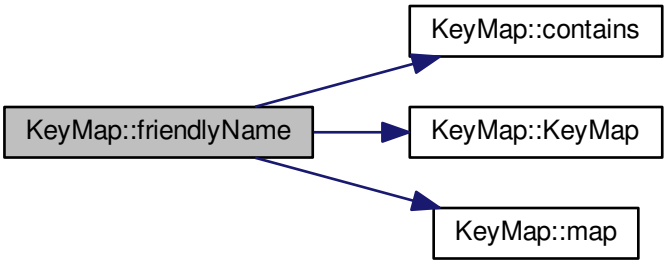
Definition at line 925 of file keymap.cpp.

References contains(), HARPOON, K95, KeyMap(), M65, map(), SCIMITAR, and STRAFE.

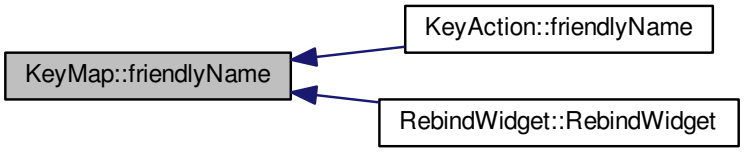
Referenced by KeyAction::friendlyName(), and RebindWidget::RebindWidget().

```
925                                                         {
926     // Try K95 map first
927     // FIXME: This is an odd function and probably should be refactored
928     // it would probably be best to remove the friendly names from the maps and have a completely separate
      name->friendlyName store
929     KeyMap map(K95, layout);
930     if(map.contains(key))
931         return map[key].friendlyName();
932
933     // The only key missing from it should be Fn, which is found on STRAFE
934     map = KeyMap(STRAFE, layout);
935     if(map.contains(key))
936         return map[key].friendlyName();
937
938     // Additionally, there are a handful of keys not present on any physical keyboard, but we need names
      for them
939     if(key == "f13" || key == "f14" || key == "f15" || key == "f16" ||
     key == "f17" || key == "f18" || key == "f19" || key == "f20")
940         return key.toUpper();
941     else if(key == "lightup")
942         return "Screen Brightness Up";
943     else if(key == "lightdn")
944         return "Screen Brightness Down";
945     else if(key == "eject" || key == "power")
946         return key[0].toUpper() + key.mid(1);   // capitalize first letter
947
948     // All other names are found on mice
949     map = KeyMap(SCIMITAR, layout);
950     if(map.contains(key))
951         return map[key].friendlyName();
952     map = KeyMap(M65, layout);
953     if(map.contains(key))
954         return map[key].friendlyName();
955     map = KeyMap(HARPOON, layout);
956     if(map.contains(key))
957         return map[key].friendlyName();
958
959     // Not found at all
960     return "";
961 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.42.4.5 KeyMap KeyMap::fromName (const QString & *name*) [static]

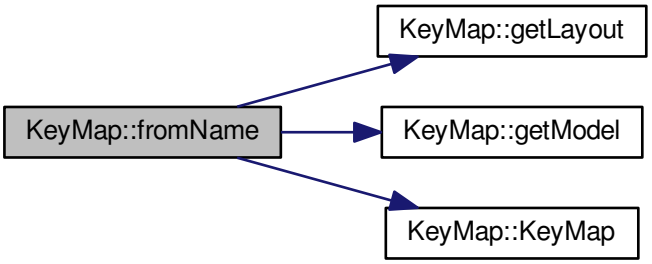
Definition at line 835 of file keymap.cpp.

References getLayout(), getModel(), and KeyMap().

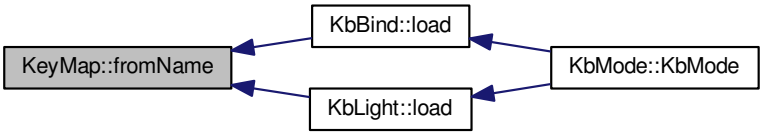
Referenced by KbBind::load(), and KbLight::load().

```
835                                         {
836     QStringList list = name.trimmed().split(" ");
837     if(list.length() != 2)
838         return KeyMap();
839     return KeyMap(getModel(list[0]), getLayout(list[1]));
840 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



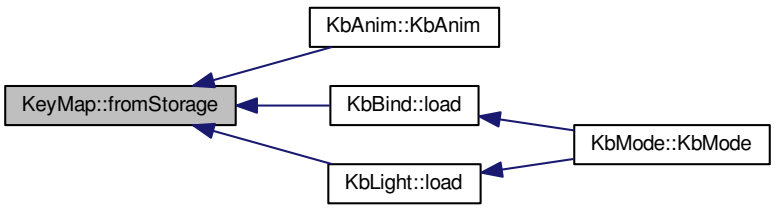
12.42.4.6 QString KeyMap::fromStorage (const QString & *storage*) [inline]

Definition at line 151 of file keymap.h.

Referenced by KbAnim::KbAnim(), KbBind::load(), and KbLight::load().

```
151 { QHashIterator<QString, Key> i(*this); while(i.hasNext()) { i.next(); const char* s = i.value().
      storageName(); if(s == storage) return i.value().name; } return storage; }
```

Here is the caller graph for this function:



12.42.4.7 KeyMap::Layout KeyMap::getLayout (const QString & *name*) [static]

Definition at line 677 of file keymap.cpp.

References DE, DK, ES, EU, EU_DVORAK, FR, GB, GB_DVORAK, IT, JP, MX, NO, NO_LAYOUT, PL, PT_BR, SE, US, and US_DVORAK.

Referenced by fromName(), SettingsWidget::on_layoutBox_activated(), SettingsWidget::SettingsWidget(), and strLayout().

```
677                                                 {
678     QString lower = name.toLower();
679     if(lower == "dk")
680         return DK;
681     if(lower == "eu")
682         return EU;
683     if(lower == "eu_dvorak")
684         return EU_DVORAK;
685     if(lower == "gb_dvorak")
686         return GB_DVORAK;
687     if(lower == "us")
688         return US;
689     if(lower == "us_dvorak")
690         return US_DVORAK;
691     if(lower == "fr")
692         return FR;
693     if(lower == "de")
694         return DE;
695     if(lower == "it")
696         return IT;
697     if(lower == "no")
698         return NO;
699     if(lower == "jp")
700         return JP;
701     if(lower == "pl")
702         return PL;
703     if(lower == "pt_br")
704         return PT_BR;
705     if(lower == "mx")
706         return MX;
707     if(lower == "es")
708         return ES;
709     if(lower == "se")
710         return SE;
711     if(lower == "gb")
712         return GB;
713     return NO_LAYOUT;
714 }
```

Here is the caller graph for this function:

12.42.4.8 QString KeyMap::getLayout (KeyMap::Layout *layout*) [static]

Definition at line 716 of file keymap.cpp.

References DE, DK, ES, EU, EU_DVORAK, FR, GB, GB_DVORAK, IT, JP, MX, NO, PL, PT_BR, SE, US, and US_DVORAK.

```
716                                                 {
717     switch(layout){
718     case DK:
719         return "dk";
720     case EU:
721         return "eu";
722     case EU_DVORAK:
```

```
723         return "eu_dvorak";
724     case GB:
725         return "gb";
726     case GB_DVORAK:
727         return "gb_dvorak";
728     case US:
729         return "us";
730     case US_DVORAK:
731         return "us_dvorak";
732     case FR:
733         return "fr";
734     case DE:
735         return "de";
736     case IT:
737         return "it";
738     case JP:
739         return "jp";
740     case NO:
741         return "no";
742     case PL:
743         return "pl";
744     case PT_BR:
745         return "pt_br";
746     case MX:
747         return "mx";
748     case ES:
749         return "es";
750     case SE:
751         return "se";
752     default:
753         return "";
754     }
755 }
```

12.42.4.9 KeyMap::Model KeyMap::getModel (const QString & *name*) [static]

Definition at line 775 of file keymap.cpp.

References GLAIVE, HARPOON, K63, K65, K68, K70, K95, K95P, M65, NO_MODEL, SABRE, SCIMITAR, and STRAFE.

Referenced by fromName(), Kb::Kb(), and strModel().

```
775                                                 {
776     QString lower = name.toLower();
777     if(lower == "k63")
778         return K63;
779     if(lower == "k65")
780         return K65;
781     if(lower == "k68")
782         return K68;
783     if(lower == "k70")
784         return K70;
785     if(lower == "k95")
786         return K95;
787     if(lower == "k95p")
788         return K95P;
789     if(lower == "strafe")
790         return STRAFE;
791     if(lower == "m65")
792         return M65;
793     if(lower == "sabre")
794         return SABRE;
795     if(lower == "scimitar")
796         return SCIMITAR;
797     if(lower == "harpoon")
798         return HARPOON;
799     if(lower == "glaive")
800         return GLAIVE;
801     return NO_MODEL;
802 }
```

Here is the caller graph for this function:

12.42.4.10 QString KeyMap::getModel (KeyMap::Model *model*) [static]

Definition at line 804 of file keymap.cpp.

References GLAIVE, HARPOON, K63, K65, K68, K70, K95, K95P, M65, SABRE, SCIMITAR, and STRAFE.

```
804                                             {
805     switch(model){
806     case K63:
807         return "k63";
808     case K65:
809         return "k65";
810     case K68:
811         return "k68";
812     case K70:
813         return "k70";
814     case K95:
815         return "k95";
816     case K95P:
817         return "k95P";
818     case STRAFE:
819         return "strafe";
820     case M65:
821         return "m65";
822     case SABRE:
823         return "sabre";
824     case SCIMITAR:
825         return "scimitar";
826     case HARPOON:
827         return "harpoon";
828     case GLAIVE:
829         return "glaive";
830     default:
831         return "";
832     }
833 }
```

12.42.4.11 uint KeyMap::height ( ) const [inline]

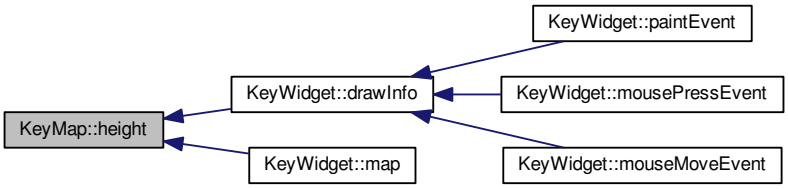
Definition at line 138 of file keymap.h.

References keyHeight.

Referenced by KeyWidget::drawInfo(), and KeyWidget::map().

```
138 { return keyHeight; }
```

Here is the caller graph for this function:



12.42.4.12 static bool KeyMap::isISO (Layout *layout*) [inline], [static]

Definition at line 94 of file keymap.h.

References PL, US, and US_DVORAK.

Referenced by KbBind::isISO(), and RebindWidget::setBind().

```
94 { return layout != US && layout != US_DVORAK && layout !=
       PL; }
```

Here is the caller graph for this function:

12.42.4.13 bool KeyMap::isISO ( ) const [inline]

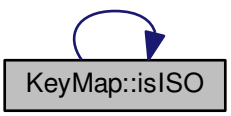
Definition at line 95 of file keymap.h.

References isISO(), and keyLayout.

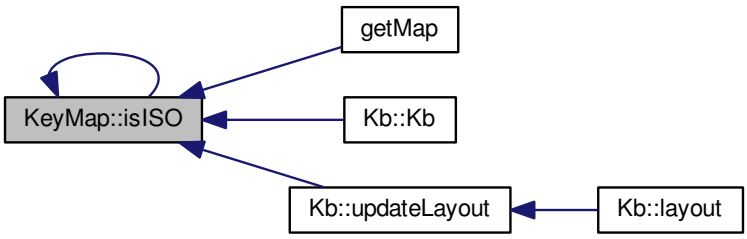
Referenced by getMap(), isISO(), Kb::Kb(), and Kb::updateLayout().

```
95 { return isISO(keyLayout); }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.42.4.14 static bool KeyMap::isJP (Layout *layout*) [inline], [static]

Definition at line 98 of file keymap.h.

References JP.

```
98 { return layout == JP; }
```

12.42.4.15 bool KeyMap::isJP ( ) const [inline]

Definition at line 99 of file keymap.h.

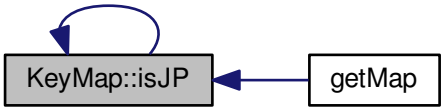
References isJP(), and keyLayout.

Referenced by getMap(), and isJP().

```
99 { return isJP(keyLayout); }
```

Here is the call graph for this function:

Here is the caller graph for this function:



12.42.4.16 static bool KeyMap::isKeyboard (Model *model*) [inline], [static]

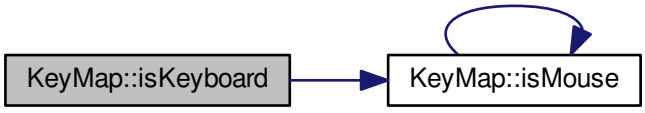
Definition at line 109 of file keymap.h.

References isMouse(), and NO_MODEL.

Referenced by KbBind::isKeyboard().

```
109 { return !isMouse(model) && model != NO_MODEL; }
```

Here is the call graph for this function:



Here is the caller graph for this function:

12.42.4.17 bool KeyMap::isKeyboard ( ) const [inline]

Definition at line 110 of file keymap.h.

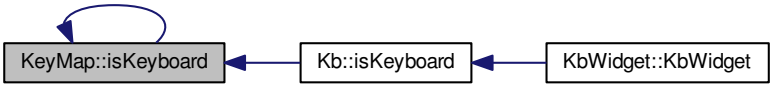
References isKeyboard(), and keyModel.

Referenced by Kb::isKeyboard(), and isKeyboard().

```
110 { return isKeyboard(keyModel); }
```

Here is the call graph for this function:

Here is the caller graph for this function:



12.42.4.18 static bool KeyMap::isMouse (Model *model*) [inline], [static]

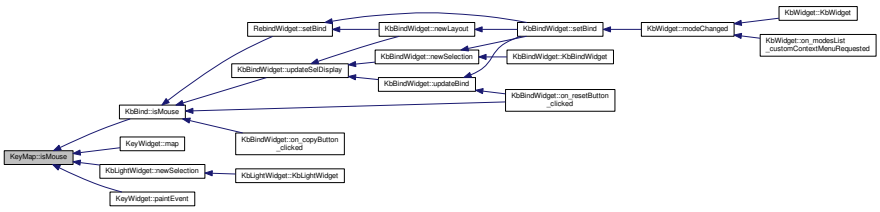
Definition at line 111 of file keymap.h.

References GLAIVE, HARPOON, M65, SABRE, and SCIMITAR.

Referenced by KbBind::isMouse(), KeyWidget::map(), KbLightWidget::newSelection(), and KeyWidget::paintEvent().

```
111 { return model == M65 || model == SABRE || model ==
      SCIMITAR || model == HARPOON || model == GLAIVE; }
```

Here is the caller graph for this function:



12.42.4.19 bool KeyMap::isMouse ( ) const [inline]

Definition at line 112 of file keymap.h.

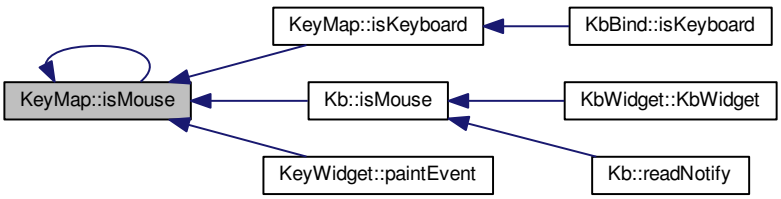
References isMouse(), and keyModel.

Referenced by isKeyboard(), Kb::isMouse(), isMouse(), and KeyWidget::paintEvent().

```
112 { return isMouse(keyModel); }
```

Here is the call graph for this function:

Here is the caller graph for this function:



12.42.4.20 static bool KeyMap::isPTBR (Layout *layout*) [inline], [static]

Definition at line 102 of file keymap.h.

References PT_BR.

```
102 { return layout == PT_BR; }
```

12.42.4.21 bool KeyMap::isPTBR ( ) const [inline]

Definition at line 103 of file keymap.h.

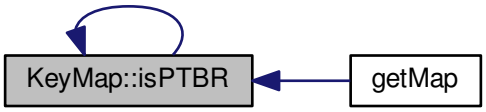
References isPTBR(), and keyLayout.

Referenced by getMap(), and isPTBR().

```
103 { return isPTBR(keyLayout); }
```

Here is the call graph for this function:

Here is the caller graph for this function:



12.42.4.22 Key KeyMap::key (const QString & *name*) const [inline]

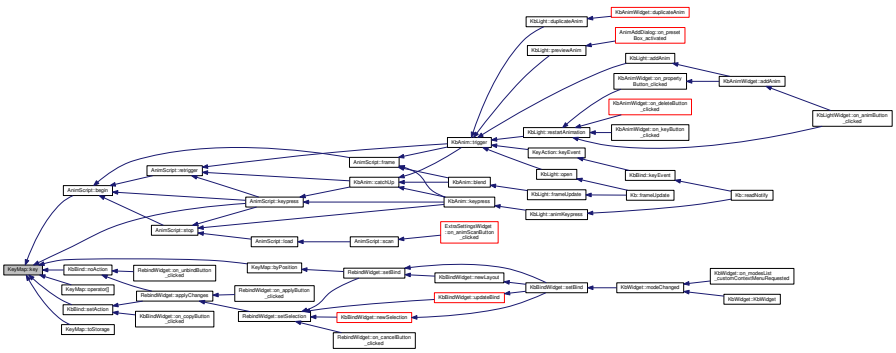
Definition at line 141 of file keymap.h.

References _keys.

Referenced by AnimScript::begin(), byPosition(), AnimScript::keypress(), KbBind::noAction(), operator[](), KbBind::setAction(), and toStorage().

```
141 { Key empty = {0,0,0,0,0,0,0,0,0,0}; return _keys.value(name, empty); }
```

Here is the caller graph for this function:



12.42.4.23 QStringList KeyMap::keys ( ) const [inline]

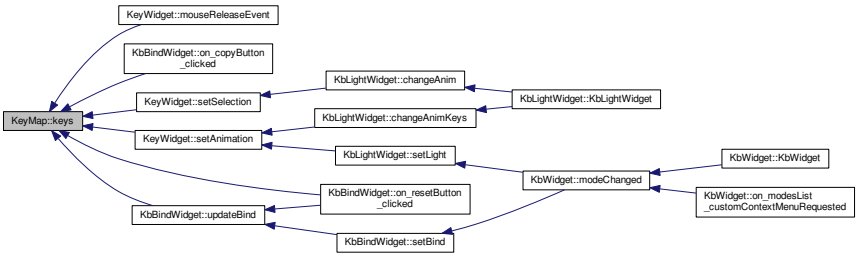
Definition at line 147 of file keymap.h.

References _keys.

Referenced by KeyWidget::mouseReleaseEvent(), KbBindWidget::on_copyButton_clicked(), KbBindWidget::on_resetButton_clicked(), KeyWidget::setAnimation(), KeyWidget::setSelection(), and KbBindWidget::updateBind().

```
147 { return _keys.keys(); }
```

Here is the caller graph for this function:



12.42.4.24 Layout KeyMap::layout ( ) const [inline]

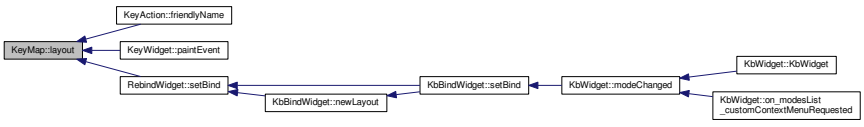
Definition at line 130 of file keymap.h.

References keyLayout.

Referenced by KeyAction::friendlyName(), KeyWidget::paintEvent(), and RebindWidget::setBind().

```
130 { return keyLayout; }
```

Here is the caller graph for this function:



12.42.4.25 QStringList KeyMap::layoutNames ( ) [static]

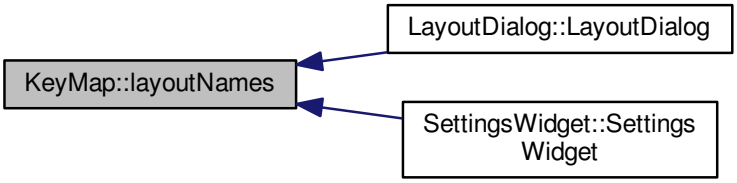
Definition at line 757 of file keymap.cpp.

Referenced by LayoutDialog::LayoutDialog(), and SettingsWidget::SettingsWidget().

```
757                                {
758     return QStringList()
759             << "Danish"
760             << "English (ISO/European)" << "English (ISO/European, Dvorak)"
761             << "English (United Kingdom)" << "English (United Kingdom, Dvorak)"
762             << "English (United States)" << "English (United States, Dvorak)"
763             << "French"
764             << "German"
765             << "Italian"
```

```
766             << "Japanese"
767             << "Norwegian"
768             << "Polish"
769             << "Portuguese (Brazil)"
770             << "Spanish (Latin America)"
771             << "Spanish (Spain)"
772             << "Swedish";
773 }
```

Here is the caller graph for this function:



12.42.4.26 KeyMap::Layout KeyMap::locale ( ) [static]

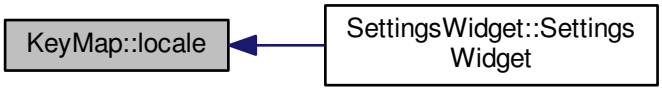
Definition at line 641 of file keymap.cpp.

References DE, DK, ES, FR, GB, IT, JP, MX, NO, PL, PT_BR, SE, and US.

Referenced by SettingsWidget::SettingsWidget().

```
641                         {
642     setlocale(LC_ALL, "");
643     QString loc = setlocale(LC_CTYPE, 0);
644     loc = loc.toLower().replace('_', '-');
645     if(loc.startsWith("dk-"))
646         return KeyMap::DK;
647     else if(loc.startsWith("fr-"))
648         return KeyMap::FR;
649     else if(loc.startsWith("de-"))
650         return KeyMap::DE;
651     else if(loc.startsWith("it-"))
652         return KeyMap::IT;
653     else if(loc.startsWith("ja-"))
654         return KeyMap::JP;
655     else if(loc.startsWith("pl-"))
656         return KeyMap::PL;
657     else if(loc.startsWith("pt-br"))
658         return KeyMap::PT_BR;
659     else if(loc.startsWith("no-"))
660         return KeyMap::NO;
661     else if(loc.startsWith("es-es"))
662         // Spain uses the ES layout
663         return KeyMap::ES;
664     else if(loc.startsWith("es-"))
665         // Other Spanish locales use MX
666         return KeyMap::MX;
667     else if(loc.startsWith("se-"))
668         return KeyMap::SE;
669     else if(loc.startsWith("en-us") || loc.startsWith("en-au") || loc.startsWith("en-ca") || loc.startsWith
    ("en-hk") || loc.startsWith("en-in") || loc.startsWith("en-nz") || loc.startsWith("en-ph") || loc.startsWith
    ("en-sg") || loc.startsWith("en-za"))
670         // Most English locales use US
671         return KeyMap::US;
672     else
673         // Default to GB
674         return KeyMap::GB;
675 }
```

Here is the caller graph for this function:



12.42.4.27 const QHash<QString, Key>& KeyMap::map ( ) const [inline]

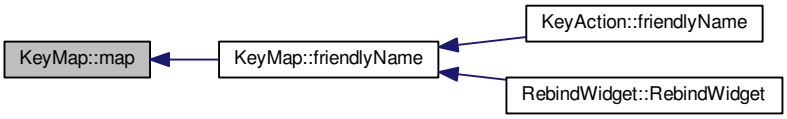
Definition at line 145 of file keymap.h.

References _keys.

Referenced by friendlyName().

```
145 { return _keys; }
```

Here is the caller graph for this function:



12.42.4.28 Model KeyMap::model ( ) const [inline]

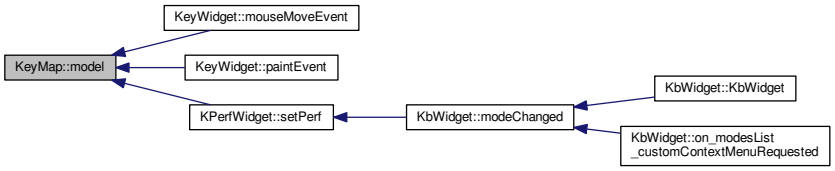
Definition at line 129 of file keymap.h.

References keyModel.

Referenced by KeyWidget::mouseMoveEvent(), KeyWidget::paintEvent(), and KPerfWidget::setPerf().

```
129 { return keyModel; }
```

Here is the caller graph for this function:



12.42.4.29 int KeyMap::modelHeight (Model *model*) [static], [private]

Definition at line 869 of file keymap.cpp.

References GLAIVE, HARPOON, K63, K65, K68, K70, K95, K95_HEIGHT, K95P, K95P_HEIGHT, M65, M65_HEIGHT, SABRE, SCIMITAR, and STRAFE.

```
869                                     {
870     switch(model){
871     case K63:
872     case K65:
873     case K68:
874     case K70:
875     case K95:
876     case STRAFE:
877         return K95_HEIGHT;
878     case K95P:
879         return K95P_HEIGHT;
880     case M65:
881     case SABRE:
882     case SCIMITAR:
883     case HARPOON:
884     case GLAIVE:
885         return M65_HEIGHT;
886     default:
887         return 0;
888     }
889 }
```

12.42.4.30 int KeyMap::modelWidth (Model *model*) [static], [private]

Definition at line 842 of file keymap.cpp.

References GLAIVE, HARPOON, K63, K63_WIDTH, K65, K65_WIDTH, K68, K68_WIDTH, K70, K70_WIDTH, K95, K95_WIDTH, K95P, K95P_WIDTH, KSTRAFE_WIDTH, M65, M65_WIDTH, SABRE, SCIMITAR, and STRAFE.

```
842                                     {
843     switch(model){
844     case K63:
845         return K63_WIDTH;
846     case K65:
847         return K65_WIDTH;
848     case K68:
849         return K68_WIDTH;
850     case K70:
851         return K70_WIDTH;
852     case K95:
853          return K95_WIDTH;
854     case K95P:
855         return K95P_WIDTH;
856     case STRAFE:
857         return KSTRAFE_WIDTH;
858     case M65:
859     case SABRE:
860     case SCIMITAR:
861     case HARPOON:
862     case GLAIVE:
863         return M65_WIDTH;
864     default:
865         return 0;
866     }
867 }
```

12.42.4.31 QString KeyMap::name ( ) const [inline]

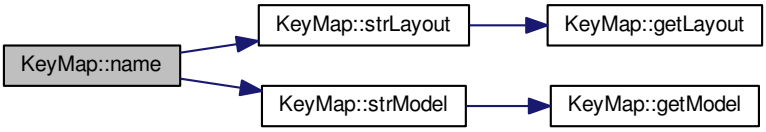
Definition at line 131 of file keymap.h.

References strLayout(), and strModel().

Referenced by KbBind::save(), KbLight::save(), and toStorage().

```
131 { return (strModel() + " " + strLayout()).toUpper(); }
```

Here is the call graph for this function:



Here is the caller graph for this function:

12.42.4.32 KeyMap::operator const QHash< QString, Key > & ( ) const [inline]

Definition at line 146 of file keymap.h.

References _keys.

```
146 { return _keys; }
```

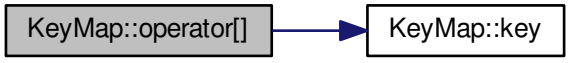
12.42.4.33 Key KeyMap::operator[] (const QString & *name*) const [inline]

Definition at line 142 of file keymap.h.

References key().

```
142 { return key(name); }
```

Here is the call graph for this function:



12.42.4.34 QList<Key> KeyMap::positions ( ) const [inline]

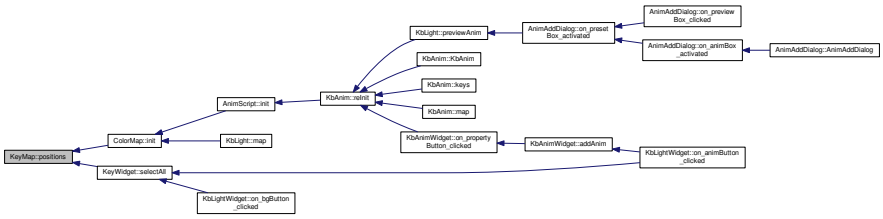
Definition at line 148 of file keymap.h.

References _keys.

Referenced by ColorMap::init(), and KeyWidget::selectAll().

```
148 { return _keys.values(); }
```

Here is the caller graph for this function:



12.42.4.35 QString KeyMap::strLayout ( ) const [inline]

Definition at line 122 of file keymap.h.

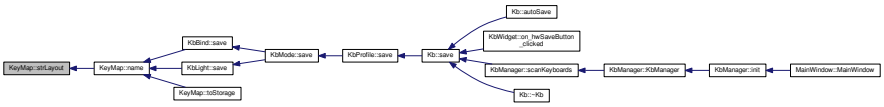
References getLayout(), and keyLayout.

Referenced by name().

```
122 { return getLayout(keyLayout); }
```

Here is the call graph for this function:

Here is the caller graph for this function:



12.42.4.36 QString KeyMap::strModel ( ) const [inline]

Definition at line 126 of file keymap.h.

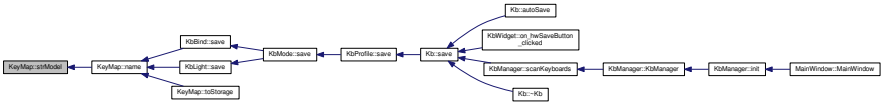
References getModel(), and keyModel.

Referenced by name().

```
126 { return getModel(keyModel); }
```

Here is the call graph for this function:

Here is the caller graph for this function:



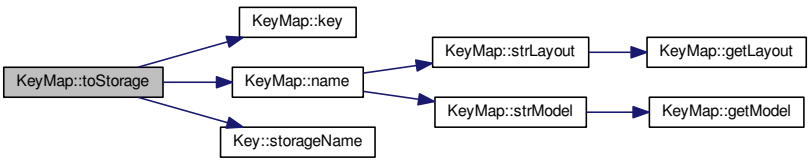
12.42.4.37 QString KeyMap::toStorage (const QString & *name*) [inline]

Definition at line 150 of file keymap.h.

References key(), name(), and Key::storageName().

```
150 { const char* storage = key(name).storageName(); if(!storage) return
      name; return storage; }
```

Here is the call graph for this function:



12.42.4.38 uint KeyMap::width ( ) const [inline]

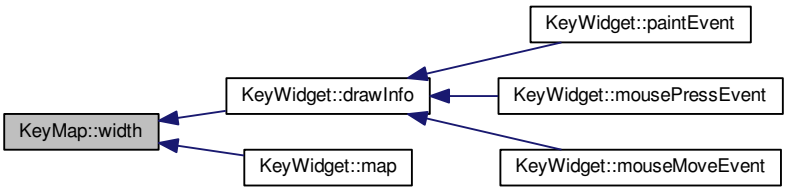
Definition at line 136 of file keymap.h.

References keyWidth.

Referenced by KeyWidget::drawInfo(), and KeyWidget::map().

```
136 { return keyWidth; }
```

Here is the caller graph for this function:



12.42.5 Field Documentation

12.42.5.1 QHash<QString, Key> KeyMap::_keys [private]

Definition at line 163 of file keymap.h.

Referenced by contains(), count(), key(), keys(), map(), operator const QHash< QString, Key > &(), and positions().

12.42.5.2 short KeyMap::keyHeight [private]

Definition at line 164 of file keymap.h.

Referenced by height().

12.42.5.3 Layout KeyMap::keyLayout [private]

Definition at line 166 of file keymap.h.

Referenced by isISO(), isJP(), isPTBR(), layout(), and strLayout().

12.42.5.4 Model KeyMap::keyModel [private]

Definition at line 165 of file keymap.h.

Referenced by isKeyboard(), isMouse(), model(), and strModel().

12.42.5.5 short KeyMap::keyWidth [private]

Definition at line 164 of file keymap.h.

Referenced by width().

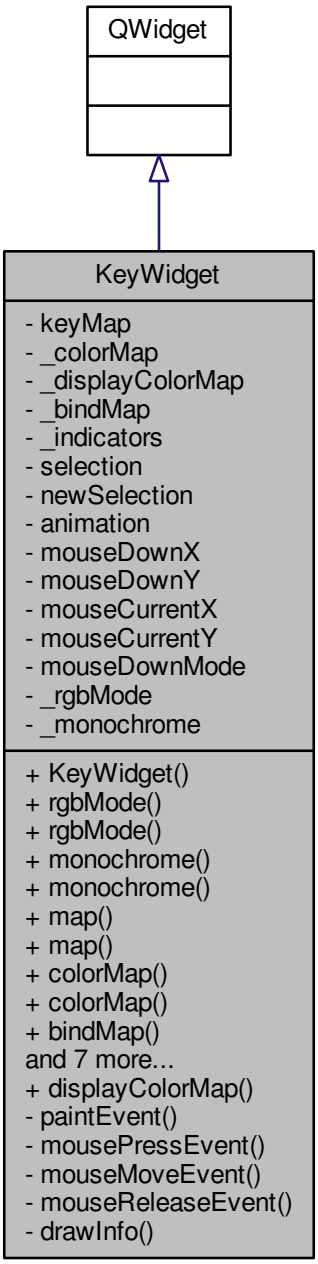
The documentation for this class was generated from the following files:

- src/ckb/keymap.h
- src/ckb/keymap.cpp

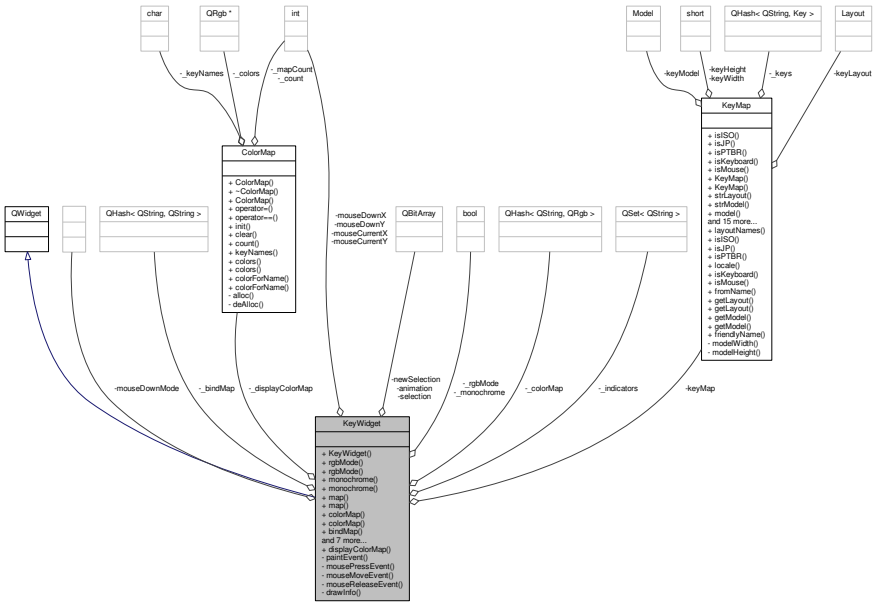
12.43 KeyWidget Class Reference

`#include <src/ckb/keywidget.h>`

Inheritance diagram for KeyWidget:



Collaboration diagram for KeyWidget:



Public Types

- typedef QHash< QString, QString > BindMap

Public Slots

- void displayColorMap (const ColorMap &newDisplayMap, const QSet< QString > &indicators=QSet< QString >())

Signals

- void selectionChanged (QStringList selected)
- void sidelightToggled ()

Public Member Functions

- KeyWidget (QWidget *parent=0, bool rgbMode=true)
- bool rgbMode ()
- void rgbMode (bool newRgbMode)
- bool monochrome ()
- void monochrome (bool newMono)
- const KeyMap & map () const
- void map (const KeyMap &newMap)
- const QColorMap & colorMap () const
- void colorMap (const QColorMap &newColorMap)
- const BindMap & bindMap () const
- void bindMap (const BindMap &newBindMap)
- void setSelection (const QStringList &keys)
- void selectAll ()

- void clearSelection ()
- void setAnimation (const QStringList &keys)
- void setAnimationToSelection ()
- void clearAnimation ()

Private Types

- enum {
  NONE, SET, ADD, SUBTRACT,
  TOGGLE }

Private Member Functions

- void paintEvent (QPaintEvent ∗)
- void mousePressEvent (QMouseEvent ∗event)
- void mouseMoveEvent (QMouseEvent ∗event)
- void mouseReleaseEvent (QMouseEvent ∗event)
- void drawInfo (float &scale, float &offsetX, float &offsetY, int ratio=1)

Private Attributes

- KeyMap keyMap
- QColorMap _colorMap
- ColorMap _displayColorMap
- BindMap _bindMap
- QSet< QString > _indicators
- QBitArray selection
- QBitArray newSelection
- QBitArray animation
- int mouseDownX
- int mouseDownY
- int mouseCurrentX
- int mouseCurrentY
- enum KeyWidget:: { ... } mouseDownMode
- bool _rgbMode
- bool _monochrome

12.43.1 Detailed Description

Definition at line 11 of file keywidget.h.

12.43.2 Member Typedef Documentation

12.43.2.1 typedef QHash<QString, QString> KeyWidget::BindMap

Definition at line 30 of file keywidget.h.

12.43.3 Member Enumeration Documentation

12.43.3.1 anonymous enum [private]

Enumerator

NONE

SET

ADD

SUBTRACT

TOGGLE

Definition at line 66 of file keywidget.h.

```
66          {
67         NONE,
68         SET,
69         ADD,
70         SUBTRACT,
71         TOGGLE,
72     } mouseDownMode;
```

12.43.4 Constructor & Destructor Documentation

12.43.4.1 KeyWidget::KeyWidget (QWidget * *parent* = 0, bool *rgbMode* = true) [explicit]

Definition at line 17 of file keywidget.cpp.

```
17                                                  :
18     QWidget(parent), mouseDownX(-1), mouseDownY(-1),
       mouseCurrentX(-1), mouseCurrentY(-1), mouseDownMode(
       NONE), _rgbMode(rgbMode), _monochrome(false)
19 {
20     setMouseTracking(true);
21     setAutoFillBackground(false);
22 }
```

12.43.5 Member Function Documentation

12.43.5.1 const BindMap& KeyWidget::bindMap ( ) const [inline]

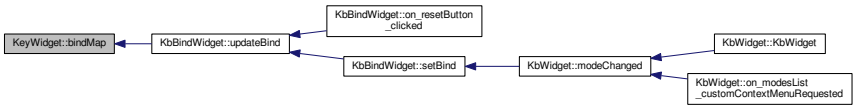
Definition at line 31 of file keywidget.h.

References _bindMap.

Referenced by KbBindWidget::updateBind().

```
31 { return _bindMap; }
```

Here is the caller graph for this function:



12.43.5.2 void KeyWidget::bindMap (const BindMap & *newBindMap*)

Definition at line 65 of file keywidget.cpp.

References _bindMap.

```
65                                                  {
66     _bindMap = newBindMap;
67     update();
68 }
```

12.43.5.3 void KeyWidget::clearAnimation ( )

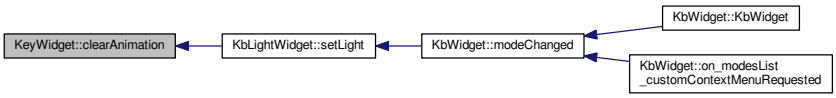
Definition at line 644 of file keywidget.cpp.

References animation.

Referenced by KbLightWidget::setLight().

```
644                                 {
645     animation.fill(false);
646     update();
647 }
```

Here is the caller graph for this function:



12.43.5.4 void KeyWidget::clearSelection ( )

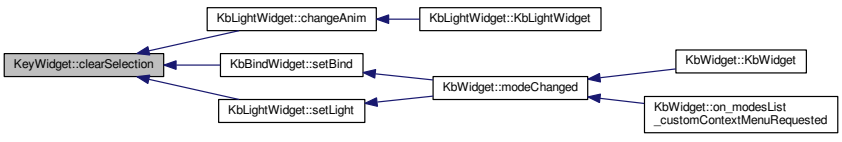
Definition at line 617 of file keywidget.cpp.

References mouseDownMode, newSelection, NONE, selection, and selectionChanged().

Referenced by KbLightWidget::changeAnim(), KbBindWidget::setBind(), and KbLightWidget::setLight().

```
617                                 {
618     selection.fill(false);
619     newSelection.fill(false);
620     mouseDownMode = NONE;
621     update();
622     emit selectionChanged(QStringList());
623 }
```

Here is the caller graph for this function:



12.43.5.5 const QColorMap& KeyWidget::colorMap ( ) const [inline]

Definition at line 27 of file keywidget.h.

References _colorMap.

Referenced by KbLightWidget::changeColor(), KbLightWidget::toggleSidelight(), and KbLightWidget::updateLight().

```
27 { return _colorMap; }
```

Here is the caller graph for this function:

12.43.5.6 void KeyWidget::colorMap (const QColorMap & *newColorMap*)

Definition at line 52 of file keywidget.cpp.

References _colorMap.

```
52                                                  {
53     _colorMap = newColorMap;
54     update();
55 }
```

12.43.5.7 void KeyWidget::displayColorMap (const ColorMap & *newDisplayMap*, const QSet< QString > & *indicators* = QSet<QString>()) [slot]

Definition at line 57 of file keywidget.cpp.

References _displayColorMap, and _indicators.

Referenced by KbLightWidget::on_showAnimBox_clicked().

```
57                                                                                  {
58     if(!isVisible())
59         return;
60     _displayColorMap = newDisplayMap;
61     _indicators = indicators;
62     update();
63 }
```

Here is the caller graph for this function:

12.43.5.8 void KeyWidget::drawInfo (float & *scale*, float & *offsetX*, float & *offsetY*, int *ratio* = 1) [private]

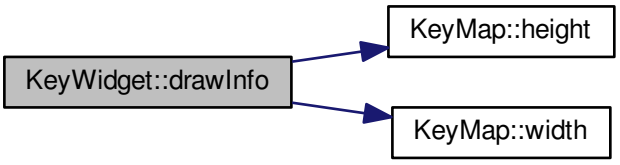
Definition at line 43 of file keywidget.cpp.

References KeyMap::height(), KEY_SIZE, keyMap, and KeyMap::width().

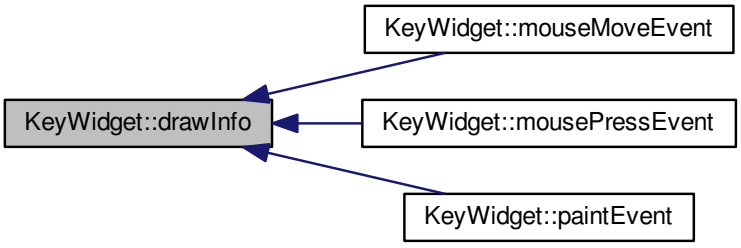
Referenced by mouseMoveEvent(), mousePressEvent(), and paintEvent().

```
43                                                                                  {
44     int w = width() * ratio, h = height() * ratio;
45     float xScale = (float)w / (keyMap.width() + KEY_SIZE);
46     float yScale = (float)h / (keyMap.height() + KEY_SIZE);
47     scale = fmin(xScale, yScale);
48     offsetX = (w / scale - keyMap.width()) / 2.f;
49     offsetY = (h / scale - keyMap.height()) / 2.f;
50 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.43.5.9 const KeyMap& KeyWidget::map ( ) const [inline]

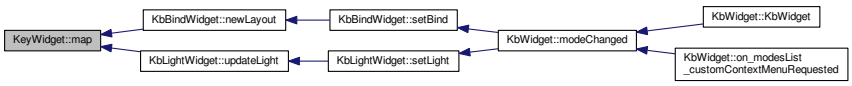
Definition at line 24 of file keywidget.h.

References keyMap.

Referenced by KbBindWidget::newLayout(), and KbLightWidget::updateLight().

```
24 { return keyMap; }
```

Here is the caller graph for this function:



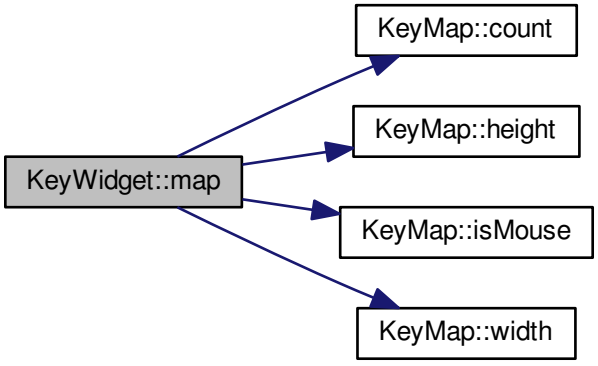
12.43.5.10 void KeyWidget::map (const KeyMap & *newMap*)

Definition at line 24 of file keywidget.cpp.

References animation, KeyMap::count(), KeyMap::height(), KeyMap::isMouse(), KEY_SIZE, keyMap, newSelection, selection, and KeyMap::width().

```
24                                      {
25     keyMap = newMap;
26     selection = QBitArray(keyMap.count());
27     newSelection = QBitArray(keyMap.count());
28     animation = QBitArray(keyMap.count());
29     int width, height;
30     if(keyMap.isMouse()){
31         width = (keyMap.width() + KEY_SIZE) * 2.6;
32         height = (keyMap.height() + KEY_SIZE) * 2.6;
33     } else {
34         width = (keyMap.width() + KEY_SIZE) * 2.3;
35         height = (keyMap.height() + KEY_SIZE) * 2.3;
36     }
37     if(width < 500)
38         width = 500;
39     setFixedSize(width, height);
40     update();
41 }
```

Here is the call graph for this function:



12.43.5.11 bool KeyWidget::monochrome ( ) [inline]

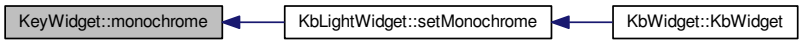
Definition at line 20 of file keywidget.h.

References _monochrome.

Referenced by KbLightWidget::setMonochrome().

```
20 { return _monochrome; }
```

Here is the caller graph for this function:



12.43.5.12 void KeyWidget::monochrome (bool *newMono*) [inline]

Definition at line 21 of file keywidget.h.

References _monochrome.

```
21 { _monochrome = newMono; update(); }
```

12.43.5.13 void KeyWidget::mouseMoveEvent (QMouseEvent * *event*) [private]

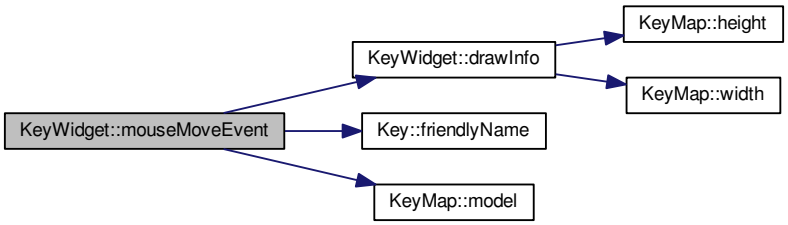
Definition at line 487 of file keywidget.cpp.

References _indicators, _rgbMode, drawInfo(), Key::friendlyName(), Key::hasLed, Key::hasScan, Key::height, keyMap, KeyMap::model(), mouseCurrentX, mouseCurrentY, mouseDownMode, mouseDownX, mouseDownY, Key::name, newSelection, NONE, KeyMap::STRAFE, Key::width, Key::x, and Key::y.

```
487                                                 {
488     event->accept();
489     QString tooltip;
490
491     // Find selection rectangle
492     mouseCurrentX = event->x();
493     mouseCurrentY = event->y();
494     float scale, offX, offY;
495     drawInfo(scale, offX, offY);
496     float mx = mouseCurrentX / scale - offX, my = mouseCurrentY / scale - offY;
497     float mx1, mx2, my1, my2;
498     if(mouseCurrentX >= mouseDownX){
499         mx1 = mouseDownX / scale - offX;
500         mx2 = mouseCurrentX / scale - offX;
501     } else {
502         mx1 = mouseCurrentX / scale - offX;
503         mx2 = mouseDownX / scale - offX;
504     }
505     if(mouseCurrentY >= mouseDownY){
506         my1 = mouseDownY / scale - offY;
507         my2 = mouseCurrentY / scale - offY;
508     } else {
509         my1 = mouseCurrentY / scale - offY;
510         my2 = mouseDownY / scale - offY;
511     }
512     // Clear new selection
513     if(mouseDownMode != NONE)
514         newSelection.fill(false);
515     // See if the event hit any keys
516     QHashIterator<QString, Key> k(keyMap);
517     uint i = -1;
518     while(k.hasNext()){
519         k.next();
520         i++;
521         const Key& key = k.value();
522         if((_rgbMode && !key.hasLed)
523                 || (!_rgbMode && !key.hasScan))
524             continue;
525         // Update tooltip with the mouse hover (if any), even if it's not selectable
526         if(fabs(key.x - mx) <= key.width / 2.f - 1.f && fabs(key.y - my) <= key.
```

```
    height / 2.f - 1.f
527                 && tooltip.isEmpty())
528             tooltip = key.friendlyName(false);
529         // on STRAFE Sidelights and indicators can't be assigned color the way other keys are colored
530         if(keyMap.model() == KeyMap::STRAFE && (!strcmp(key.
    name, "lsidel") || !strcmp(key.name, "rsidel") || _indicators.contains(key.
    name))) // FIX: _indicators check fails whenever _indicators is empty because "show animated" is
     unchecked
531             continue;
532         float kx1 = key.x - key.width / 2.f + 1.f;
533         float ky1 = key.y - key.height / 2.f + 1.f;
534         float kx2 = kx1 + key.width - 2.f;
535         float ky2 = ky1 + key.height - 2.f;
536         // If they overlap, add the key to the selection
537         if(!(mx1 >= kx2 || kx1 >= mx2)
538                 && !(my1 >= ky2 || ky1 >= my2)
539                 && mouseDownMode != NONE)
540             newSelection.setBit(i);
541     }
542
543     if(mouseDownMode != NONE)
544         update();
545     setToolTip(tooltip);
546 }
```

Here is the call graph for this function:



12.43.5.14 void KeyWidget::mousePressEvent (QMouseEvent * *event*) [private]

Definition at line 455 of file keywidget.cpp.

References _rgbMode, ADD, drawInfo(), Key::hasLed, Key::hasScan, Key::height, keyMap, mouseCurrentX, mouseCurrentY, mouseDownMode, mouseDownX, mouseDownY, Key::name, newSelection, SET, sidelightToggled(), SUBTRACT, TOGGLE, Key::width, Key::x, and Key::y.

```
455                                                 {
456     event->accept();
457     mouseDownMode = (event->modifiers() & Qt::AltModifier) ?
    SUBTRACT : (event->modifiers() & Qt::ShiftModifier) ? ADD : (event->modifiers() &
    Qt::ControlModifier) ? TOGGLE : SET;
458     mouseDownX = mouseCurrentX = event->x();
459     mouseDownY = mouseCurrentY = event->y();
460     // See if the event hit a key
461     float scale, offX, offY;
462     drawInfo(scale, offX, offY);
463     float mx = mouseCurrentX / scale - offX, my = mouseCurrentY / scale - offY;
464     QHashIterator<QString, Key> k(keyMap);
465     uint i = -1;
466     while(k.hasNext()){
467         k.next();
468         i++;
469         const Key& key = k.value();
470         if((_rgbMode && !key.hasLed)
471                 || (!_rgbMode && !key.hasScan))
472             continue;
473         if(fabs(key.x - mx) <= key.width / 2.f - 1.f && fabs(key.y - my) <= key.
    height / 2.f - 1.f){
474             // Sidelights can't have a color, but they can be toggled
475             if(!strcmp(key.name, "lsidel") || !strcmp(key.name, "rsidel")){
```

```
476                 emit sidelightToggled(); // get the kblightwidget to record it
477                 update();
478                 break;
479             }
480             newSelection.setBit(i);
481             update();
482             break;
483         }
484     }
485 }
```

Here is the call graph for this function:

12.43.5.15 void KeyWidget::mouseReleaseEvent (QMouseEvent * *event*) [private]

Definition at line 548 of file keywidget.cpp.

References ADD, keyMap, KeyMap::keys(), mouseDownMode, newSelection, NONE, selection, selectionChanged(), SET, SUBTRACT, and TOGGLE.

```
548                                                 {
549     event->accept();
550     if(mouseDownMode == NONE)
551         return;
552     // Apply the new selection
553     switch(mouseDownMode){
554     case SET:
555         selection = newSelection;
556         break;
557     case ADD:
558         selection |= newSelection;
559         break;
560     case SUBTRACT:
561         selection &= ~newSelection;
562         break;
563     case TOGGLE:
564         selection ^= newSelection;
565         break;
566     default:;
567     }
568     // Clear mousedown state.
569     newSelection.fill(false);
570     mouseDownMode = NONE;
571     // Emit signal with the names of the keys
572     QStringList selectedNames;
573     uint i = 0;
574     foreach(const QString& key, keyMap.keys()){
575         if(selection.testBit(i++))
576             selectedNames << key;
577     }
578     emit selectionChanged(selectedNames);
579     update();
580 }
```

Here is the call graph for this function:

12.43.5.16 void KeyWidget::paintEvent (QPaintEvent *) [private]

Definition at line 70 of file keywidget.cpp.

References _bindMap, _colorMap, _displayColorMap, _indicators, _monochrome, _rgbMode, ADD, animation, ColorMap::colorForName(), KbBind::defaultAction(), drawInfo(), KeyMap::EU, KeyMap::EU_DVORAK, Key::friendlyName(), KeyMap::GLAIVE, glaiveOverlay, KbBind::globalRemap(), KeyMap::HARPOON, harpOverlay, Key::hasLed, Key::hasScan, Key::height, KeyMap::isMouse(), KeyMap::K65, KeyMap::K95P, keyMap, KeyMap::layout(), KeyMap::M65, m65Overlay, KeyMap::model(), monoRgb(), mouseCurrentX, mouseCurrentY, mouseDownMode, mouseDownX, mouseDownY, Key::name, newSelection, NONE, sabOverlay, KeyMap::SABRE, KeyMap::SCIMITAR, scimOverlay, selection, SET, KeyMap::STRAFE, SUBTRACT, TOGGLE, Key::width, Key::x, and Key::y.

```
70                                      {
71     const QColor bgColor(68, 64, 64);
72     const QColor keyColor(112, 110, 110);
73     const QColor sniperColor(130, 90, 90);
74     const QColor thumbColor(34, 32, 32);
75     const QColor transparentColor(0, 0, 0, 0);
76     const QColor highlightColor(136, 176, 240);
77     const QColor highlightAnimColor(136, 200, 240);
78     const QColor animColor(112, 200, 110);
79
80     // Determine which keys to highlight
81     QBitArray highlight;
82     switch(mouseDownMode){
83     case SET:
84         highlight = newSelection;
85         break;
86     case ADD:
87         highlight = selection | newSelection;
88         break;
89     case SUBTRACT:
90         highlight = selection & ~newSelection;
91         break;
92     case TOGGLE:
93         highlight = selection ^ newSelection;
94         break;
95     default:
96         highlight = selection;
97     }
98
99     QPainter painter(this);
100 #if QT_VERSION >= QT_VERSION_CHECK(5, 3, 0)
101     int ratio = painter.device()->devicePixelRatio();
102 #else
103     int ratio = 1;
104 #endif
105     int wWidth = width(), wHeight = height();
106     KeyMap::Model model = keyMap.model();
107     KeyMap::Layout layout = keyMap.layout();
108     float scale, offX, offY;
109     drawInfo(scale, offX, offY, ratio);
110     // Draw background
111     painter.setPen(Qt::NoPen);
112     painter.setRenderHint(QPainter::Antialiasing, true);
113
114     if(keyMap.isMouse()){
115         // Draw mouse overlays
116         const QImage* overlay = 0;
117         float xpos = 0.f, ypos = 0.f;
```

```
118         if(model == KeyMap::M65){
119             if(!m65Overlay)
120                 m65Overlay = new QImage(":/img/overlay_m65.png");
121             overlay = m65Overlay;
122             xpos = 2.f;
123             ypos = -2.f;
124         } else if(model == KeyMap::SABRE){
125             if(!sabOverlay)
126                 sabOverlay = new QImage(":/img/overlay_sabre.png");
127             overlay = sabOverlay;
128             xpos = 1.f;
129             ypos = -2.f;
130         } else if(model == KeyMap::SCIMITAR){
131             if(!scimOverlay)
132                 scimOverlay = new QImage(":/img/overlay_scimitar.png");
133             overlay = scimOverlay;
134             xpos = 3.5f;
135             ypos = -2.f;
136         } else if(model == KeyMap::HARPOON){
137             if(!harpOverlay)
138                 harpOverlay = new QImage(":/img/overlay_harpoon.png");
139             overlay = harpOverlay;
140             xpos = 3.5f;
141             ypos = -2.f;
142         } else if(model == KeyMap::GLAIVE){
143             if(!glaiveOverlay)
144                 glaiveOverlay = new QImage(":/img/overlay_glaive.png");
145             overlay = glaiveOverlay;
146             xpos = 3.5f;
147             ypos = -2.f;
148         }
149         if(overlay){
150             painter.setBrush(palette().brush(QPalette::Window));
151             painter.drawRect(0, 0, width(), height());
152             float oXScale = scale / 9.f, oYScale = scale / 9.f;             // The overlay has a resolution
     of 9px per keymap unit
153             float x = (xpos + offX) * scale, y = (ypos + offY) * scale;
154             int w = overlay->width() * oXScale, h = overlay->height() * oYScale;
155             // We need to transform the image with QImage::scaled() because painter.drawImage() will
     butcher it, even with smoothing enabled
156             // However, the width/height need to be rounded to integers
157             int iW = round(w), iH = round(h);
158             painter.drawImage(QRectF((x - (iW - w) / 2.f) / ratio, (y - (iH - h) / 2.f) / ratio, iW / ratio
    , iH / ratio), overlay->scaled(iW, iH, Qt::IgnoreAspectRatio, Qt::SmoothTransformation));
159         }
160     } else {
161         // Otherwise, draw a solid background
162         painter.setBrush(QBrush(bgColor));
163         painter.drawRect(0, 0, width(), height());
164     }
165
166     // Draw mouse highlight (if any)
167     if(mouseDownMode != NONE && (mouseDownX !=
    mouseCurrentX || mouseDownY != mouseCurrentY)){
168         int x1 = (mouseDownX > mouseCurrentX) ?
    mouseCurrentX : mouseDownX;
169         int x2 = (mouseDownX > mouseCurrentX) ?
    mouseDownX : mouseCurrentX;
170         int y1 = (mouseDownY > mouseCurrentY) ?
    mouseCurrentY : mouseDownY;
171         int y2 = (mouseDownY > mouseCurrentY) ?
    mouseDownY : mouseCurrentY;
172         painter.setPen(QPen(highlightColor, 0.5));
173         QColor bColor = highlightColor;
174         bColor.setAlpha(128);
175         painter.setBrush(QBrush(bColor));
176         painter.drawRect(x1, y1, x2 - x1, y2 - y1);
177     }
178
179     // Draw key backgrounds on a separate pixmap so that a drop shadow can be applied to them.
180     QPixmap keyBG(wWidth * ratio, wHeight * ratio);
181     keyBG.fill(QColor(0, 0, 0, 0));
182     QPainter bgPainter(&keyBG);
183     bgPainter.setRenderHints(QPainter::Antialiasing | QPainter::TextAntialiasing);
184     bgPainter.setPen(Qt::NoPen);
185     QHashIterator<QString, Key> k(keyMap);
186     uint i = -1;
187     while(k.hasNext()){
188         k.next();
189         i++;
190         const Key& key = k.value();
191         float x = key.x + offX - key.width / 2.f + 1.f;
192         float y = key.y + offY - key.height / 2.f + 1.f;
193         float w = key.width - 2.f;
194         float h = key.height - 2.f;
195         // In RGB mode, ignore keys without LEDs
196         if((_rgbMode && !key.hasLed)
```

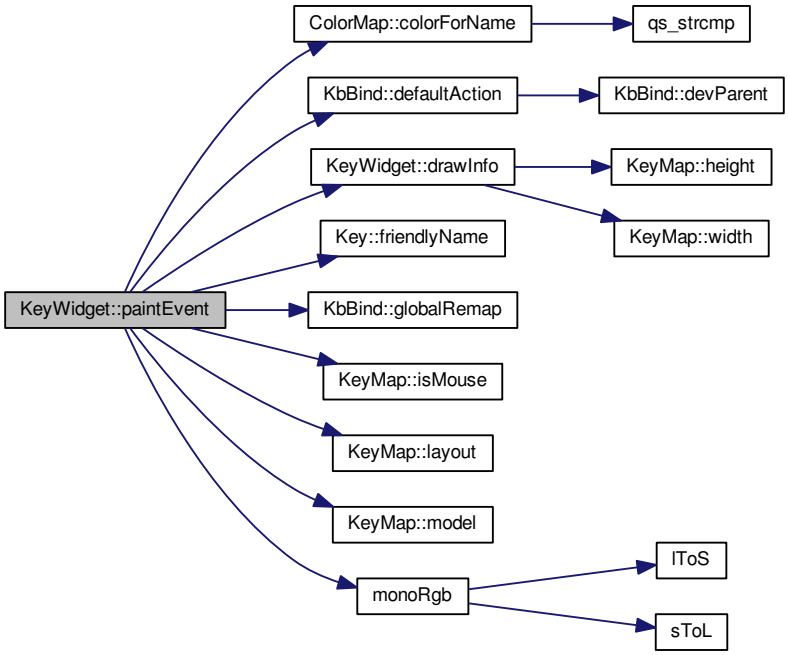
```
197                 || (!_rgbMode && !key.hasScan))
198             continue;
199         // Set color based on key highlight
200         bgPainter.setOpacity(1.);
201         if(highlight.testBit(i)){
202             if(animation.testBit(i))
203                 bgPainter.setBrush(QBrush(highlightAnimColor));
204             else
205                 bgPainter.setBrush(QBrush(highlightColor));
206         } else if(animation.testBit(i)){
207             bgPainter.setBrush(QBrush(animColor));
208         } else {
209             if(!strcmp(key.name, "sniper"))
210                 // Sniper key uses a reddish base color instead of the usual grey
211                 bgPainter.setBrush(QBrush(sniperColor));
212             else if(model == KeyMap::SCIMITAR && !strncmp(key.
    name, "thumb", 5) && strcmp(key.name, "thumb"))
213                 // Thumbgrid keys use a black color
214                 bgPainter.setBrush(QBrush(thumbColor));
215             else if(!strcmp(key.name, "lsidel") || !strcmp(key.name, "rsidel") || !strcmp(key.
    name, "logo"))
216                 // Strafe side lights have different background
217                 bgPainter.setBrush(QBrush(transparentColor));
218             else {
219                 bgPainter.setBrush(QBrush(keyColor));
220                 if(KeyMap::isMouse(model))
221                     bgPainter.setOpacity(0.7);
222             }
223         }
224         if((model != KeyMap::STRAFE && model != KeyMap::K95P) && (!strcmp(key.
    name, "mr") || !strcmp(key.name, "m1") || !strcmp(key.name, "m2") || !strcmp(key.
    name, "m3")
225                 || !strcmp(key.name, "light") || !strcmp(key.name, "lock") || (model ==
    KeyMap::K65 && !strcmp(key.name, "mute")))){
226             // Not all devices have circular buttons
227             x += w / 8.f;
228             y += h / 8.f;
229             w *= 0.75f;
230             h *= 0.75f;
231             bgPainter.drawEllipse(QRectF(x * scale, y * scale, w * scale, h * scale));
232         } else {
233             if(!strcmp(key.name, "enter")){
234                 if(key.height == 24){
235                     // ISO enter key isn't rectangular
236                     y = key.y + 1.f;
237                     h = 10.f;
238                     bgPainter.drawRect(QRectF((x + w - 13.f) * scale, y * scale, 13.f * scale, 22.f * scale
    ));
239                 } else {
240                     // US enter key isn't perfectly centered, needs an extra pixel on the left to appear
     correctly
241                     x -= 1.f;
242                     w += 1.f;
243                 }
244             } else if(!strcmp(key.name, "rshift") || !strcmp(key.name, "stop")){
245                 // A few other keys also need extra pixels
246                 x -= 1.f;
247                 w += 1.f;
248             } else if(!strcmp(key.name, "caps") || !strcmp(key.name, "lshift") || !strcmp(key.
    name, "next")){
249                 w += 1.f;
250             }
251             bgPainter.drawRect(QRectF(x * scale, y * scale, w * scale, h * scale));
252         }
253     }
254
255     // Render the key decorations (RGB -> light circles, binding -> key names) on yet another layer
256     QPixmap decoration(wWidth * ratio, wHeight * ratio);
257     decoration.fill(QColor(0, 0, 0, 0));
258     QPainter decPainter(&decoration);
259     decPainter.setRenderHints(QPainter::Antialiasing | QPainter::TextAntialiasing);
260     if(_rgbMode){
261         // Draw key colors (RGB mode)
262         QHashIterator<QString, Key> k(keyMap);
263         uint i = -1;
264         while(k.hasNext()){
265             k.next();
266             i++;
267             const Key& key = k.value();
268             if(!key.hasLed)
269                 continue;
270             float x = key.x + offX - 1.8f;
271             float y = key.y + offY - 1.8f;
272             float w = 3.6f;
273             float h = 3.6f;
274             // Display a white circle around regular keys, red circle around indicators
275             if(_indicators.contains(key.name))
```

```
276                 decPainter.setPen(QPen(QColor(255, 248, 136), 1.5));
277             else
278                 decPainter.setPen(QPen(QColor(255, 255, 255), 1.5));
279             QRgb color;
280             const QRgb* inDisplay = _displayColorMap.
    colorForName(key.name);
281             if(inDisplay)
282                 // Color in display map? Grab it from there
283                 // (monochrome conversion not necessary as this would have been done by the animation)
284                 color = *inDisplay;
285             else {
286                 // Otherwise, read from base map
287                 color = _colorMap.value(k.key());
288                 if(_monochrome)
289                     color = monoRgb(qRed(color), qGreen(color), qBlue(color));
290             }
291             decPainter.setBrush(QBrush(color));
292             if (model == KeyMap::STRAFE) { // STRAFE custom design and special keys
293                 float kx = key.x + offX - key.width / 2.f + 1.f;
294                 float ky = key.y + offY - key.height / 2.f + 1.f;
295                 float kw = key.width - 2.f;
296                 float kh = key.height - 2.f;
297                 decPainter.setPen(QPen(QColor(255, 255, 255), 1.2)); // less invasive outline to show the
    key color better
298                 if(!strcmp(key.name, "logo")) { // stylized logo
299                     float lx = key.x + offX - key.width / 2.f + 2.f;
300                     float ly = key.y + offY - key.height / 2.f + 2.f;
301                     float lw = key.width - 4.f;
302                     float lh = key.height - 4.f;
303                     QPainterPath logo;
304                     logo.moveTo(lx*scale,(ly+lh)*scale);
305                     logo.quadTo((lx+2.f)*scale,(ly+lh/2.f)*scale,lx*scale,ly*scale);
306                     logo.quadTo((lx+lw)*scale,ly*scale,(lx+lw)*scale,(ly+lh)*scale);
307                     logo.quadTo((lx+lw/2.f)*scale,(ly+lh-4.f)*scale,lx*scale,(ly+lh)*scale);
308                     decPainter.drawPath(logo);
309                     //decPainter.setPen(QPen(Qt::green, 1.2)); //QColor(125,125,125)
310                     //decPainter.drawRect(QRectF(lx * scale, ly * scale, lw * scale, lh * scale)); // don't
    really know why the 12 and 24 make it work here, but they do
311                 } else if(!strcmp(key.name, "lsidel") || !strcmp(key.name, "rsidel")) { // Strafe
    side lights (toggle lights with no animation)
312                     QRadialGradient gradient(QPointF(wWidth/2.f * ratio, wHeight/2.f * ratio), wWidth/2.f *
    ratio);//,QPointF(10, 5));
313                     gradient.setColorAt(0, color);
314                     gradient.setColorAt(0.9, color); // bring up intensity
315                     gradient.setColorAt(1, bgColor);
316                     decPainter.setBrush(QBrush(gradient));
317                     decPainter.setPen(QPen(keyColor, 1.2)); //QColor(125,125,125)
318                     decPainter.drawRect(QRectF(kx * scale, ky * scale - 12 , kw * scale, kh * scale+24));
    // don't really know why the 12 and 24 make it work here, but they do
319                 } else if(_indicators.contains(key.name)) { // FIX: This check fails
    whenever _indicators is empty because "show animated" is unchecked
320                     decPainter.setPen(QPen(QColor(0,0,0,0), 1));    // no outline for the indicators, you
    can't change their color the standard way
321                     decPainter.drawRect(QRectF((kx+2.f) * scale, (ky+2.f) * scale, (kw-4.f) * scale, (kh-4.
    f) * scale)); // square indicators
322                 } else //everything else is a circle, just a tad bigger to show the key color better
323                     decPainter.drawEllipse(QRectF((x-1.f) * scale, (y-1.f) * scale, (w+2.f) * scale, (h+2.f
    ) * scale));
324             } else
325                 decPainter.drawEllipse(QRectF(x * scale, y * scale, w * scale, h * scale));
326         }
327       } else {
328         // Draw key names
329         decPainter.setBrush(Qt::NoBrush);
330         QFont font = painter.font();
331         font.setBold(true);
332         font.setPixelSize(5.25f * scale);
333         QFont font0 = font;
334         QHashIterator<QString, Key> k(keyMap);
335         uint i = -1;
336         while(k.hasNext()){
337             k.next();
338             i++;
339             const Key& key = k.value();
340             if(!key.hasScan)
341                 continue;
342             float x = key.x + offX - key.width / 2.f + 1.f;
343             float y = key.y + offY - key.height / 2.f;
344             float w = key.width - 2.f;
345             float h = key.height;
346             // Print the key's friendly name (with some exceptions)
347             QString keyName = KbBind::globalRemap(key.name);
348             QString name = key.friendlyName(false);
349             name = name.split(" ").last();
350             struct {
351                 const char* keyName, *displayName;
352             } names[] = {
```

```
353                 {"light", ""}, {"lock", ""}, {"mute", ""}, {"volup", keyMap.
    model() == KeyMap::K65 ? "))" : ""}, {"voldn", keyMap.
    model() == KeyMap::K65 ? ")" : ""},
354                 {"prtscn",  "PrtScn\nSysRq"}, {"scroll", "Scroll\nLock"}, {"pause", "Pause\nBreak"}, {"stop
    ", ""}, {"prev", "|"}, {"play", "||"}, {"next", "|"},
355                 {"pgup", "Page\nUp"}, {"pgdn", "Page\nDown"}, {"numlock", "Num\nLock"},
356                 {"caps", "Caps"}, {"lshift", "Shift"}, {"rshift", "Shift"},
357 #ifdef Q_OS_MACX
358                 {"lctrl", ""}, {"rctrl", ""}, {"lwin", ""}, {"rwin", ""}, {"lalt", ""}, {"ralt", ""},
359 #else
360                 {"lctrl", "Ctrl"}, {"rctrl", "Ctrl"}, {"lwin", ""}, {"rwin", ""}, {"lalt", "Alt"}, {"ralt
    ", "Alt"},
361 #endif
362                 {"rmenu", ""}, {"up", ""}, {"left", ""}, {"down", ""}, {"right", ""}, {"fn","Fn"},
363                 {"mouse1", ""}, {"mouse2", ""}, {"mouse3", ""}, {"dpiup", ""}, {"dpidn", ""}, {"wheelup"
    , ""}, {"wheeldn", ""}, {"dpi", ""}, {"mouse5", ""}, {"mouse4", ""}, {"sniper", ""}
364             };
365             for(uint k = 0; k < sizeof(names) / sizeof(names[0]); k++){
366                 if(keyName == names[k].keyName){
367                     name = names[k].displayName;
368                     break;
369                 }
370             }
371             if(keyName == "thumb1" && model == KeyMap::SABRE)
372                 name = "";
373             if(keyName == "mr" || keyName == "m1" || keyName == "m2" || keyName == "m3" || keyName == "up"
    || keyName == "down" || keyName == "left" || keyName == "right")
374                 // Use a smaller size for MR, M1 - M3, and arrow keys
375                 font.setPixelSize(font.pixelSize() * 0.75);
376             else if(keyName == "end")
377                 // Use a smaller size for "End" to match everything else in that area
378                 font.setPixelSize(font.pixelSize() * 0.65);
379             else if(keyName == "light"
380 #ifndef Q_OS_MACX
381                     || keyName == "lwin" || keyName == "rwin"
382 #endif
383                     )
384                 // Use a larger font size for Super (Linux only) and Brightness to compensate for the
    unicode symbols looking smaller
385                 font.setPixelSize(font.pixelSize() * 1.3);
386             if((layout == KeyMap::EU || layout == KeyMap::EU_DVORAK) && (keyName
    == "hash" || keyName == "bslash_iso"))
387                 // Don't differentiate backslashes on the EU layout
388                 name = "\\";
389             // Determine the appropriate size to draw the text at
390             decPainter.setFont(font);
391             QRectF rect(x * scale, y * scale - 1, w * scale, h * scale);
392             int flags = Qt::AlignHCenter | Qt::AlignVCenter | Qt::TextWordWrap;
393             QRectF bounds = decPainter.boundingRect(rect, flags, name);
394             while((bounds.height() >= rect.height() - 8. || bounds.width() >= rect.width() - 2.) && font.
    pixelSize() >= 5){
395                 // Scale font size down until it fits inside the key
396                 font.setPixelSize(font.pixelSize() - 2);
397                 decPainter.setFont(font);
398                 bounds = decPainter.boundingRect(rect, flags, name);
399             }
400             // Pick color based on key function
401             QString bind = _bindMap.value(key.name);
402             QString def = KbBind::defaultAction(key.name, model);
403             if(bind.isEmpty())
404                 // Unbound - red
405                 decPainter.setPen(QColor(255, 136, 136));
406             else if(KeyAction(bind).isProgram())
407                 // Custom program - orange
408                 decPainter.setPen(QColor(255, 224, 192));
409             else if(KeyAction(bind).isSpecial() && (bind == def || !
    KeyAction(def).isSpecial()))
410                 // Special function - blue (only if not mapped to a different function - if a special
    function is remapped, color it yellow)
411                 decPainter.setPen(QColor(128, 224, 255));
412             else if(KeyAction(bind).isMedia() && (bind == def || !
    KeyAction(def).isMedia()))
413                 // Media key - green
414                 decPainter.setPen(QColor(160, 255, 168));
415             else if(bind == def)
416                 // Standard key - white
417                 decPainter.setPen(QColor(255, 255, 255));
418             else
419                 // Remapped key - yellow
420                 decPainter.setPen(QColor(255, 248, 128));
421             decPainter.drawText(rect, flags, name);
422             font = font0;
423         }
424     }
425     // Create drop shadow effects
426     QGraphicsDropShadowEffect* bgEffect = new QGraphicsDropShadowEffect;  // Have to use "new", creating
    these on the stack causes a crash...
```

```
427     bgEffect->setBlurRadius(2.);
428     bgEffect->setColor(QColor(0, 0, 0, 32));
429     bgEffect->setOffset(0, 1);
430     QGraphicsDropShadowEffect* decEffect = new QGraphicsDropShadowEffect;
431     decEffect->setBlurRadius(4.);
432     decEffect->setColor(QColor(0, 0, 0, 104));
433     decEffect->setOffset(0, 1);
434     // Apply them to the pixmaps
435     QGraphicsPixmapItem* bgItem = new QGraphicsPixmapItem(keyBG);
436     bgItem->setGraphicsEffect(bgEffect);
437     QGraphicsPixmapItem* decItem = new QGraphicsPixmapItem(decoration);
438     decItem->setGraphicsEffect(decEffect);
439     // Render everything
440     QGraphicsScene* scene = new QGraphicsScene;
441     scene->addItem(bgItem);
442     scene->addItem(decItem);
443     // It has to be rendered onto yet another pixmap or else DPI scaling will look terrible...
444     QPixmap final(wWidth * ratio, wHeight * ratio);
445     final.fill(QColor(0, 0, 0, 0));
446     QPainter finalPainter(&final);
447     scene->render(&finalPainter, QRectF(0, 0, wWidth * ratio, wHeight * ratio), QRectF(0, 0, wWidth * ratio
    , wHeight * ratio));
448     delete scene;   // <- Automatically cleans up the rest of the objects
449 #if QT_VERSION >= QT_VERSION_CHECK(5, 3, 0)
450     final.setDevicePixelRatio(ratio);
451 #endif
452     painter.drawPixmap(QPointF(0., 0.), final);
453 }
```

Here is the call graph for this function:



12.43.5.17 bool KeyWidget::rgbMode ( ) [inline]

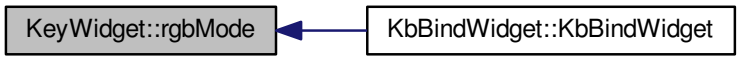
Definition at line 17 of file keywidget.h.

References _rgbMode.

Referenced by KbBindWidget::KbBindWidget().

```
17 { return _rgbMode; }
```

Here is the caller graph for this function:



12.43.5.18 void KeyWidget::rgbMode (bool *newRgbMode*) [inline]

Definition at line 18 of file keywidget.h.

References _rgbMode.

```
18 { _rgbMode = newRgbMode; update(); }
```

12.43.5.19 void KeyWidget::selectAll ( )

Definition at line 596 of file keywidget.cpp.

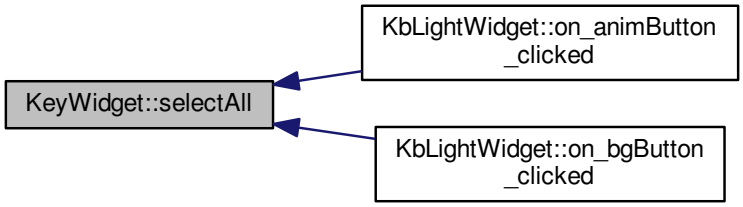
References _rgbMode, Key::hasLed, Key::hasScan, keyMap, mouseDownMode, Key::name, newSelection, NONE, KeyMap::positions(), selection, and selectionChanged().

Referenced by KbLightWidget::on_animButton_clicked(), and KbLightWidget::on_bgButton_clicked().

```
596                          {
597     selection.fill(false);
598     // Fill the selection with all keys that have an LED/scancode (depending on widget mode)
599     int i = 0;
600     QStringList selectedNames;
601     foreach(const Key& key, keyMap.positions()){
602         // Sidelights can't be selected
603         if(strcmp(key.name, "lsidel") && strcmp(key.name, "rsidel")
604            && ((_rgbMode && key.hasLed) || !(_rgbMode && key.
      hasScan))){
605             selection.setBit(i);
606             selectedNames << key.name;
607         }
608         i++;
609     }
610     // Clear mousedown state
611     newSelection.fill(false);
612     mouseDownMode = NONE;
613     update();
614     emit selectionChanged(selectedNames);
615 }
```

Here is the call graph for this function:

Here is the caller graph for this function:



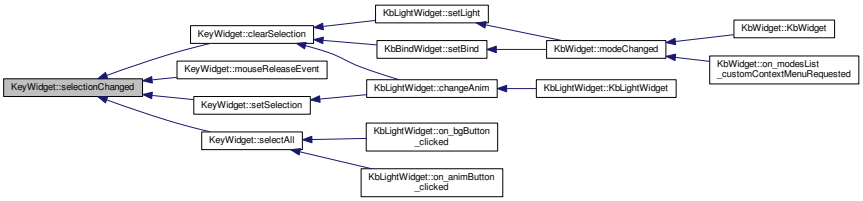
12.43.5.20 void KeyWidget::selectionChanged (QStringList *selected*) [signal]

Definition at line 164 of file moc_keywidget.cpp.

Referenced by clearSelection(), mouseReleaseEvent(), selectAll(), and setSelection().

```
165 {
166     void *_a[] = { Q_NULLPTR, const_cast<void*>(reinterpret_cast<const void*>(&_t1)) };
167     QMetaObject::activate(this, &staticMetaObject, 0, _a);
168 }
```

Here is the caller graph for this function:



12.43.5.21 void KeyWidget::setAnimation (const QStringList & *keys*)

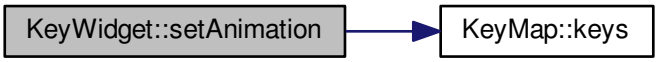
Definition at line 625 of file keywidget.cpp.

References animation, keyMap, and KeyMap::keys().

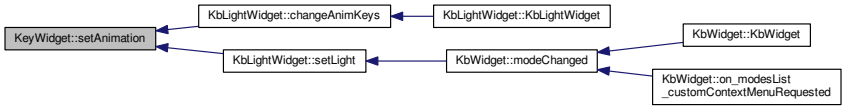
Referenced by KbLightWidget::changeAnimKeys(), and KbLightWidget::setLight().

```
625                                                 {
626     animation.fill(false);
627     QStringList allNames = keyMap.keys();
628     foreach(const QString& key, keys){
629         // Sidelights can't be selected
630         if(!strcmp(key.toLatin1(), "lsidel") || !strcmp(key.toLatin1(), "rsidel"))
631             continue;
632         int index = allNames.indexOf(key);
633         if(index >= 0)
634             animation.setBit(index);
635     }
636     update();
637 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.43.5.22 void KeyWidget::setAnimationToSelection ( )

Definition at line 639 of file keywidget.cpp.

References animation, and selection.

Referenced by KbLightWidget::changeAnim().

```
639                                         {
640     animation = selection;
641     update();
642 }
```

Here is the caller graph for this function:

12.43.5.23 void KeyWidget::setSelection (const QStringList & *keys*)

Definition at line 582 of file keywidget.cpp.

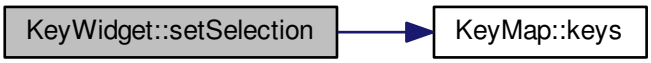
References keyMap, KeyMap::keys(), mouseDownMode, newSelection, NONE, selection, and selectionChanged().

Referenced by KbLightWidget::changeAnim().

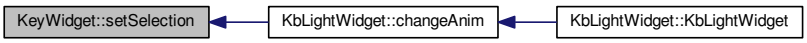
```
582                                                 {
583     selection.fill(false);
584     QStringList allNames = keyMap.keys();
585     foreach(const QString& key, keys){
586         int index = allNames.indexOf(key);
587         if(index >= 0)
```

```
588             selection.setBit(index);
589     }
590     newSelection.fill(false);
591     mouseDownMode = NONE;
592     update();
593     emit selectionChanged(keys);
594 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.43.5.24 void KeyWidget::sidelightToggled ( ) [signal]

Definition at line 171 of file moc_keywidget.cpp.

Referenced by mousePressEvent().

```
172 {
173     QMetaObject::activate(this, &staticMetaObject, 1, Q_NULLPTR);
174 }
```

Here is the caller graph for this function:

12.43.6 Field Documentation

12.43.6.1 BindMap KeyWidget::_bindMap [private]

Definition at line 58 of file keywidget.h.

Referenced by bindMap(), and paintEvent().

12.43.6.2 QColorMap KeyWidget::_colorMap [private]

Definition at line 56 of file keywidget.h.

Referenced by colorMap(), and paintEvent().

12.43.6.3 ColorMap KeyWidget::_displayColorMap [private]

Definition at line 57 of file keywidget.h.

Referenced by displayColorMap(), and paintEvent().

12.43.6.4 QSet<QString> KeyWidget::_indicators [private]

Definition at line 59 of file keywidget.h.

Referenced by displayColorMap(), mouseMoveEvent(), and paintEvent().

12.43.6.5 bool KeyWidget::_monochrome [private]

Definition at line 73 of file keywidget.h.

Referenced by monochrome(), and paintEvent().

12.43.6.6 bool KeyWidget::_rgbMode [private]

Definition at line 73 of file keywidget.h.

Referenced by mouseMoveEvent(), mousePressEvent(), paintEvent(), rgbMode(), and selectAll().

12.43.6.7 QBitArray KeyWidget::animation [private]

Definition at line 63 of file keywidget.h.

Referenced by clearAnimation(), map(), paintEvent(), setAnimation(), and setAnimationToSelection().

12.43.6.8 KeyMap KeyWidget::keyMap [private]

Definition at line 55 of file keywidget.h.

Referenced by drawInfo(), map(), mouseMoveEvent(), mousePressEvent(), mouseReleaseEvent(), paintEvent(), selectAll(), setAnimation(), and setSelection().

12.43.6.9 int KeyWidget::mouseCurrentX [private]

Definition at line 65 of file keywidget.h.

Referenced by mouseMoveEvent(), mousePressEvent(), and paintEvent().

12.43.6.10 int KeyWidget::mouseCurrentY [private]

Definition at line 65 of file keywidget.h.

Referenced by mouseMoveEvent(), mousePressEvent(), and paintEvent().

12.43.6.11 enum { ... } KeyWidget::mouseDownMode [private]

Referenced by clearSelection(), mouseMoveEvent(), mousePressEvent(), mouseReleaseEvent(), paintEvent(), selectAll(), and setSelection().

12.43.6.12 int KeyWidget::mouseDownX [private]

Definition at line 64 of file keywidget.h.

Referenced by mouseMoveEvent(), mousePressEvent(), and paintEvent().

12.43.6.13 int KeyWidget::mouseDownY [private]

Definition at line 64 of file keywidget.h.

Referenced by mouseMoveEvent(), mousePressEvent(), and paintEvent().

12.43.6.14 QBitArray KeyWidget::newSelection [private]

Definition at line 62 of file keywidget.h.

Referenced by clearSelection(), map(), mouseMoveEvent(), mousePressEvent(), mouseReleaseEvent(), paintEvent(), selectAll(), and setSelection().

12.43.6.15 QBitArray KeyWidget::selection [private]

Definition at line 61 of file keywidget.h.

Referenced by clearSelection(), map(), mouseReleaseEvent(), paintEvent(), selectAll(), setAnimationToSelection(), and setSelection().

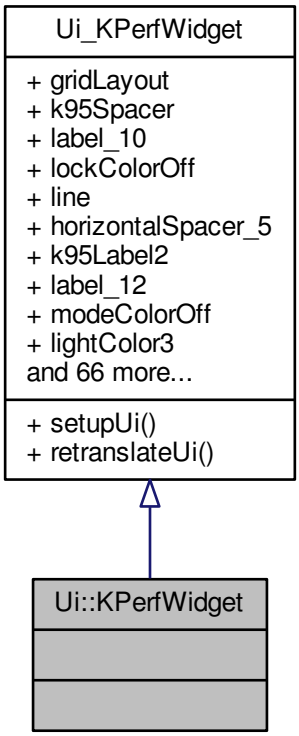
The documentation for this class was generated from the following files:

- src/ckb/keywidget.h
- src/ckb/keywidget.cpp
- src/ckb/release/moc_keywidget.cpp

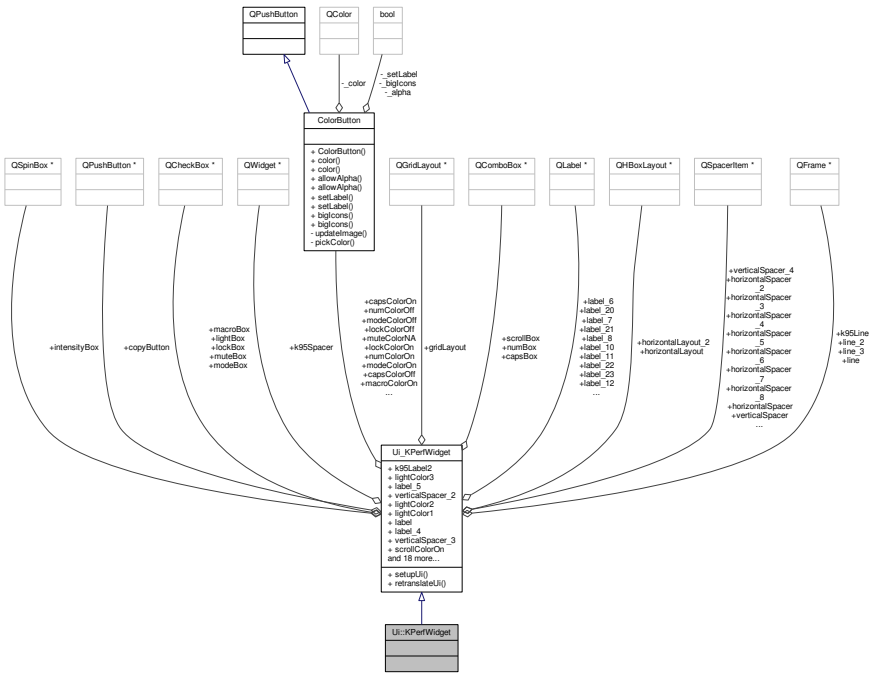
12.44 Ui::KPerfWidget Class Reference

`#include <src/ckb/ui_kperfwidget.h>`

Inheritance diagram for Ui::KPerfWidget:



Collaboration diagram for Ui::KPerfWidget:



Public Member Functions

- void setupUi (QWidget *KPerfWidget)
- void retranslateUi (QWidget *KPerfWidget)

Data Fields

- QGridLayout * gridLayout
- QWidget * k95Spacer
- QLabel * label_10
- ColorButton * lockColorOff
- QFrame * line
- QSpacerItem * horizontalSpacer_5
- QLabel * k95Label2
- QLabel * label_12
- ColorButton * modeColorOff
- ColorButton * lightColor3
- QSpacerItem * horizontalSpacer
- QCheckBox * lockBox
- ColorButton * modeColorOn
- QCheckBox * macroBox
- ColorButton * numColorOn
- QComboBox * numBox
- QLabel * label_5
- QSpacerItem * verticalSpacer_2
- ColorButton * lightColor2
- QSpacerItem * horizontalSpacer_7
- QLabel * label_8

- QCheckBox * lightBox
- ColorButton * lightColor1
- QSpacerItem * verticalSpacer_4
- QLabel * label_6
- QLabel * label
- QSpacerItem * horizontalSpacer_6
- ColorButton * lockColorOn
- QLabel * label_4
- QSpacerItem * verticalSpacer
- QSpacerItem * verticalSpacer_3
- QCheckBox * modeBox
- ColorButton * capsColorOff
- ColorButton * numColorOff
- ColorButton * capsColorOn
- QFrame * line_3
- QLabel * label_22
- QComboBox * scrollBox
- QLabel * label_7
- ColorButton * scrollColorOn
- QComboBox * capsBox
- ColorButton * scrollColorOff
- QCheckBox * muteBox
- ColorButton * muteColorOn
- ColorButton * macroColorOff
- QLabel * label_3
- ColorButton * muteColorOff
- QLabel * k95Label3
- ColorButton * macroColorOn
- ColorButton * muteColorNA
- QLabel * label_19
- QFrame * k95Line
- QLabel * label_21
- QLabel * label_24
- QLabel * label_20
- QLabel * label_16
- QLabel * label_14
- QLabel * label_18
- QSpacerItem * horizontalSpacer_2
- QLabel * label_15
- QLabel * k95Label4
- QLabel * label_17
- QLabel * label_11
- QLabel * label_23
- QLabel * k95Label1
- QHBoxLayout * horizontalLayout_2
- QPushButton * copyButton
- QSpacerItem * horizontalSpacer_8
- QHBoxLayout * horizontalLayout
- QLabel * label_2
- QSpacerItem * horizontalSpacer_4
- QSpinBox * intensityBox
- QSpacerItem * horizontalSpacer_3
- QLabel * k95Label5
- QFrame * line_2
- QLabel * k95Label6

12.44.1 Detailed Description

Definition at line 704 of file ui_kperfwidget.h.

12.44.2 Member Function Documentation

12.44.2.1 void Ui_KPerfWidget::retranslateUi (QWidget * *KPerfWidget*) [inline],[inherited]

Definition at line 616 of file ui_kperfwidget.h.

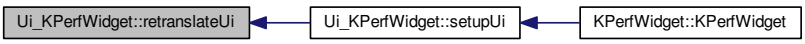
References Ui_KPerfWidget::capsBox, Ui_KPerfWidget::capsColorOff, Ui_KPerfWidget::capsColorOn, Ui_KPerfWidget::copyButton, Ui_KPerfWidget::intensityBox, Ui_KPerfWidget::k95Label1, Ui_KPerfWidget::k95Label2, Ui_KPerfWidget::k95Label3, Ui_KPerfWidget::k95Label4, Ui_KPerfWidget::k95Label5, Ui_KPerfWidget::k95Label6, Ui_KPerfWidget::label, Ui_KPerfWidget::label_10, Ui_KPerfWidget::label_11, Ui_KPerfWidget::label_12, Ui_KPerfWidget::label_14, Ui_KPerfWidget::label_15, Ui_KPerfWidget::label_16, Ui_KPerfWidget::label_17, Ui_KPerfWidget::label_18, Ui_KPerfWidget::label_19, Ui_KPerfWidget::label_2, Ui_KPerfWidget::label_20, Ui_KPerfWidget::label_21, Ui_KPerfWidget::label_22, Ui_KPerfWidget::label_23, Ui_KPerfWidget::label_24, Ui_KPerfWidget::label_3, Ui_KPerfWidget::label_4, Ui_KPerfWidget::label_5, Ui_KPerfWidget::label_6, Ui_KPerfWidget::label_7, Ui_KPerfWidget::label_8, Ui_KPerfWidget::lightBox, Ui_KPerfWidget::lightColor1, Ui_KPerfWidget::lightColor2, Ui_KPerfWidget::lightColor3, Ui_KPerfWidget::lockBox, Ui_KPerfWidget::lockColorOff, Ui_KPerfWidget::lockColorOn, Ui_KPerfWidget::macroBox, Ui_KPerfWidget::macroColorOff, Ui_KPerfWidget::macroColorOn, Ui_KPerfWidget::modeBox, Ui_KPerfWidget::modeColorOff, Ui_KPerfWidget::modeColorOn, Ui_KPerfWidget::muteBox, Ui_KPerfWidget::muteColorNA, Ui_KPerfWidget::muteColorOff, Ui_KPerfWidget::muteColorOn, Ui_KPerfWidget::numBox, Ui_KPerfWidget::numColorOff, Ui_KPerfWidget::numColorOn, Ui_KPerfWidget::scrollBox, Ui_KPerfWidget::scrollColorOff, and Ui_KPerfWidget::scrollColorOn.

Referenced by Ui_KPerfWidget::setupUi().

```
617     {
618         KPerfWidget->setWindowTitle(QApplication::translate("KPerfWidget", "Form", Q_NULLPTR));
619         label_10->setText(QApplication::translate("KPerfWidget", "100%", Q_NULLPTR));
620         lockColorOff->setText(QString());
621         k95Label2->setText(QApplication::translate("KPerfWidget", "On", Q_NULLPTR));
622         label_12->setText(QApplication::translate("KPerfWidget", "On", Q_NULLPTR));
623         modeColorOff->setText(QString());
624         lightColor3->setText(QString());
625         lockBox->setText(QApplication::translate("KPerfWidget", "Indicate Windows Lock:", Q_NULLPTR)
    );
626         modeColorOn->setText(QString());
627         macroBox->setText(QApplication::translate("KPerfWidget", "Indicate macro recording:",
    Q_NULLPTR));
628         numColorOn->setText(QString());
629         numBox->clear();
630         numBox->insertItems(0, QStringList()
631          << QApplication::translate("KPerfWidget", "Normal", Q_NULLPTR)
632          << QApplication::translate("KPerfWidget", "Always on", Q_NULLPTR)
633          << QApplication::translate("KPerfWidget", "Always off", Q_NULLPTR)
634          << QApplication::translate("KPerfWidget", "RGB indicator", Q_NULLPTR)
635          << QApplication::translate("KPerfWidget", "RGB + normal", Q_NULLPTR)
636         );
637         label_5->setText(QApplication::translate("KPerfWidget", "Num lock:", Q_NULLPTR));
638         lightColor2->setText(QString());
639         label_8->setText(QApplication::translate("KPerfWidget", "Note: many desktop environments do
     not use scroll lock. The indicator may not turn on even when enabled.", Q_NULLPTR));
640         lightBox->setText(QApplication::translate("KPerfWidget", "Indicate brightness:", Q_NULLPTR)
    );
641         lightColor1->setText(QString());
642         label_6->setText(QApplication::translate("KPerfWidget", "Caps lock:", Q_NULLPTR));
643         label->setText(QApplication::translate("KPerfWidget", "Miscellaneous", Q_NULLPTR));
644         lockColorOn->setText(QString());
645         label_4->setText(QApplication::translate("KPerfWidget", "Lock lights", Q_NULLPTR));
646         modeBox->setText(QApplication::translate("KPerfWidget", "Indicate current mode:", Q_NULLPTR)
    );
647         capsColorOff->setText(QString());
648         numColorOff->setText(QString());
649         capsColorOn->setText(QString());
650         label_22->setText(QApplication::translate("KPerfWidget", "Off", Q_NULLPTR));
651         scrollBox->clear();
652         scrollBox->insertItems(0, QStringList()
653          << QApplication::translate("KPerfWidget", "Normal", Q_NULLPTR)
654          << QApplication::translate("KPerfWidget", "Always on", Q_NULLPTR)
655          << QApplication::translate("KPerfWidget", "Always off", Q_NULLPTR)
656          << QApplication::translate("KPerfWidget", "RGB indicator", Q_NULLPTR)
```

```
657          << QApplication::translate("KPerfWidget", "RGB + normal", Q_NULLPTR)
658         );
659         label_7->setText(QApplication::translate("KPerfWidget", "Scroll lock:", Q_NULLPTR));
660         scrollColorOn->setText(QString());
661         capsBox->clear();
662         capsBox->insertItems(0, QStringList()
663          << QApplication::translate("KPerfWidget", "Normal", Q_NULLPTR)
664          << QApplication::translate("KPerfWidget", "Always on", Q_NULLPTR)
665          << QApplication::translate("KPerfWidget", "Always off", Q_NULLPTR)
666          << QApplication::translate("KPerfWidget", "RGB indicator", Q_NULLPTR)
667          << QApplication::translate("KPerfWidget", "RGB + normal", Q_NULLPTR)
668         );
669         scrollColorOff->setText(QString());
670         muteBox->setText(QApplication::translate("KPerfWidget", "Indicate mute:", Q_NULLPTR));
671         muteColorOn->setText(QString());
672         macroColorOff->setText(QString());
673         label_3->setText(QApplication::translate("KPerfWidget", "Top row", Q_NULLPTR));
674         muteColorOff->setText(QString());
675         k95Label3->setText(QApplication::translate("KPerfWidget", "Off", Q_NULLPTR));
676         macroColorOn->setText(QString());
677 #ifndef QT_NO_TOOLTIP
678         muteColorNA->setToolTip(QApplication::translate("KPerfWidget", "If the sound device does
       not support muting, or if mute status is unknown, this color will be displayed.", Q_NULLPTR));
679 #endif // QT_NO_TOOLTIP
680         muteColorNA->setText(QString());
681         label_19->setText(QApplication::translate("KPerfWidget", "On", Q_NULLPTR));
682         label_21->setText(QApplication::translate("KPerfWidget", "On", Q_NULLPTR));
683         label_24->setText(QApplication::translate("KPerfWidget", "Off", Q_NULLPTR));
684         label_20->setText(QApplication::translate("KPerfWidget", "On", Q_NULLPTR));
685         label_16->setText(QApplication::translate("KPerfWidget", "67%", Q_NULLPTR));
686         label_14->setText(QApplication::translate("KPerfWidget", "Off", Q_NULLPTR));
687         label_18->setText(QApplication::translate("KPerfWidget", "33%", Q_NULLPTR));
688         label_15->setText(QApplication::translate("KPerfWidget", "Off", Q_NULLPTR));
689         k95Label4->setText(QApplication::translate("KPerfWidget", "On", Q_NULLPTR));
690         label_17->setText(QApplication::translate("KPerfWidget", "Unknown", Q_NULLPTR));
691         label_11->setText(QApplication::translate("KPerfWidget", "On", Q_NULLPTR));
692         label_23->setText(QApplication::translate("KPerfWidget", "Off", Q_NULLPTR));
693         k95Label1->setText(QApplication::translate("KPerfWidget", "K95 buttons", Q_NULLPTR));
694         copyButton->setText(QApplication::translate("KPerfWidget", "Copy to mode...", Q_NULLPTR))
    ;
695         label_2->setText(QApplication::translate("KPerfWidget", "Indicator intensity:", Q_NULLPTR));
696         intensityBox->setSuffix(QApplication::translate("KPerfWidget", "%", Q_NULLPTR));
697         k95Label5->setText(QApplication::translate("KPerfWidget", "Off", Q_NULLPTR));
698         k95Label6->setText(QApplication::translate("KPerfWidget", "Note: macro recording not yet
      implemented. Indicator will always display \"off\".", Q_NULLPTR));
699     } // retranslateUi
```

Here is the caller graph for this function:



12.44.2.2 void Ui_KPerfWidget::setupUi (QWidget * *KPerfWidget*) [inline], [inherited]

Definition at line 111 of file ui_kperfwidget.h.

References Ui_KPerfWidget::capsBox, Ui_KPerfWidget::capsColorOff, Ui_KPerfWidget::capsColorOn, Ui_KPerfWidget::copyButton, Ui_KPerfWidget::gridLayout, Ui_KPerfWidget::horizontalLayout, Ui_KPerfWidget::horizontalLayout_2, Ui_KPerfWidget::horizontalSpacer, Ui_KPerfWidget::horizontalSpacer_2, Ui_KPerfWidget::horizontalSpacer_3, Ui_KPerfWidget::horizontalSpacer_4, Ui_KPerfWidget::horizontalSpacer_5, Ui_KPerfWidget::horizontalSpacer_6, Ui_KPerfWidget::horizontalSpacer_7, Ui_KPerfWidget::horizontalSpacer_8, Ui_KPerfWidget::intensityBox, Ui_KPerfWidget::k95Label1, Ui_KPerfWidget::k95Label2, Ui_KPerfWidget::k95Label3, Ui_KPerfWidget::k95Label4, Ui_KPerfWidget::k95Label5, Ui_KPerfWidget::k95Label6, Ui_KPerfWidget::k95Line, Ui_KPerfWidget::k95Spacer, Ui_KPerfWidget::label, Ui_KPerfWidget::label_10, Ui_KPerfWidget::label_11, Ui_KPerfWidget::label_12, Ui_KPerfWidget::label_14, Ui_KPerfWidget::label_15, Ui_KPerfWidget::label_16, Ui_KPerfWidget::label_17, Ui_KPerfWidget::label_18, Ui_KPerfWidget::label_19, Ui_KPerfWidget::label_2, Ui_KPerfWidget::label_20, Ui_KPerfWidget::label_21, Ui_KPerfWidget::label_22, Ui_KPerfWidget::label_23, Ui_KPerfWidget::label_24, Ui_KPerfWidget::label_3, Ui_KPerfWidget::label_4, Ui_KPerfWidget::label_5, Ui_KPerfWidget::label_6, Ui_KPerfWidget-

::label_7, Ui_KPerfWidget::label_8, Ui_KPerfWidget::lightBox, Ui_KPerfWidget::lightColor1, Ui_KPerfWidget::lightColor2, Ui_KPerfWidget::lightColor3, Ui_KPerfWidget::line, Ui_KPerfWidget::line_2, Ui_KPerfWidget::line_3, Ui_KPerfWidget::lockBox, Ui_KPerfWidget::lockColorOff, Ui_KPerfWidget::lockColorOn, Ui_KPerfWidget::macroBox, Ui_KPerfWidget::macroColorOff, Ui_KPerfWidget::macroColorOn, Ui_KPerfWidget::modeBox, Ui_KPerfWidget::modeColorOff, Ui_KPerfWidget::modeColorOn, Ui_KPerfWidget::muteBox, Ui_KPerfWidget::muteColorNA, Ui_KPerfWidget::muteColorOff, Ui_KPerfWidget::muteColorOn, Ui_KPerfWidget::numBox, Ui_KPerfWidget::numColorOff, Ui_KPerfWidget::numColorOn, Ui_KPerfWidget::retranslateUi(), Ui_KPerfWidget::scrollBox, Ui_KPerfWidget::scrollColorOff, Ui_KPerfWidget::scrollColorOn, Ui_KPerfWidget::verticalSpacer, Ui_KPerfWidget::verticalSpacer_2, Ui_KPerfWidget::verticalSpacer_3, and Ui_KPerfWidget::verticalSpacer_4.

Referenced by KPerfWidget::KPerfWidget().

```
112     {
113         if (KPerfWidget->objectName().isEmpty())
114             KPerfWidget->setObjectName(QStringLiteral("KPerfWidget"));
115         KPerfWidget->resize(840, 679);
116         gridLayout = new QGridLayout(KPerfWidget);
117         gridLayout->setObjectName(QStringLiteral("gridLayout"));
118         k95Spacer = new QWidget(KPerfWidget);
119         k95Spacer->setObjectName(QStringLiteral("k95Spacer"));
120         QSizePolicy sizePolicy(QSizePolicy::Fixed, QSizePolicy::Fixed);
121         sizePolicy.setHorizontalStretch(0);
122         sizePolicy.setVerticalStretch(0);
123         sizePolicy.setHeightForWidth(k95Spacer->sizePolicy().hasHeightForWidth());
124         k95Spacer->setSizePolicy(sizePolicy);
125         k95Spacer->setMinimumSize(QSize(15, 10));
126         k95Spacer->setMaximumSize(QSize(15, 10));
127 
128         gridLayout->addWidget(k95Spacer, 12, 4, 1, 1);
129 
130         label_10 = new QLabel(KPerfWidget);
131         label_10->setObjectName(QStringLiteral("label_10"));
132 
133         gridLayout->addWidget(label_10, 15, 5, 1, 1);
134 
135         lockColorOff = new ColorButton(KPerfWidget);
136         lockColorOff->setObjectName(QStringLiteral("lockColorOff"));
137         sizePolicy.setHeightForWidth(lockColorOff->sizePolicy().hasHeightForWidth());
138         lockColorOff->setSizePolicy(sizePolicy);
139         lockColorOff->setMinimumSize(QSize(0, 40));
140         lockColorOff->setMaximumSize(QSize(16777215, 40));
141 
142         gridLayout->addWidget(lockColorOff, 16, 7, 1, 1);
143 
144         line = new QFrame(KPerfWidget);
145         line->setObjectName(QStringLiteral("line"));
146         line->setFrameShape(QFrame::HLine);
147         line->setFrameShadow(QFrame::Sunken);
148 
149         gridLayout->addWidget(line, 2, 0, 1, 13);
150 
151         horizontalSpacer_5 = new QSpacerItem(10, 20, QSizePolicy::Fixed,
    QSizePolicy::Minimum);
152 
153         gridLayout->addItem(horizontalSpacer_5, 17, 6, 1, 1);
154 
155         k95Label2 = new QLabel(KPerfWidget);
156         k95Label2->setObjectName(QStringLiteral("k95Label2"));
157 
158         gridLayout->addWidget(k95Label2, 9, 5, 1, 1);
159 
160         label_12 = new QLabel(KPerfWidget);
161         label_12->setObjectName(QStringLiteral("label_12"));
162 
163         gridLayout->addWidget(label_12, 17, 5, 1, 1);
164 
165         modeColorOff = new ColorButton(KPerfWidget);
166         modeColorOff->setObjectName(QStringLiteral("modeColorOff"));
167         sizePolicy.setHeightForWidth(modeColorOff->sizePolicy().hasHeightForWidth());
168         modeColorOff->setSizePolicy(sizePolicy);
169         modeColorOff->setMinimumSize(QSize(0, 40));
170         modeColorOff->setMaximumSize(QSize(16777215, 40));
171 
172         gridLayout->addWidget(modeColorOff, 9, 7, 1, 1);
173 
174         lightColor3 = new ColorButton(KPerfWidget);
175         lightColor3->setObjectName(QStringLiteral("lightColor3"));
176         sizePolicy.setHeightForWidth(lightColor3->sizePolicy().hasHeightForWidth());
177         lightColor3->setSizePolicy(sizePolicy);
178         lightColor3->setMinimumSize(QSize(0, 40));
179         lightColor3->setMaximumSize(QSize(16777215, 40));
180 
181         gridLayout->addWidget(lightColor3, 15, 4, 1, 1);
```

```
182
183         horizontalSpacer = new QSpacerItem(40, 20, QSizePolicy::Expanding,
    QSizePolicy::Minimum);
184
185         gridLayout->addItem(horizontalSpacer, 15, 12, 1, 1);
186
187         lockBox = new QCheckBox(KPerfWidget);
188         lockBox->setObjectName(QStringLiteral("lockBox"));
189
190         gridLayout->addWidget(lockBox, 16, 1, 1, 2);
191
192         modeColorOn = new ColorButton(KPerfWidget);
193         modeColorOn->setObjectName(QStringLiteral("modeColorOn"));
194         sizePolicy.setHeightForWidth(modeColorOn->sizePolicy().hasHeightForWidth());
195         modeColorOn->setSizePolicy(sizePolicy);
196         modeColorOn->setMinimumSize(QSize(0, 40));
197         modeColorOn->setMaximumSize(QSize(16777215, 40));
198
199         gridLayout->addWidget(modeColorOn, 9, 4, 1, 1);
200
201         macroBox = new QCheckBox(KPerfWidget);
202         macroBox->setObjectName(QStringLiteral("macroBox"));
203
204         gridLayout->addWidget(macroBox, 10, 1, 1, 2);
205
206         numColorOn = new ColorButton(KPerfWidget);
207         numColorOn->setObjectName(QStringLiteral("numColorOn"));
208         sizePolicy.setHeightForWidth(numColorOn->sizePolicy().hasHeightForWidth());
209         numColorOn->setSizePolicy(sizePolicy);
210         numColorOn->setMinimumSize(QSize(0, 40));
211         numColorOn->setMaximumSize(QSize(16777215, 40));
212
213         gridLayout->addWidget(numColorOn, 21, 4, 1, 1);
214
215         numBox = new QComboBox(KPerfWidget);
216         numBox->setObjectName(QStringLiteral("numBox"));
217
218         gridLayout->addWidget(numBox, 21, 2, 1, 1);
219
220         label_5 = new QLabel(KPerfWidget);
221         label_5->setObjectName(QStringLiteral("label_5"));
222
223         gridLayout->addWidget(label_5, 21, 1, 1, 1);
224
225         verticalSpacer_2 = new QSpacerItem(20, 20, QSizePolicy::Minimum, QSizePolicy::Fixed
    );
226
227         gridLayout->addItem(verticalSpacer_2, 4, 1, 1, 1);
228
229         lightColor2 = new ColorButton(KPerfWidget);
230         lightColor2->setObjectName(QStringLiteral("lightColor2"));
231         sizePolicy.setHeightForWidth(lightColor2->sizePolicy().hasHeightForWidth());
232         lightColor2->setSizePolicy(sizePolicy);
233         lightColor2->setMinimumSize(QSize(0, 40));
234         lightColor2->setMaximumSize(QSize(16777215, 40));
235
236         gridLayout->addWidget(lightColor2, 15, 7, 1, 1);
237
238         horizontalSpacer_7 = new QSpacerItem(10, 20, QSizePolicy::Fixed,
    QSizePolicy::Minimum);
239
240         gridLayout->addItem(horizontalSpacer_7, 17, 3, 1, 1);
241
242         label_8 = new QLabel(KPerfWidget);
243         label_8->setObjectName(QStringLiteral("label_8"));
244         QFont font;
245         font.setPointSize(8);
246         font.setItalic(true);
247         label_8->setFont(font);
248
249         gridLayout->addWidget(label_8, 25, 1, 1, 12);
250
251         lightBox = new QCheckBox(KPerfWidget);
252         lightBox->setObjectName(QStringLiteral("lightBox"));
253
254         gridLayout->addWidget(lightBox, 15, 1, 1, 2);
255
256         lightColor1 = new ColorButton(KPerfWidget);
257         lightColor1->setObjectName(QStringLiteral("lightColor1"));
258         sizePolicy.setHeightForWidth(lightColor1->sizePolicy().hasHeightForWidth());
259         lightColor1->setSizePolicy(sizePolicy);
260         lightColor1->setMinimumSize(QSize(0, 40));
261         lightColor1->setMaximumSize(QSize(16777215, 40));
262
263         gridLayout->addWidget(lightColor1, 15, 10, 1, 1);
264
265         verticalSpacer_4 = new QSpacerItem(0, 5, QSizePolicy::Minimum, QSizePolicy::Fixed);
```

```
266
267         gridLayout->addItem(verticalSpacer_4, 24, 4, 1, 1);
268
269         label_6 = new QLabel(KPerfWidget);
270         label_6->setObjectName(QStringLiteral("label_6"));
271
272         gridLayout->addWidget(label_6, 22, 1, 1, 1);
273
274         label = new QLabel(KPerfWidget);
275         label->setObjectName(QStringLiteral("label"));
276         QFont font1;
277         font1.setBold(true);
278         font1.setWeight(75);
279         label->setFont(font1);
280
281         gridLayout->addWidget(label, 1, 0, 1, 13);
282
283         horizontalSpacer_6 = new QSpacerItem(10, 20, QSizePolicy::Fixed,
    QSizePolicy::Minimum);
284
285         gridLayout->addItem(horizontalSpacer_6, 17, 9, 1, 1);
286
287         lockColorOn = new ColorButton(KPerfWidget);
288         lockColorOn->setObjectName(QStringLiteral("lockColorOn"));
289         sizePolicy.setHeightForWidth(lockColorOn->sizePolicy().hasHeightForWidth());
290         lockColorOn->setSizePolicy(sizePolicy);
291         lockColorOn->setMinimumSize(QSize(0, 40));
292         lockColorOn->setMaximumSize(QSize(16777215, 40));
293
294         gridLayout->addWidget(lockColorOn, 16, 4, 1, 1);
295
296         label_4 = new QLabel(KPerfWidget);
297         label_4->setObjectName(QStringLiteral("label_4"));
298         label_4->setFont(font1);
299
300         gridLayout->addWidget(label_4, 19, 0, 1, 13);
301
302         verticalSpacer = new QSpacerItem(20, 40, QSizePolicy::Minimum, QSizePolicy::Expanding
    );
303
304         gridLayout->addItem(verticalSpacer, 26, 2, 1, 1);
305
306         verticalSpacer_3 = new QSpacerItem(20, 20, QSizePolicy::Minimum, QSizePolicy::Fixed
    );
307
308         gridLayout->addItem(verticalSpacer_3, 18, 4, 1, 1);
309
310         modeBox = new QCheckBox(KPerfWidget);
311         modeBox->setObjectName(QStringLiteral("modeBox"));
312
313         gridLayout->addWidget(modeBox, 9, 1, 1, 2);
314
315         capsColorOff = new ColorButton(KPerfWidget);
316         capsColorOff->setObjectName(QStringLiteral("capsColorOff"));
317         sizePolicy.setHeightForWidth(capsColorOff->sizePolicy().hasHeightForWidth());
318         capsColorOff->setSizePolicy(sizePolicy);
319         capsColorOff->setMinimumSize(QSize(0, 40));
320         capsColorOff->setMaximumSize(QSize(16777215, 40));
321
322         gridLayout->addWidget(capsColorOff, 22, 7, 1, 1);
323
324         numColorOff = new ColorButton(KPerfWidget);
325         numColorOff->setObjectName(QStringLiteral("numColorOff"));
326         QSizePolicy sizePolicy1(QSizePolicy::Fixed, QSizePolicy::Fixed);
327         sizePolicy1.setHorizontalStretch(0);
328         sizePolicy1.setVerticalStretch(40);
329         sizePolicy1.setHeightForWidth(numColorOff->sizePolicy().hasHeightForWidth());
330         numColorOff->setSizePolicy(sizePolicy1);
331         numColorOff->setMaximumSize(QSize(16777215, 40));
332
333         gridLayout->addWidget(numColorOff, 21, 7, 1, 1);
334
335         capsColorOn = new ColorButton(KPerfWidget);
336         capsColorOn->setObjectName(QStringLiteral("capsColorOn"));
337         sizePolicy.setHeightForWidth(capsColorOn->sizePolicy().hasHeightForWidth());
338         capsColorOn->setSizePolicy(sizePolicy);
339         capsColorOn->setMinimumSize(QSize(0, 40));
340         capsColorOn->setMaximumSize(QSize(16777215, 40));
341
342         gridLayout->addWidget(capsColorOn, 22, 4, 1, 1);
343
344         line_3 = new QFrame(KPerfWidget);
345         line_3->setObjectName(QStringLiteral("line_3"));
346         line_3->setFrameShape(QFrame::HLine);
347         line_3->setFrameShadow(QFrame::Sunken);
348
349         gridLayout->addWidget(line_3, 20, 0, 1, 13);
```

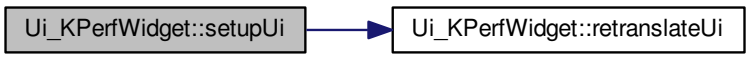
```
350
351         label_22 = new QLabel(KPerfWidget);
352         label_22->setObjectName(QStringLiteral("label_22"));
353
354         gridLayout->addWidget(label_22, 21, 8, 1, 1);
355
356         scrollBox = new QComboBox(KPerfWidget);
357         scrollBox->setObjectName(QStringLiteral("scrollBox"));
358
359         gridLayout->addWidget(scrollBox, 23, 2, 1, 1);
360
361         label_7 = new QLabel(KPerfWidget);
362         label_7->setObjectName(QStringLiteral("label_7"));
363
364         gridLayout->addWidget(label_7, 23, 1, 1, 1);
365
366         scrollColorOn = new ColorButton(KPerfWidget);
367         scrollColorOn->setObjectName(QStringLiteral("scrollColorOn"));
368         sizePolicy.setHeightForWidth(scrollColorOn->sizePolicy().hasHeightForWidth());
369         scrollColorOn->setSizePolicy(sizePolicy);
370         scrollColorOn->setMinimumSize(QSize(0, 40));
371         scrollColorOn->setMaximumSize(QSize(16777215, 40));
372
373         gridLayout->addWidget(scrollColorOn, 23, 4, 1, 1);
374
375         capsBox = new QComboBox(KPerfWidget);
376         capsBox->setObjectName(QStringLiteral("capsBox"));
377
378         gridLayout->addWidget(capsBox, 22, 2, 1, 1);
379
380         scrollColorOff = new ColorButton(KPerfWidget);
381         scrollColorOff->setObjectName(QStringLiteral("scrollColorOff"));
382         sizePolicy.setHeightForWidth(scrollColorOff->sizePolicy().hasHeightForWidth());
383         scrollColorOff->setSizePolicy(sizePolicy);
384         scrollColorOff->setMinimumSize(QSize(0, 40));
385         scrollColorOff->setMaximumSize(QSize(16777215, 40));
386
387         gridLayout->addWidget(scrollColorOff, 23, 7, 1, 1);
388
389         muteBox = new QCheckBox(KPerfWidget);
390         muteBox->setObjectName(QStringLiteral("muteBox"));
391
392         gridLayout->addWidget(muteBox, 17, 1, 1, 2);
393
394         muteColorOn = new ColorButton(KPerfWidget);
395         muteColorOn->setObjectName(QStringLiteral("muteColorOn"));
396         sizePolicy.setHeightForWidth(muteColorOn->sizePolicy().hasHeightForWidth());
397         muteColorOn->setSizePolicy(sizePolicy);
398         muteColorOn->setMinimumSize(QSize(0, 40));
399         muteColorOn->setMaximumSize(QSize(16777215, 40));
400
401         gridLayout->addWidget(muteColorOn, 17, 4, 1, 1);
402
403         macroColorOff = new ColorButton(KPerfWidget);
404         macroColorOff->setObjectName(QStringLiteral("macroColorOff"));
405         QSizePolicy sizePolicy2(QSizePolicy::Minimum, QSizePolicy::Fixed);
406         sizePolicy2.setHorizontalStretch(0);
407         sizePolicy2.setVerticalStretch(40);
408         sizePolicy2.setHeightForWidth(macroColorOff->sizePolicy().hasHeightForWidth());
409         macroColorOff->setSizePolicy(sizePolicy2);
410         macroColorOff->setMinimumSize(QSize(0, 40));
411
412         gridLayout->addWidget(macroColorOff, 10, 7, 1, 1);
413
414         label_3 = new QLabel(KPerfWidget);
415         label_3->setObjectName(QStringLiteral("label_3"));
416         label_3->setFont(font1);
417
418         gridLayout->addWidget(label_3, 13, 0, 1, 3);
419
420         muteColorOff = new ColorButton(KPerfWidget);
421         muteColorOff->setObjectName(QStringLiteral("muteColorOff"));
422         sizePolicy.setHeightForWidth(muteColorOff->sizePolicy().hasHeightForWidth());
423         muteColorOff->setSizePolicy(sizePolicy);
424         muteColorOff->setMinimumSize(QSize(0, 40));
425         muteColorOff->setMaximumSize(QSize(16777215, 40));
426
427         gridLayout->addWidget(muteColorOff, 17, 7, 1, 1);
428
429         k95Label3 = new QLabel(KPerfWidget);
430         k95Label3->setObjectName(QStringLiteral("k95Label3"));
431
432         gridLayout->addWidget(k95Label3, 9, 8, 1, 1);
433
434         macroColorOn = new ColorButton(KPerfWidget);
435         macroColorOn->setObjectName(QStringLiteral("macroColorOn"));
436         sizePolicy2.setHeightForWidth(macroColorOn->sizePolicy().hasHeightForWidth());
```

```
437         macroColorOn->setSizePolicy(sizePolicy2);
438         macroColorOn->setMinimumSize(QSize(0, 40));
439
440         gridLayout->addWidget(macroColorOn, 10, 4, 1, 1);
441
442         muteColorNA = new ColorButton(KPerfWidget);
443         muteColorNA->setObjectName(QStringLiteral("muteColorNA"));
444         muteColorNA->setMinimumSize(QSize(0, 40));
445         muteColorNA->setMaximumSize(QSize(16777215, 40));
446
447         gridLayout->addWidget(muteColorNA, 17, 10, 1, 1);
448
449         label_19 = new QLabel(KPerfWidget);
450         label_19->setObjectName(QStringLiteral("label_19"));
451
452         gridLayout->addWidget(label_19, 21, 5, 1, 1);
453
454         k95Line = new QFrame(KPerfWidget);
455         k95Line->setObjectName(QStringLiteral("k95Line"));
456         k95Line->setFrameShape(QFrame::HLine);
457         k95Line->setFrameShadow(QFrame::Sunken);
458
459         gridLayout->addWidget(k95Line, 8, 0, 1, 13);
460
461         label_21 = new QLabel(KPerfWidget);
462         label_21->setObjectName(QStringLiteral("label_21"));
463
464         gridLayout->addWidget(label_21, 23, 5, 1, 1);
465
466         label_24 = new QLabel(KPerfWidget);
467         label_24->setObjectName(QStringLiteral("label_24"));
468
469         gridLayout->addWidget(label_24, 23, 8, 1, 1);
470
471         label_20 = new QLabel(KPerfWidget);
472         label_20->setObjectName(QStringLiteral("label_20"));
473
474         gridLayout->addWidget(label_20, 22, 5, 1, 1);
475
476         label_16 = new QLabel(KPerfWidget);
477         label_16->setObjectName(QStringLiteral("label_16"));
478
479         gridLayout->addWidget(label_16, 15, 8, 1, 1);
480
481         label_14 = new QLabel(KPerfWidget);
482         label_14->setObjectName(QStringLiteral("label_14"));
483
484         gridLayout->addWidget(label_14, 17, 8, 1, 1);
485
486         label_18 = new QLabel(KPerfWidget);
487         label_18->setObjectName(QStringLiteral("label_18"));
488
489         gridLayout->addWidget(label_18, 15, 11, 1, 1);
490
491         horizontalSpacer_2 = new QSpacerItem(5, 20, QSizePolicy::Fixed,
    QSizePolicy::Minimum);
492
493         gridLayout->addItem(horizontalSpacer_2, 15, 0, 1, 1);
494
495         label_15 = new QLabel(KPerfWidget);
496         label_15->setObjectName(QStringLiteral("label_15"));
497
498         gridLayout->addWidget(label_15, 16, 8, 1, 1);
499
500         k95Label4 = new QLabel(KPerfWidget);
501         k95Label4->setObjectName(QStringLiteral("k95Label4"));
502
503         gridLayout->addWidget(k95Label4, 10, 5, 1, 1);
504
505         label_17 = new QLabel(KPerfWidget);
506         label_17->setObjectName(QStringLiteral("label_17"));
507
508         gridLayout->addWidget(label_17, 17, 11, 1, 1);
509
510         label_11 = new QLabel(KPerfWidget);
511         label_11->setObjectName(QStringLiteral("label_11"));
512
513         gridLayout->addWidget(label_11, 16, 5, 1, 1);
514
515         label_23 = new QLabel(KPerfWidget);
516         label_23->setObjectName(QStringLiteral("label_23"));
517
518         gridLayout->addWidget(label_23, 22, 8, 1, 1);
519
520         k95Label1 = new QLabel(KPerfWidget);
521         k95Label1->setObjectName(QStringLiteral("k95Label1"));
522         k95Label1->setFont(font1);
```

```
523
524         gridLayout->addWidget(k95Label1, 7, 0, 1, 5);
525
526         horizontalLayout_2 = new QHBoxLayout();
527         horizontalLayout_2->setObjectName(QStringLiteral("horizontalLayout_2"));
528         copyButton = new QPushButton(KPerfWidget);
529         copyButton->setObjectName(QStringLiteral("copyButton"));
530
531         horizontalLayout_2->addWidget(copyButton);
532
533         horizontalSpacer_8 = new QSpacerItem(40, 20, QSizePolicy::Expanding,
    QSizePolicy::Minimum);
534
535         horizontalLayout_2->addItem(horizontalSpacer_8);
536
537
538         gridLayout->addLayout(horizontalLayout_2, 3, 10, 1, 3);
539
540         horizontalLayout = new QHBoxLayout();
541         horizontalLayout->setObjectName(QStringLiteral("horizontalLayout"));
542         label_2 = new QLabel(KPerfWidget);
543         label_2->setObjectName(QStringLiteral("label_2"));
544
545         horizontalLayout->addWidget(label_2);
546
547         horizontalSpacer_4 = new QSpacerItem(10, 20, QSizePolicy::Fixed,
    QSizePolicy::Minimum);
548
549         horizontalLayout->addItem(horizontalSpacer_4);
550
551         intensityBox = new QSpinBox(KPerfWidget);
552         intensityBox->setObjectName(QStringLiteral("intensityBox"));
553         intensityBox->setMaximum(100);
554         intensityBox->setValue(100);
555
556         horizontalLayout->addWidget(intensityBox);
557
558         horizontalSpacer_3 = new QSpacerItem(40, 20, QSizePolicy::Expanding,
    QSizePolicy::Minimum);
559
560         horizontalLayout->addItem(horizontalSpacer_3);
561
562
563         gridLayout->addLayout(horizontalLayout, 3, 1, 1, 9);
564
565         k95Label5 = new QLabel(KPerfWidget);
566         k95Label5->setObjectName(QStringLiteral("k95Label5"));
567
568         gridLayout->addWidget(k95Label5, 10, 8, 1, 1);
569
570         line_2 = new QFrame(KPerfWidget);
571         line_2->setObjectName(QStringLiteral("line_2"));
572         line_2->setFrameShape(QFrame::HLine);
573         line_2->setFrameShadow(QFrame::Sunken);
574
575         gridLayout->addWidget(line_2, 14, 0, 1, 13);
576
577         k95Label6 = new QLabel(KPerfWidget);
578         k95Label6->setObjectName(QStringLiteral("k95Label6"));
579         k95Label6->setFont(font);
580
581         gridLayout->addWidget(k95Label6, 11, 1, 1, 10);
582
583         QWidget::setTabOrder(intensityBox, copyButton);
584         QWidget::setTabOrder(copyButton, modeBox);
585         QWidget::setTabOrder(modeBox, modeColorOn);
586         QWidget::setTabOrder(modeColorOn, modeColorOff);
587         QWidget::setTabOrder(modeColorOff, macroBox);
588         QWidget::setTabOrder(macroBox, macroColorOn);
589         QWidget::setTabOrder(macroColorOn, macroColorOff);
590         QWidget::setTabOrder(macroColorOff, lightBox);
591         QWidget::setTabOrder(lightBox, lightColor3);
592         QWidget::setTabOrder(lightColor3, lightColor2);
593         QWidget::setTabOrder(lightColor2, lightColor1);
594         QWidget::setTabOrder(lightColor1, lockBox);
595         QWidget::setTabOrder(lockBox, lockColorOn);
596         QWidget::setTabOrder(lockColorOn, lockColorOff);
597         QWidget::setTabOrder(lockColorOff, muteBox);
598         QWidget::setTabOrder(muteBox, muteColorOn);
599         QWidget::setTabOrder(muteColorOn, muteColorOff);
600         QWidget::setTabOrder(muteColorOff, muteColorNA);
601         QWidget::setTabOrder(muteColorNA, numBox);
602         QWidget::setTabOrder(numBox, numColorOn);
603         QWidget::setTabOrder(numColorOn, numColorOff);
604         QWidget::setTabOrder(numColorOff, capsBox);
605         QWidget::setTabOrder(capsBox, capsColorOn);
606         QWidget::setTabOrder(capsColorOn, capsColorOff);
```

```
607         QWidget::setTabOrder(capsColorOff, scrollBox);
608         QWidget::setTabOrder(scrollBox, scrollColorOn);
609         QWidget::setTabOrder(scrollColorOn, scrollColorOff);
610 
611         retranslateUi(KPerfWidget);
612 
613         QMetaObject::connectSlotsByName(KPerfWidget);
614     } // setupUi
```

Here is the call graph for this function:



Here is the caller graph for this function:

12.44.3 Field Documentation

12.44.3.1 QComboBox* Ui_KPerfWidget::capsBox [inherited]

Definition at line 74 of file ui_kperfwidget.h.

Referenced by KPerfWidget::KPerfWidget(), Ui_KPerfWidget::retranslateUi(), and Ui_KPerfWidget::setupUi().

12.44.3.2 ColorButton* Ui_KPerfWidget::capsColorOff [inherited]

Definition at line 66 of file ui_kperfwidget.h.

Referenced by KPerfWidget::KPerfWidget(), Ui_KPerfWidget::retranslateUi(), and Ui_KPerfWidget::setupUi().

12.44.3.3 ColorButton* Ui_KPerfWidget::capsColorOn [inherited]

Definition at line 68 of file ui_kperfwidget.h.

Referenced by KPerfWidget::KPerfWidget(), Ui_KPerfWidget::retranslateUi(), and Ui_KPerfWidget::setupUi().

12.44.3.4 QPushButton* Ui_KPerfWidget::copyButton [inherited]

Definition at line 100 of file ui_kperfwidget.h.

Referenced by Ui_KPerfWidget::retranslateUi(), and Ui_KPerfWidget::setupUi().

12.44.3.5 QGridLayout* Ui_KPerfWidget::gridLayout [inherited]

Definition at line 34 of file ui_kperfwidget.h.

Referenced by Ui_KPerfWidget::setupUi().

12.44.3.6 QHBoxLayout* Ui_KPerfWidget::horizontalLayout [inherited]

Definition at line 102 of file ui_kperfwidget.h.

Referenced by Ui_KPerfWidget::setupUi().

12.44.3.7 QHBoxLayout* Ui_KPerfWidget::horizontalLayout_2 [inherited]

Definition at line 99 of file ui_kperfwidget.h.

Referenced by Ui_KPerfWidget::setupUi().

12.44.3.8 QSpacerItem* Ui_KPerfWidget::horizontalSpacer [inherited]

Definition at line 44 of file ui_kperfwidget.h.

Referenced by Ui_KPerfWidget::setupUi().

12.44.3.9 QSpacerItem* Ui_KPerfWidget::horizontalSpacer_2 [inherited]

Definition at line 92 of file ui_kperfwidget.h.

Referenced by Ui_KPerfWidget::setupUi().

12.44.3.10 QSpacerItem* Ui_KPerfWidget::horizontalSpacer_3 [inherited]

Definition at line 106 of file ui_kperfwidget.h.

Referenced by Ui_KPerfWidget::setupUi().

12.44.3.11 QSpacerItem* Ui_KPerfWidget::horizontalSpacer_4 [inherited]

Definition at line 104 of file ui_kperfwidget.h.

Referenced by Ui_KPerfWidget::setupUi().

12.44.3.12 QSpacerItem* Ui_KPerfWidget::horizontalSpacer_5 [inherited]

Definition at line 39 of file ui_kperfwidget.h.

Referenced by Ui_KPerfWidget::setupUi().

12.44.3.13 QSpacerItem* Ui_KPerfWidget::horizontalSpacer_6 [inherited]

Definition at line 60 of file ui_kperfwidget.h.

Referenced by Ui_KPerfWidget::setupUi().

12.44.3.14 QSpacerItem* Ui_KPerfWidget::horizontalSpacer_7 [inherited]

Definition at line 53 of file ui_kperfwidget.h.

Referenced by Ui_KPerfWidget::setupUi().

12.44.3.15 QSpacerItem* Ui_KPerfWidget::horizontalSpacer_8 [inherited]

Definition at line 101 of file ui_kperfwidget.h.

Referenced by Ui_KPerfWidget::setupUi().

12.44.3.16 QSpinBox* Ui_KPerfWidget::intensityBox [inherited]

Definition at line 105 of file ui_kperfwidget.h.

Referenced by Ui_KPerfWidget::retranslateUi(), KPerfWidget::setPerf(), and Ui_KPerfWidget::setupUi().

12.44.3.17 QLabel* Ui_KPerfWidget::k95Label1 [inherited]

Definition at line 98 of file ui_kperfwidget.h.

Referenced by KPerfWidget::KPerfWidget(), Ui_KPerfWidget::retranslateUi(), and Ui_KPerfWidget::setupUi().

12.44.3.18 QLabel* Ui_KPerfWidget::k95Label2 [inherited]

Definition at line 40 of file ui_kperfwidget.h.

Referenced by KPerfWidget::KPerfWidget(), Ui_KPerfWidget::retranslateUi(), and Ui_KPerfWidget::setupUi().

12.44.3.19 QLabel* Ui_KPerfWidget::k95Label3 [inherited]

Definition at line 81 of file ui_kperfwidget.h.

Referenced by KPerfWidget::KPerfWidget(), Ui_KPerfWidget::retranslateUi(), and Ui_KPerfWidget::setupUi().

12.44.3.20 QLabel* Ui_KPerfWidget::k95Label4 [inherited]

Definition at line 94 of file ui_kperfwidget.h.

Referenced by KPerfWidget::KPerfWidget(), Ui_KPerfWidget::retranslateUi(), and Ui_KPerfWidget::setupUi().

12.44.3.21 QLabel* Ui_KPerfWidget::k95Label5 [inherited]

Definition at line 107 of file ui_kperfwidget.h.

Referenced by KPerfWidget::KPerfWidget(), Ui_KPerfWidget::retranslateUi(), and Ui_KPerfWidget::setupUi().

12.44.3.22 QLabel* Ui_KPerfWidget::k95Label6 [inherited]

Definition at line 109 of file ui_kperfwidget.h.

Referenced by KPerfWidget::KPerfWidget(), Ui_KPerfWidget::retranslateUi(), and Ui_KPerfWidget::setupUi().

12.44.3.23 QFrame* Ui_KPerfWidget::k95Line [inherited]

Definition at line 85 of file ui_kperfwidget.h.

Referenced by KPerfWidget::KPerfWidget(), and Ui_KPerfWidget::setupUi().

12.44.3.24 QWidget* Ui_KPerfWidget::k95Spacer [inherited]

Definition at line 35 of file ui_kperfwidget.h.

Referenced by KPerfWidget::KPerfWidget(), and Ui_KPerfWidget::setupUi().

12.44.3.25 QLabel* Ui_KPerfWidget::label [inherited]

Definition at line 59 of file ui_kperfwidget.h.

Referenced by Ui_KPerfWidget::retranslateUi(), and Ui_KPerfWidget::setupUi().

12.44.3.26 QLabel* Ui_KPerfWidget::label_10 [inherited]

Definition at line 36 of file ui_kperfwidget.h.

Referenced by Ui_KPerfWidget::retranslateUi(), and Ui_KPerfWidget::setupUi().

12.44.3.27 QLabel* Ui_KPerfWidget::label_11 [inherited]

Definition at line 96 of file ui_kperfwidget.h.

Referenced by Ui_KPerfWidget::retranslateUi(), and Ui_KPerfWidget::setupUi().

12.44.3.28 QLabel* Ui_KPerfWidget::label_12 [inherited]

Definition at line 41 of file ui_kperfwidget.h.

Referenced by Ui_KPerfWidget::retranslateUi(), KPerfWidget::setPerf(), and Ui_KPerfWidget::setupUi().

12.44.3.29 QLabel* Ui_KPerfWidget::label_14 [inherited]

Definition at line 90 of file ui_kperfwidget.h.

Referenced by Ui_KPerfWidget::retranslateUi(), KPerfWidget::setPerf(), and Ui_KPerfWidget::setupUi().

12.44.3.30 QLabel* Ui_KPerfWidget::label_15 [inherited]

Definition at line 93 of file ui_kperfwidget.h.

Referenced by Ui_KPerfWidget::retranslateUi(), and Ui_KPerfWidget::setupUi().

12.44.3.31 QLabel* Ui_KPerfWidget::label_16 [inherited]

Definition at line 89 of file ui_kperfwidget.h.

Referenced by Ui_KPerfWidget::retranslateUi(), and Ui_KPerfWidget::setupUi().

12.44.3.32 QLabel* Ui_KPerfWidget::label_17 [inherited]

Definition at line 95 of file ui_kperfwidget.h.

Referenced by Ui_KPerfWidget::retranslateUi(), KPerfWidget::setPerf(), and Ui_KPerfWidget::setupUi().

12.44.3.33 QLabel* Ui_KPerfWidget::label_18 [inherited]

Definition at line 91 of file ui_kperfwidget.h.

Referenced by Ui_KPerfWidget::retranslateUi(), and Ui_KPerfWidget::setupUi().

12.44.3.34 QLabel* Ui_KPerfWidget::label_19 [inherited]

Definition at line 84 of file ui_kperfwidget.h.

Referenced by Ui_KPerfWidget::retranslateUi(), and Ui_KPerfWidget::setupUi().

12.44.3.35 QLabel* Ui_KPerfWidget::label_2 [inherited]

Definition at line 103 of file ui_kperfwidget.h.

Referenced by Ui_KPerfWidget::retranslateUi(), and Ui_KPerfWidget::setupUi().

12.44.3.36 QLabel* Ui_KPerfWidget::label_20 [inherited]

Definition at line 88 of file ui_kperfwidget.h.

Referenced by Ui_KPerfWidget::retranslateUi(), and Ui_KPerfWidget::setupUi().

12.44.3.37 QLabel* Ui_KPerfWidget::label_21 [inherited]

Definition at line 86 of file ui_kperfwidget.h.

Referenced by Ui_KPerfWidget::retranslateUi(), and Ui_KPerfWidget::setupUi().

12.44.3.38 QLabel* Ui_KPerfWidget::label_22 [inherited]

Definition at line 70 of file ui_kperfwidget.h.

Referenced by Ui_KPerfWidget::retranslateUi(), and Ui_KPerfWidget::setupUi().

12.44.3.39 QLabel* Ui_KPerfWidget::label_23 [inherited]

Definition at line 97 of file ui_kperfwidget.h.

Referenced by Ui_KPerfWidget::retranslateUi(), and Ui_KPerfWidget::setupUi().

12.44.3.40 QLabel* Ui_KPerfWidget::label_24 [inherited]

Definition at line 87 of file ui_kperfwidget.h.

Referenced by Ui_KPerfWidget::retranslateUi(), and Ui_KPerfWidget::setupUi().

12.44.3.41 QLabel* Ui_KPerfWidget::label_3 [inherited]

Definition at line 79 of file ui_kperfwidget.h.

Referenced by Ui_KPerfWidget::retranslateUi(), and Ui_KPerfWidget::setupUi().

12.44.3.42 QLabel* Ui_KPerfWidget::label_4 [inherited]

Definition at line 62 of file ui_kperfwidget.h.

Referenced by Ui_KPerfWidget::retranslateUi(), and Ui_KPerfWidget::setupUi().

12.44.3.43 QLabel* Ui_KPerfWidget::label_5 [inherited]

Definition at line 50 of file ui_kperfwidget.h.

Referenced by Ui_KPerfWidget::retranslateUi(), and Ui_KPerfWidget::setupUi().

12.44.3.44 QLabel* Ui_KPerfWidget::label_6 [inherited]

Definition at line 58 of file ui_kperfwidget.h.

Referenced by Ui_KPerfWidget::retranslateUi(), and Ui_KPerfWidget::setupUi().

12.44.3.45 QLabel* Ui_KPerfWidget::label_7 [inherited]

Definition at line 72 of file ui_kperfwidget.h.

Referenced by Ui_KPerfWidget::retranslateUi(), and Ui_KPerfWidget::setupUi().

12.44.3.46 QLabel* Ui_KPerfWidget::label_8 [inherited]

Definition at line 54 of file ui_kperfwidget.h.

Referenced by Ui_KPerfWidget::retranslateUi(), and Ui_KPerfWidget::setupUi().

12.44.3.47 QCheckBox* Ui_KPerfWidget::lightBox [inherited]

Definition at line 55 of file ui_kperfwidget.h.

Referenced by KPerfWidget::KPerfWidget(), Ui_KPerfWidget::retranslateUi(), and Ui_KPerfWidget::setupUi().

12.44.3.48 ColorButton* Ui_KPerfWidget::lightColor1 [inherited]

Definition at line 56 of file ui_kperfwidget.h.

Referenced by KPerfWidget::KPerfWidget(), Ui_KPerfWidget::retranslateUi(), and Ui_KPerfWidget::setupUi().

12.44.3.49 ColorButton* Ui_KPerfWidget::lightColor2 [inherited]

Definition at line 52 of file ui_kperfwidget.h.

Referenced by KPerfWidget::KPerfWidget(), Ui_KPerfWidget::retranslateUi(), and Ui_KPerfWidget::setupUi().

12.44.3.50 ColorButton* Ui_KPerfWidget::lightColor3 [inherited]

Definition at line 43 of file ui_kperfwidget.h.

Referenced by KPerfWidget::KPerfWidget(), Ui_KPerfWidget::retranslateUi(), and Ui_KPerfWidget::setupUi().

12.44.3.51 QFrame* Ui_KPerfWidget::line [inherited]

Definition at line 38 of file ui_kperfwidget.h.

Referenced by Ui_KPerfWidget::setupUi().

12.44.3.52 QFrame* Ui_KPerfWidget::line_2 [inherited]

Definition at line 108 of file ui_kperfwidget.h.

Referenced by Ui_KPerfWidget::setupUi().

12.44.3.53 QFrame* Ui_KPerfWidget::line_3 [inherited]

Definition at line 69 of file ui_kperfwidget.h.

Referenced by Ui_KPerfWidget::setupUi().

12.44.3.54 QCheckBox* Ui_KPerfWidget::lockBox [inherited]

Definition at line 45 of file ui_kperfwidget.h.

Referenced by KPerfWidget::KPerfWidget(), Ui_KPerfWidget::retranslateUi(), and Ui_KPerfWidget::setupUi().

12.44.3.55 ColorButton* Ui_KPerfWidget::lockColorOff [inherited]

Definition at line 37 of file ui_kperfwidget.h.

Referenced by KPerfWidget::KPerfWidget(), Ui_KPerfWidget::retranslateUi(), and Ui_KPerfWidget::setupUi().

12.44.3.56 ColorButton* Ui_KPerfWidget::lockColorOn [inherited]

Definition at line 61 of file ui_kperfwidget.h.

Referenced by KPerfWidget::KPerfWidget(), Ui_KPerfWidget::retranslateUi(), and Ui_KPerfWidget::setupUi().

12.44.3.57 QCheckBox* Ui_KPerfWidget::macroBox [inherited]

Definition at line 47 of file ui_kperfwidget.h.

Referenced by KPerfWidget::KPerfWidget(), Ui_KPerfWidget::retranslateUi(), and Ui_KPerfWidget::setupUi().

12.44.3.58 ColorButton* Ui_KPerfWidget::macroColorOff [inherited]

Definition at line 78 of file ui_kperfwidget.h.

Referenced by KPerfWidget::KPerfWidget(), Ui_KPerfWidget::retranslateUi(), and Ui_KPerfWidget::setupUi().

12.44.3.59 ColorButton* Ui_KPerfWidget::macroColorOn [inherited]

Definition at line 82 of file ui_kperfwidget.h.

Referenced by KPerfWidget::KPerfWidget(), Ui_KPerfWidget::retranslateUi(), and Ui_KPerfWidget::setupUi().

12.44.3.60 QCheckBox* Ui_KPerfWidget::modeBox [inherited]

Definition at line 65 of file ui_kperfwidget.h.

Referenced by KPerfWidget::KPerfWidget(), Ui_KPerfWidget::retranslateUi(), and Ui_KPerfWidget::setupUi().

12.44.3.61 ColorButton* Ui_KPerfWidget::modeColorOff [inherited]

Definition at line 42 of file ui_kperfwidget.h.

Referenced by KPerfWidget::KPerfWidget(), Ui_KPerfWidget::retranslateUi(), and Ui_KPerfWidget::setupUi().

12.44.3.62 ColorButton* Ui_KPerfWidget::modeColorOn [inherited]

Definition at line 46 of file ui_kperfwidget.h.

Referenced by KPerfWidget::KPerfWidget(), Ui_KPerfWidget::retranslateUi(), and Ui_KPerfWidget::setupUi().

12.44.3.63 QCheckBox* Ui_KPerfWidget::muteBox [inherited]

Definition at line 76 of file ui_kperfwidget.h.

Referenced by KPerfWidget::KPerfWidget(), Ui_KPerfWidget::retranslateUi(), KPerfWidget::setPerf(), and Ui_KPerfWidget::setupUi().

12.44.3.64 ColorButton* Ui_KPerfWidget::muteColorNA [inherited]

Definition at line 83 of file ui_kperfwidget.h.

Referenced by KPerfWidget::KPerfWidget(), Ui_KPerfWidget::retranslateUi(), KPerfWidget::setPerf(), and Ui_KPerfWidget::setupUi().

12.44.3.65 ColorButton* Ui_KPerfWidget::muteColorOff [inherited]

Definition at line 80 of file ui_kperfwidget.h.

Referenced by KPerfWidget::KPerfWidget(), Ui_KPerfWidget::retranslateUi(), KPerfWidget::setPerf(), and Ui_KPerfWidget::setupUi().

12.44.3.66 ColorButton* Ui_KPerfWidget::muteColorOn [inherited]

Definition at line 77 of file ui_kperfwidget.h.

Referenced by KPerfWidget::KPerfWidget(), Ui_KPerfWidget::retranslateUi(), KPerfWidget::setPerf(), and Ui_KPerfWidget::setupUi().

12.44.3.67 QComboBox* Ui_KPerfWidget::numBox [inherited]

Definition at line 49 of file ui_kperfwidget.h.

Referenced by KPerfWidget::KPerfWidget(), Ui_KPerfWidget::retranslateUi(), and Ui_KPerfWidget::setupUi().

12.44.3.68 ColorButton* Ui_KPerfWidget::numColorOff [inherited]

Definition at line 67 of file ui_kperfwidget.h.

Referenced by KPerfWidget::KPerfWidget(), Ui_KPerfWidget::retranslateUi(), and Ui_KPerfWidget::setupUi().

12.44.3.69 ColorButton* Ui_KPerfWidget::numColorOn [inherited]

Definition at line 48 of file ui_kperfwidget.h.

Referenced by KPerfWidget::KPerfWidget(), Ui_KPerfWidget::retranslateUi(), and Ui_KPerfWidget::setupUi().

12.44.3.70 QComboBox* Ui_KPerfWidget::scrollBox [inherited]

Definition at line 71 of file ui_kperfwidget.h.

Referenced by KPerfWidget::KPerfWidget(), Ui_KPerfWidget::retranslateUi(), and Ui_KPerfWidget::setupUi().

12.44.3.71 ColorButton* Ui_KPerfWidget::scrollColorOff [inherited]

Definition at line 75 of file ui_kperfwidget.h.

Referenced by KPerfWidget::KPerfWidget(), Ui_KPerfWidget::retranslateUi(), and Ui_KPerfWidget::setupUi().

12.44.3.72 ColorButton* Ui_KPerfWidget::scrollColorOn [inherited]

Definition at line 73 of file ui_kperfwidget.h.

Referenced by KPerfWidget::KPerfWidget(), Ui_KPerfWidget::retranslateUi(), and Ui_KPerfWidget::setupUi().

12.44.3.73 QSpacerItem* Ui_KPerfWidget::verticalSpacer [inherited]

Definition at line 63 of file ui_kperfwidget.h.

Referenced by Ui_KPerfWidget::setupUi().

12.44.3.74 QSpacerItem* Ui_KPerfWidget::verticalSpacer_2 [inherited]

Definition at line 51 of file ui_kperfwidget.h.

Referenced by Ui_KPerfWidget::setupUi().

12.44.3.75 QSpacerItem* Ui_KPerfWidget::verticalSpacer_3 [inherited]

Definition at line 64 of file ui_kperfwidget.h.

Referenced by Ui_KPerfWidget::setupUi().

12.44.3.76 QSpacerItem* Ui_KPerfWidget::verticalSpacer_4 [inherited]

Definition at line 57 of file ui_kperfwidget.h.

Referenced by Ui_KPerfWidget::setupUi().

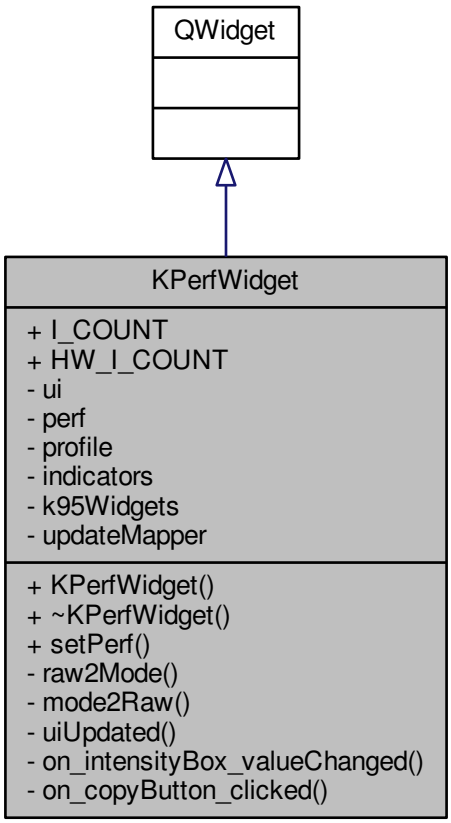
The documentation for this class was generated from the following file:

- src/ckb/ui_kperfwidget.h

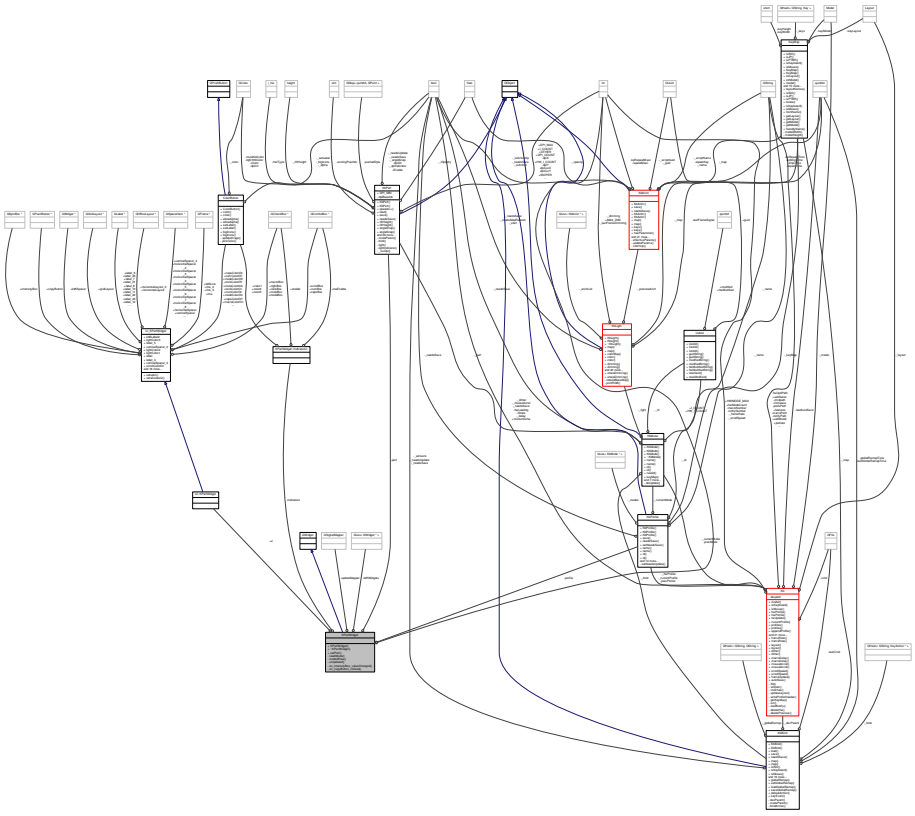
12.45 KPerfWidget Class Reference

`#include <src/ckb/kperfwidget.h>`

Inheritance diagram for KPerfWidget:



Collaboration diagram for KPerfWidget:



Data Structures

- struct IndicatorUi

Public Types

- typedef KbPerf::indicator indicator
- typedef KbPerf::i_hw i_hw

Public Member Functions

- KPerfWidget (QWidget *parent=0)

  KPerfWidget::KPerfWidget sets up the UI for Keyboard Performace panel.
- ~KPerfWidget ()

  KPerfWidget::~KPerfWidget nothing unusual - just delete the ui object.
- void setPerf (KbPerf *newPerf, KbProfile *newProfile)

Static Public Attributes

- static const int I_COUNT = KbPerf::I_COUNT
- static const int HW_I_COUNT = KbPerf::HW_I_COUNT

Private Types

- enum HwMode {
  NORMAL, ALWAYS_ON, ALWAYS_OFF, RGB,
  BOTH }

Private Slots

- void uiUpdated (int index)
- void on_intensityBox_valueChanged (int arg1)
- void on_copyButton_clicked ()

Private Member Functions

- HwMode raw2Mode (bool sw_enable, i_hw hw_enable)

  KPerfWidget::raw2Mode return hardware mode depending on setiings in sw_enable and hw_enable.
- void mode2Raw (HwMode mode, bool &sw_enable, i_hw &hw_enable)

  KPerfWidget::mode2Raw Set values of sw_enable and hw_enable to hte value corresponding to input var mode.

Private Attributes

- Ui::KPerfWidget * ui
- KbPerf * perf
- KbProfile * profile
- IndicatorUi indicators [I_COUNT]
- QList< QWidget * > k95Widgets
- QSignalMapper updateMapper

12.45.1 Detailed Description

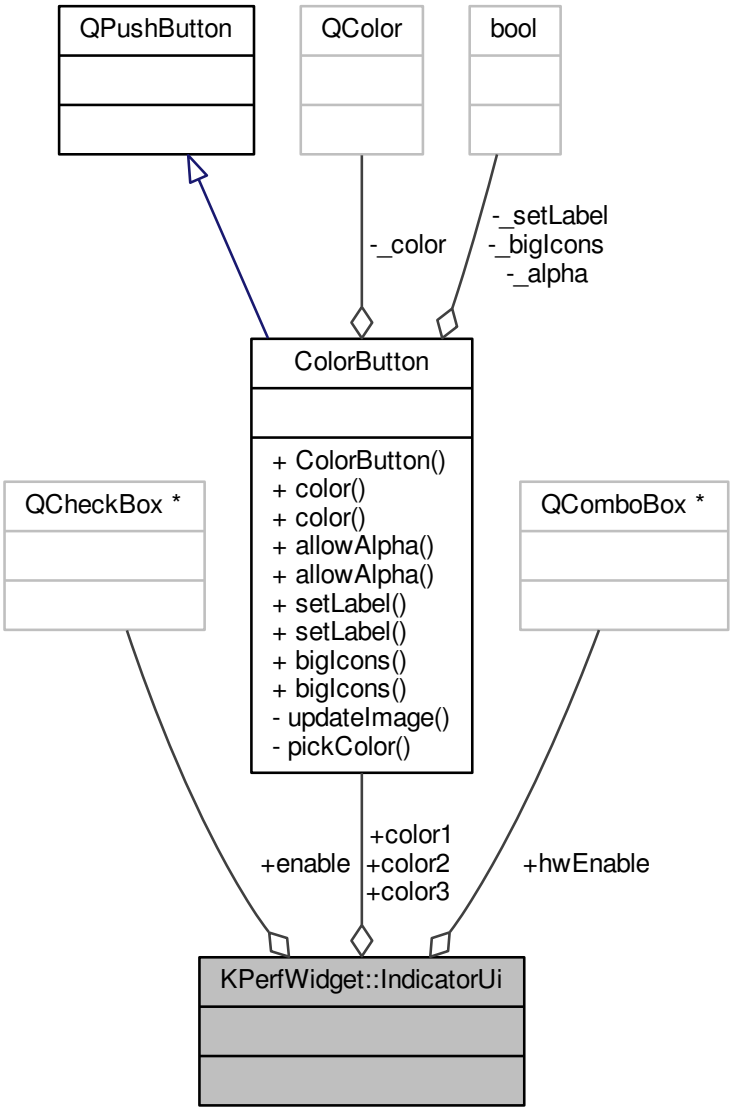
Definition at line 16 of file kperfwidget.h.

12.45.2 Data Structure Documentation

12.45.2.1 struct KPerfWidget::IndicatorUi

Definition at line 54 of file kperfwidget.h.

Collaboration diagram for KPerfWidget::IndicatorUi:



Data Fields

ColorButton *	color1	
ColorButton *	color2	
ColorButton *	color3	
QCheckBox *	enable	
QComboBox *	hwEnable	

12.45.3 Member Typedef Documentation

12.45.3.1 typedef KbPerf::i_hw KPerfWidget::i_hw

Definition at line 29 of file kperfwidget.h.

12.45.3.2 typedef KbPerf::indicator KPerfWidget::indicator

Definition at line 28 of file kperfwidget.h.

12.45.4 Member Enumeration Documentation

12.45.4.1 enum KPerfWidget::HwMode [private]

Enumerator

NORMAL

ALWAYS_ON

ALWAYS_OFF

RGB

BOTH

Definition at line 38 of file kperfwidget.h.

```
38                  {
39          NORMAL,
40          ALWAYS_ON,
41          ALWAYS_OFF,
42          RGB,
43          BOTH
44      };
```

12.45.5 Constructor & Destructor Documentation

12.45.5.1 KPerfWidget::KPerfWidget (QWidget * *parent* = 0) [explicit]

Parameters

parent	

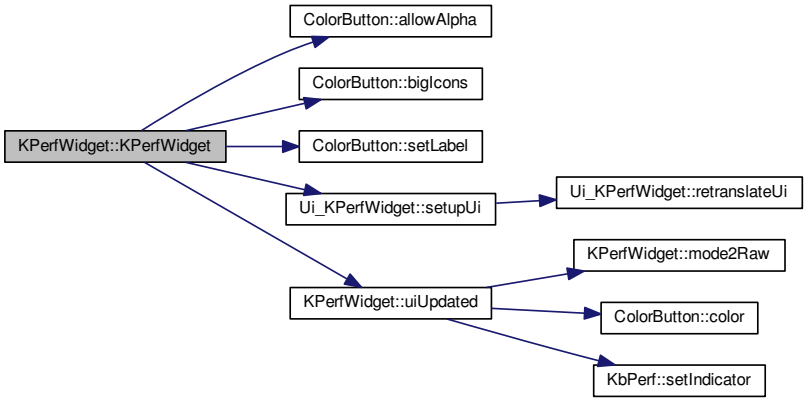
Definition at line 10 of file kperfwidget.cpp.

References ColorButton::allowAlpha(), ColorButton::bigIcons(), Ui_KPerfWidget::capsBox, Ui_KPerfWidget::capsColorOff, Ui_KPerfWidget::capsColorOn, KPerfWidget::IndicatorUi::color1, KPerfWidget::IndicatorUi::color2, KPerfWidget::IndicatorUi::color3, KPerfWidget::IndicatorUi::enable, KPerfWidget::IndicatorUi::hwEnable, I_COUNT, indicators, Ui_KPerfWidget::k95Label1, Ui_KPerfWidget::k95Label2, Ui_KPerfWidget::k95Label3, Ui_KPerfWidget::k95Label4, Ui_KPerfWidget::k95Label5, Ui_KPerfWidget::k95Label6, Ui_KPerfWidget::k95Line, Ui_KPerfWidget::k95Spacer, k95Widgets, Ui_KPerfWidget::lightBox, Ui_KPerfWidget::lightColor1, Ui_KPerfWidget::lightColor2, Ui_KPerfWidget::lightColor3, Ui_KPerfWidget::lockBox, Ui_KPerfWidget::lockColorOff, Ui_KPerfWidget::lockColorOn, Ui_KPerfWidget::macroBox, Ui_KPerfWidget::macroColorOff, Ui_KPerfWidget::macroColorOn, Ui_KPerfWidget::modeBox, Ui_KPerfWidget::modeColorOff, Ui_KPerfWidget::modeColorOn, Ui_KPerfWidget::muteBox, Ui_KPerfWidget::muteColorNA, Ui_KPerfWidget::muteColorOff, Ui_KPerfWidget::muteColorOn, Ui_KPerfWidget::numBox, Ui_KPerfWidget::numColorOff, Ui_KPerfWidget::numColorOn, Ui_KPerfWidget::scrollBox, Ui_KPerfWidget::scrollColorOff, Ui_KPerfWidget::scrollColorOn, ColorButton::setLabel(), Ui_KPerfWidget::setupUi(), ui, uiUpdated(), and updateMapper.

```
10                                      :
11      QWidget(parent),
12      ui(new Ui::KPerfWidget)
13  {
14      ui->setupUi(this);
15      // Set up indicators
```

```
16      indicators[0].enable = 0; indicators[0].hwEnable =
       ui->numBox; indicators[0].color1 = ui->numColorOn;
       indicators[0].color2 = ui->numColorOff; indicators[0].
       color3 = 0;
17      indicators[1].enable = 0; indicators[1].hwEnable =
       ui->capsBox; indicators[1].color1 = ui->capsColorOn;
       indicators[1].color2 = ui->capsColorOff;
       indicators[1].color3 = 0;
18      indicators[2].enable = 0; indicators[2].hwEnable =
       ui->scrollBox; indicators[2].color1 = ui->
       scrollColorOn; indicators[2].color2 = ui->
       scrollColorOff; indicators[2].color3 = 0;
19      indicators[3].enable = ui->modeBox; indicators[3].
       hwEnable = 0; indicators[3].color1 = ui->modeColorOn;
       indicators[3].color2 = ui->modeColorOff;
       indicators[3].color3 = 0;
20      indicators[4].enable = ui->macroBox; indicators[4].
       hwEnable = 0; indicators[4].color1 = ui->macroColorOn;
       indicators[4].color2 = ui->macroColorOff;
       indicators[4].color3 = 0;
21      indicators[5].enable = ui->lightBox; indicators[5].
       hwEnable = 0; indicators[5].color1 = ui->lightColor1;
       indicators[5].color2 = ui->lightColor2; indicators[5].
       color3 = ui->lightColor3;
22      indicators[6].enable = ui->lockBox; indicators[6].
       hwEnable = 0; indicators[6].color1 = ui->lockColorOn;
       indicators[6].color2 = ui->lockColorOff;
       indicators[6].color3 = 0;
23      indicators[7].enable = ui->muteBox; indicators[7].
       hwEnable = 0; indicators[7].color1 = ui->muteColorOn;
       indicators[7].color2 = ui->muteColorOff;
       indicators[7].color3 = ui->muteColorNA;
24      for(int i = 0; i < I_COUNT; i++){
25          indicators[i].color1->setLabel(false);
26          indicators[i].color1->bigIcons(true);
27          indicators[i].color1->allowAlpha(true);
28          indicators[i].color2->setLabel(false);
29          indicators[i].color2->bigIcons(true);
30          indicators[i].color2->allowAlpha(true);
31          if(indicators[i].color3){
32              indicators[i].color3->setLabel(false);
33              indicators[i].color3->bigIcons(true);
34              indicators[i].color3->allowAlpha(true);
35          }
36          // Map signals
37          if(indicators[i].enable){
38              connect(indicators[i].enable, SIGNAL(clicked(bool)), &
       updateMapper, SLOT(map()));
39              updateMapper.setMapping(indicators[i].enable, i);
40          }
41          if(indicators[i].hwEnable){
42              connect(indicators[i].hwEnable, SIGNAL(activated(int)), &
       updateMapper, SLOT(map()));
43              updateMapper.setMapping(indicators[i].hwEnable, i);
44          }
45          connect(indicators[i].color1, SIGNAL(colorChanged(QColor)), &
       updateMapper, SLOT(map()));
46          updateMapper.setMapping(indicators[i].color1, i);
47          connect(indicators[i].color2, SIGNAL(colorChanged(QColor)), &
       updateMapper, SLOT(map()));
48          updateMapper.setMapping(indicators[i].color2, i);
49          if(indicators[i].color3){
50              connect(indicators[i].color3, SIGNAL(colorChanged(QColor)), &
       updateMapper, SLOT(map()));
51              updateMapper.setMapping(indicators[i].color3, i);
52          }
53      }
54      connect(&updateMapper, SIGNAL(mapped(int)), this, SLOT(uiUpdated(int)));
55      k95Widgets << ui->modeBox << ui->modeColorOn <<
       ui->modeColorOff << ui->macroBox << ui->macroColorOn <<
       ui->macroColorOff << ui->k95Label1 << ui->k95Label2 <<
       ui->k95Label3 << ui->k95Label4 << ui->k95Label5 <<
       ui->k95Label6 << ui->k95Line << ui->k95Spacer;
56  }
```

Here is the call graph for this function:



12.45.5.2 KPerfWidget::~KPerfWidget ( )

Definition at line 61 of file kperfwidget.cpp.

References ui.

```
61                          {
62     delete ui;
63 }
```

12.45.6 Member Function Documentation

12.45.6.1 void KPerfWidget::mode2Raw (HwMode *mode*, bool & *sw_enable*, i_hw & *hw_enable*) [private]

Parameters

[IN]	mode
[OUT]	sw_enable
[OUT]	hw_enable
[IN]	mode
[OUT]	sw_enable
[OUT]	hw_enable mode determines how colors at a key and separate indicators should be handled: Mode | sw | hw _ | _ | _ NORMAL | No color change at the key | use the separate indicator depending on key state ALWAYS_ON | No color change at the key | switch on separate indicator ALWAYS_OFF | No color change at the key | switch off separate indicator RGB | use color change at the key depending on color sliders | switch off separate indicator BOTH | use color change at the key depending on color sliders | use the separate indicator depending on key state

Definition at line 100 of file kperfwidget.cpp.

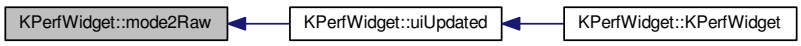
References ALWAYS_OFF, ALWAYS_ON, BOTH, NORMAL, KbPerf::NORMAL, KbPerf::OFF, KbPerf::ON, and RGB.

Referenced by uiUpdated().

```
100                                                                 {
101     switch(mode){
102     case NORMAL:
```

```
103         sw_enable = false;
104         hw_enable = KbPerf::NORMAL;
105         break;
106     case ALWAYS_ON:
107         sw_enable = false;
108         hw_enable = KbPerf::ON;
109         break;
110     case ALWAYS_OFF:
111         sw_enable = false;
112         hw_enable = KbPerf::OFF;
113         break;
114     case RGB:
115         sw_enable = true;
116         hw_enable = KbPerf::OFF;
117         break;
118     case BOTH:
119         sw_enable = true;
120         hw_enable = KbPerf::NORMAL;
121         break;
122     }
123 }
```

Here is the caller graph for this function:



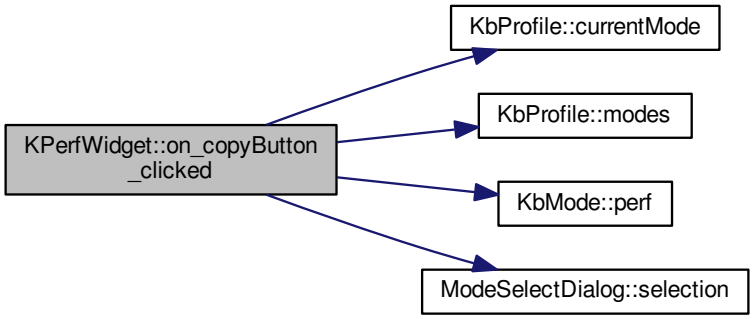
12.45.6.2 void KPerfWidget::on_copyButton_clicked ( ) [private],[slot]

Definition at line 221 of file kperfwidget.cpp.

References KbProfile::currentMode(), KbProfile::modes(), perf, KbMode::perf(), profile, and ModeSelectDialog::selection().

```
221                                     {
222     ModeSelectDialog dialog(this, profile->currentMode(),
      profile->modes(), "Copy performance settings to:");
223     if(dialog.exec() != QDialog::Accepted)
224         return;
225     QList<KbMode*> selectedModes = dialog.selection();
226     foreach(KbMode* mode, selectedModes){
227         *mode->perf() = *perf;
228     }
229 }
```

Here is the call graph for this function:



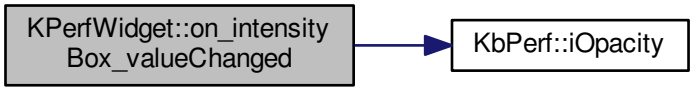
12.45.6.3 void KPerfWidget::on_intensityBox_valueChanged (int *arg1*) [private], [slot]

Definition at line 215 of file kperfwidget.cpp.

References KbPerf::iOpacity(), and perf.

```
215                                                 {
216     if(!perf)
217         return;
218     perf->iOpacity(arg1 / 100.f);
219 }
```

Here is the call graph for this function:



12.45.6.4 KPerfWidget::HwMode KPerfWidget::raw2Mode (bool *sw_enable,* i_hw *hw_enable*) [private]

Parameters

sw_enable	
hw_enable	

Returns

the mode of operation for key-coloring and separate indicators.

See Also

KPerfWidget::mode2Raw for details.

Definition at line 72 of file kperfwidget.cpp.

References ALWAYS_OFF, ALWAYS_ON, BOTH, NORMAL, KbPerf::NORMAL, KbPerf::OFF, KbPerf::ON, and RGB.

Referenced by setPerf().

```
72                                                                  {
73     if(sw_enable){
74         if(hw_enable == KbPerf::NORMAL)
75             return BOTH;
76         return RGB;
77     } else {
78         if(hw_enable == KbPerf::ON)
79             return ALWAYS_ON;
80         else if(hw_enable == KbPerf::OFF)
81             return ALWAYS_OFF;
82         return NORMAL;
83     }
84 }
```

Here is the caller graph for this function:

12.45.6.5 void KPerfWidget::setPerf (KbPerf * *newPerf,* KbProfile * *newProfile*)

Definition at line 125 of file kperfwidget.cpp.

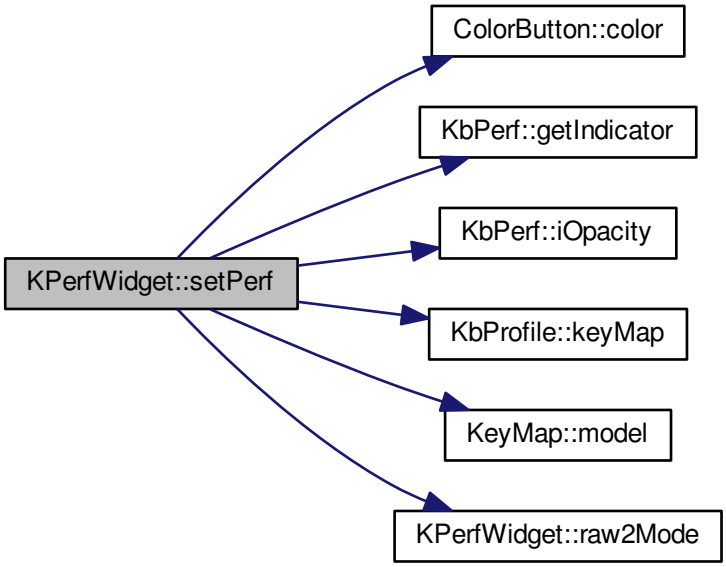
References BOTH, ColorButton::color(), KPerfWidget::IndicatorUi::color1, KPerfWidget::IndicatorUi::color2, KPerfWidget::IndicatorUi::color3, KPerfWidget::IndicatorUi::enable, KbPerf::getIndicator(), HW_I_COUNT, KPerfWidget::IndicatorUi::hwEnable, I_COUNT, indicators, Ui_KPerfWidget::intensityBox, KbPerf::iOpacity(), KeyMap::K95, k95Widgets, KbProfile::keyMap(), Ui_KPerfWidget::label_12, Ui_KPerfWidget::label_14, Ui_KPerfWidget::label_17, KeyMap::model(), Ui_KPerfWidget::muteBox, Ui_KPerfWidget::muteColorNA, Ui_KPerfWidget::muteColorOff, Ui_KPerfWidget::muteColorOn, perf, profile, raw2Mode(), RGB, KeyMap::STRAFE, and ui.

Referenced by KbWidget::modeChanged().

```
125                                                     {
126     perf = newPerf;
127     profile = newProfile;
128     // Set intensity
129     ui->intensityBox->setValue(round(perf->iOpacity() * 100.f));
130     // Set hardware indicator values
131     for(int i = 0; i < HW_I_COUNT; i++){
132         QColor c1, c2, c3;
133         bool software;
134         i_hw hardware;
135         perf->getIndicator((indicator)i, c1, c2, c3, software, hardware);
136         HwMode mode = raw2Mode(software, hardware);
137         indicators[i].hwEnable->setCurrentIndex((int)mode);
138         indicators[i].color1->color(c1);
139         indicators[i].color2->color(c2);
140         if(mode == RGB || mode == BOTH){
141             indicators[i].color1->setEnabled(true);
142             indicators[i].color2->setEnabled(true);
143         } else {
144             indicators[i].color1->setEnabled(false);
145             indicators[i].color2->setEnabled(false);
146         }
147     }
148     // Set software indicators
149     for(int i = HW_I_COUNT; i < I_COUNT; i++){
150         QColor c1, c2, c3;
151         bool software;
152         i_hw hardware;
153         perf->getIndicator((indicator)i, c1, c2, c3, software, hardware);
154         indicators[i].enable->setChecked(software);
155         indicators[i].color1->color(c1);
156         indicators[i].color2->color(c2);
157         if(indicators[i].color3) indicators[i].color3->
    color(c3);
158         if(software){
159             indicators[i].color1->setEnabled(true);
160             indicators[i].color2->setEnabled(true);
161             if(indicators[i].color3) indicators[i].color3->setEnabled(true);
162         } else {
163             indicators[i].color1->setEnabled(false);
164             indicators[i].color2->setEnabled(false);
165             if(indicators[i].color3) indicators[i].color3->setEnabled(false);
166         }
167     }
168     // Hide K95 indicators on non-K95s
169     if(profile->keyMap().model() == KeyMap::K95){
170         foreach(QWidget* w, k95Widgets)
171             w->show();
172     } else {
173         foreach(QWidget* w, k95Widgets)
174             w->hide();
175         if(profile->keyMap().model() == KeyMap::STRAFE) { // hide
```

```
        indicators unsupported by STRAFE
176             ui->muteBox->hide();
177             ui->muteColorOn->hide();
178             ui->muteColorOff->hide();
179             ui->muteColorNA->hide();
180             ui->label_12->hide();
181             ui->label_14->hide();
182             ui->label_17->hide();
183         }
184     }
185 }
```

Here is the call graph for this function:



Here is the caller graph for this function:

12.45.6.6 void KPerfWidget::uiUpdated (int *index*) [private],[slot]

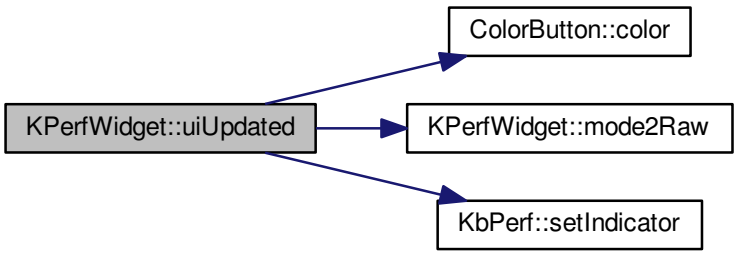
Definition at line 187 of file kperfwidget.cpp.

References ColorButton::color(), KPerfWidget::IndicatorUi::color1, KPerfWidget::IndicatorUi::color2, KPerfWidget::IndicatorUi::color3, KPerfWidget::IndicatorUi::enable, indicators, mode2Raw(), KbPerf::NONE, perf, and KbPerf::setIndicator().

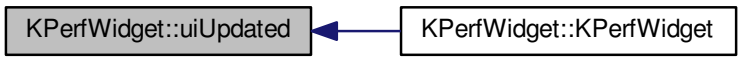
Referenced by KPerfWidget().

```
187                                     {
188     if(!perf)
189         return;
190     // Read HW/SW enable state
191     bool software;
192     i_hw hardware;
193     if(indicators[index].enable){
194         software = indicators[index].enable->isChecked();
195         hardware = KbPerf::NONE;
196     } else {
197         mode2Raw((HwMode)indicators[index].hwEnable->currentIndex(), software,
      hardware);
198     }
199     // Enable/disable color buttons based on whether or not the indicator is turned on
200     if(software){
201         indicators[index].color1->setEnabled(true);
202         indicators[index].color2->setEnabled(true);
203         if(indicators[index].color3) indicators[index].
      color3->setEnabled(true);
204     } else {
205         indicators[index].color1->setEnabled(false);
206         indicators[index].color2->setEnabled(false);
207         if(indicators[index].color3) indicators[index].
      color3->setEnabled(false);
208     }
209     QColor c1 = indicators[index].color1->color();
210     QColor c2 = indicators[index].color2->color();
211     QColor c3 = indicators[index].color3 ? indicators[index].
      color3->color() : QColor();
212     perf->setIndicator((indicator)index, c1, c2, c3, software, hardware);
213 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.45.7 Field Documentation

12.45.7.1 const int KPerfWidget::HW_I_COUNT = KbPerf::HW_I_COUNT [static]

Definition at line 27 of file kperfwidget.h.

Referenced by setPerf().

12.45.7.2 const int KPerfWidget::I_COUNT = KbPerf::I_COUNT [static]

Definition at line 26 of file kperfwidget.h.

Referenced by KPerfWidget(), and setPerf().

12.45.7.3 IndicatorUi KPerfWidget::indicators[I_COUNT] [private]

Definition at line 59 of file kperfwidget.h.

Referenced by KPerfWidget(), setPerf(), and uiUpdated().

12.45.7.4 QList<QWidget*> KPerfWidget::k95Widgets [private]

Definition at line 60 of file kperfwidget.h.

Referenced by KPerfWidget(), and setPerf().

12.45.7.5 KbPerf* KPerfWidget::perf [private]

Definition at line 34 of file kperfwidget.h.

Referenced by on_copyButton_clicked(), on_intensityBox_valueChanged(), setPerf(), and uiUpdated().

12.45.7.6 KbProfile* KPerfWidget::profile [private]

Definition at line 35 of file kperfwidget.h.

Referenced by on_copyButton_clicked(), and setPerf().

12.45.7.7 Ui::KPerfWidget* KPerfWidget::ui [private]

Definition at line 32 of file kperfwidget.h.

Referenced by KPerfWidget(), setPerf(), and ~KPerfWidget().

12.45.7.8 QSignalMapper KPerfWidget::updateMapper [private]

Definition at line 62 of file kperfwidget.h.

Referenced by KPerfWidget().

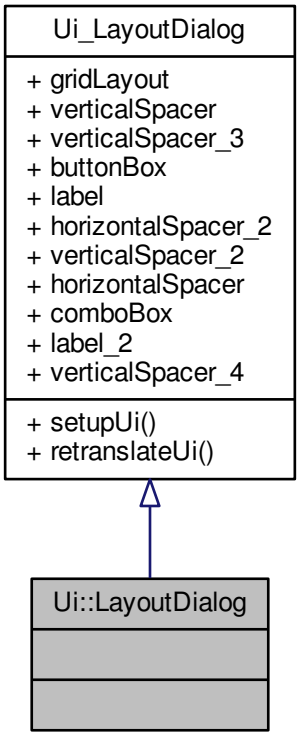
The documentation for this class was generated from the following files:

- src/ckb/kperfwidget.h
- src/ckb/kperfwidget.cpp

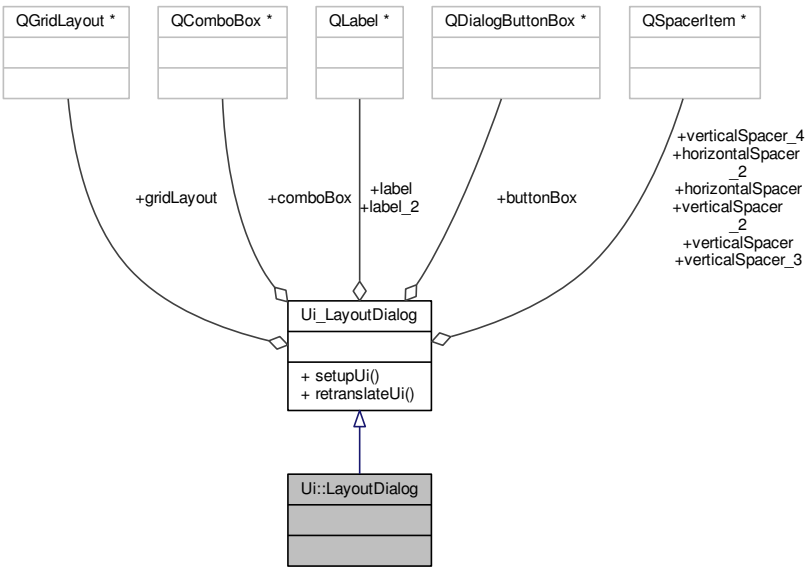
12.46 Ui::LayoutDialog Class Reference

`#include <src/ckb/ui_layoutdialog.h>`

Inheritance diagram for Ui::LayoutDialog:



Collaboration diagram for Ui::LayoutDialog:



Public Member Functions

- void setupUi (QDialog *LayoutDialog)
- void retranslateUi (QDialog *LayoutDialog)

Data Fields

- QGridLayout * gridLayout
- QSpacerItem * verticalSpacer
- QSpacerItem * verticalSpacer_3
- QDialogButtonBox * buttonBox
- QLabel * label
- QSpacerItem * horizontalSpacer_2
- QSpacerItem * verticalSpacer_2
- QSpacerItem * horizontalSpacer
- QComboBox * comboBox
- QLabel * label_2
- QSpacerItem * verticalSpacer_4

12.46.1 Detailed Description

Definition at line 119 of file ui_layoutdialog.h.

12.46.2 Member Function Documentation

12.46.2.1 void Ui_LayoutDialog::retranslateUi (QDialog * *LayoutDialog*) [inline], [inherited]

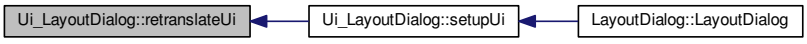
Definition at line 109 of file ui_layoutdialog.h.

References Ui_LayoutDialog::label, and Ui_LayoutDialog::label_2.

Referenced by Ui_LayoutDialog::setupUi().

```
110     {
111         LayoutDialog->setWindowTitle(QApplication::translate("LayoutDialog", "Keyboard Layout", Q_NULLPTR))
    ;
112         label->setText(QApplication::translate("LayoutDialog", "Please select your keyboard layout:",
    Q_NULLPTR));
113         label_2->setText(QApplication::translate("LayoutDialog", "Note: this only controls the
     layout within the ckb application. If your keys are typing the wrong characters, check your system preferences.",
     Q_NULLPTR));
114     } // retranslateUi
```

Here is the caller graph for this function:



12.46.2.2 void Ui_LayoutDialog::setupUi (QDialog * *LayoutDialog*) [inline], [inherited]

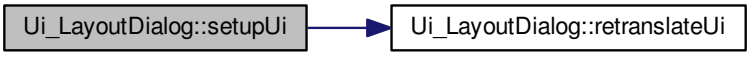
Definition at line 41 of file ui_layoutdialog.h.

References Ui_LayoutDialog::buttonBox, Ui_LayoutDialog::comboBox, Ui_LayoutDialog::gridLayout, Ui_LayoutDialog::horizontalSpacer, Ui_LayoutDialog::horizontalSpacer_2, Ui_LayoutDialog::label, Ui_LayoutDialog::label_2, Ui_LayoutDialog::retranslateUi(), Ui_LayoutDialog::verticalSpacer, Ui_LayoutDialog::verticalSpacer_2, Ui_LayoutDialog::verticalSpacer_3, and Ui_LayoutDialog::verticalSpacer_4.

Referenced by LayoutDialog::LayoutDialog().

```
42     {
43         if (LayoutDialog->objectName().isEmpty())
44             LayoutDialog->setObjectName(QStringLiteral("LayoutDialog"));
45         LayoutDialog->resize(400, 240);
46         gridLayout = new QGridLayout(LayoutDialog);
47         gridLayout->setObjectName(QStringLiteral("gridLayout"));
48         verticalSpacer = new QSpacerItem(20, 40, QSizePolicy::Minimum, QSizePolicy::Expanding
    );
49
50         gridLayout->addItem(verticalSpacer, 6, 0, 1, 3);
51
52         verticalSpacer_3 = new QSpacerItem(20, 40, QSizePolicy::Minimum,
    QSizePolicy::Expanding);
53
54         gridLayout->addItem(verticalSpacer_3, 0, 1, 1, 1);
55
56         buttonBox = new QDialogButtonBox(LayoutDialog);
57         buttonBox->setObjectName(QStringLiteral("buttonBox"));
58         buttonBox->setOrientation(Qt::Horizontal);
59         buttonBox->setStandardButtons(QDialogButtonBox::Ok);
60
61         gridLayout->addWidget(buttonBox, 7, 0, 1, 3);
62
63         label = new QLabel(LayoutDialog);
64         label->setObjectName(QStringLiteral("label"));
65         label->setAlignment(Qt::AlignCenter);
66
67         gridLayout->addWidget(label, 1, 0, 1, 3);
68
69         horizontalSpacer_2 = new QSpacerItem(40, 20, QSizePolicy::Expanding,
    QSizePolicy::Minimum);
70
71         gridLayout->addItem(horizontalSpacer_2, 3, 2, 1, 1);
72
73         verticalSpacer_2 = new QSpacerItem(20, 15, QSizePolicy::Minimum, QSizePolicy::Fixed
    );
74
75         gridLayout->addItem(verticalSpacer_2, 4, 1, 1, 1);
76
77         horizontalSpacer = new QSpacerItem(40, 20, QSizePolicy::Expanding,
    QSizePolicy::Minimum);
78
79         gridLayout->addItem(horizontalSpacer, 3, 0, 1, 1);
80
81         comboBox = new QComboBox(LayoutDialog);
82         comboBox->setObjectName(QStringLiteral("comboBox"));
83
84         gridLayout->addWidget(comboBox, 3, 1, 1, 1);
85
86         label_2 = new QLabel(LayoutDialog);
87         label_2->setObjectName(QStringLiteral("label_2"));
88         QFont font;
89         font.setPointSize(8);
90         font.setItalic(true);
91         label_2->setFont(font);
92         label_2->setAlignment(Qt::AlignCenter);
93         label_2->setWordWrap(true);
94
95         gridLayout->addWidget(label_2, 5, 0, 1, 3);
96
97         verticalSpacer_4 = new QSpacerItem(20, 5, QSizePolicy::Minimum, QSizePolicy::Fixed)
    ;
98
99         gridLayout->addItem(verticalSpacer_4, 2, 1, 1, 1);
100
101
102         retranslateUi(LayoutDialog);
103         QObject::connect(buttonBox, SIGNAL(accepted()), LayoutDialog, SLOT(accept()));
104         QObject::connect(buttonBox, SIGNAL(rejected()), LayoutDialog, SLOT(reject()));
105
106         QMetaObject::connectSlotsByName(LayoutDialog);
107     } // setupUi
```

Here is the call graph for this function:



Here is the caller graph for this function:

12.46.3 Field Documentation

12.46.3.1 QDialogButtonBox* Ui_LayoutDialog::buttonBox [inherited]

Definition at line 32 of file ui_layoutdialog.h.

Referenced by Ui_LayoutDialog::setupUi().

12.46.3.2 QComboBox* Ui_LayoutDialog::comboBox [inherited]

Definition at line 37 of file ui_layoutdialog.h.

Referenced by LayoutDialog::LayoutDialog(), LayoutDialog::selected(), and Ui_LayoutDialog::setupUi().

12.46.3.3 QGridLayout* Ui_LayoutDialog::gridLayout [inherited]

Definition at line 29 of file ui_layoutdialog.h.

Referenced by Ui_LayoutDialog::setupUi().

12.46.3.4 QSpacerItem* Ui_LayoutDialog::horizontalSpacer [inherited]

Definition at line 36 of file ui_layoutdialog.h.

Referenced by Ui_LayoutDialog::setupUi().

12.46.3.5 QSpacerItem* Ui_LayoutDialog::horizontalSpacer_2 [inherited]

Definition at line 34 of file ui_layoutdialog.h.

Referenced by Ui_LayoutDialog::setupUi().

12.46.3.6 QLabel* Ui_LayoutDialog::label [inherited]

Definition at line 33 of file ui_layoutdialog.h.

Referenced by Ui_LayoutDialog::retranslateUi(), and Ui_LayoutDialog::setupUi().

12.46.3.7 QLabel* Ui_LayoutDialog::label_2 [inherited]

Definition at line 38 of file ui_layoutdialog.h.

Referenced by Ui_LayoutDialog::retranslateUi(), and Ui_LayoutDialog::setupUi().

12.46.3.8 QSpacerItem* Ui_LayoutDialog::verticalSpacer [inherited]

Definition at line 30 of file ui_layoutdialog.h.

Referenced by Ui_LayoutDialog::setupUi().

12.46.3.9 QSpacerItem* Ui_LayoutDialog::verticalSpacer_2 [inherited]

Definition at line 35 of file ui_layoutdialog.h.

Referenced by Ui_LayoutDialog::setupUi().

12.46.3.10 QSpacerItem* Ui_LayoutDialog::verticalSpacer_3 [inherited]

Definition at line 31 of file ui_layoutdialog.h.

Referenced by Ui_LayoutDialog::setupUi().

12.46.3.11 QSpacerItem* Ui_LayoutDialog::verticalSpacer_4 [inherited]

Definition at line 39 of file ui_layoutdialog.h.

Referenced by Ui_LayoutDialog::setupUi().

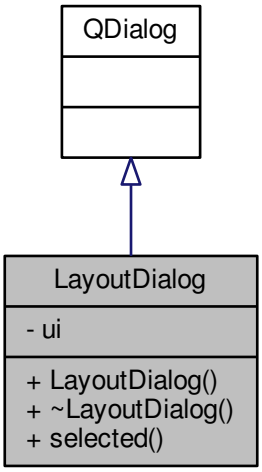
The documentation for this class was generated from the following file:

- src/ckb/ui_layoutdialog.h

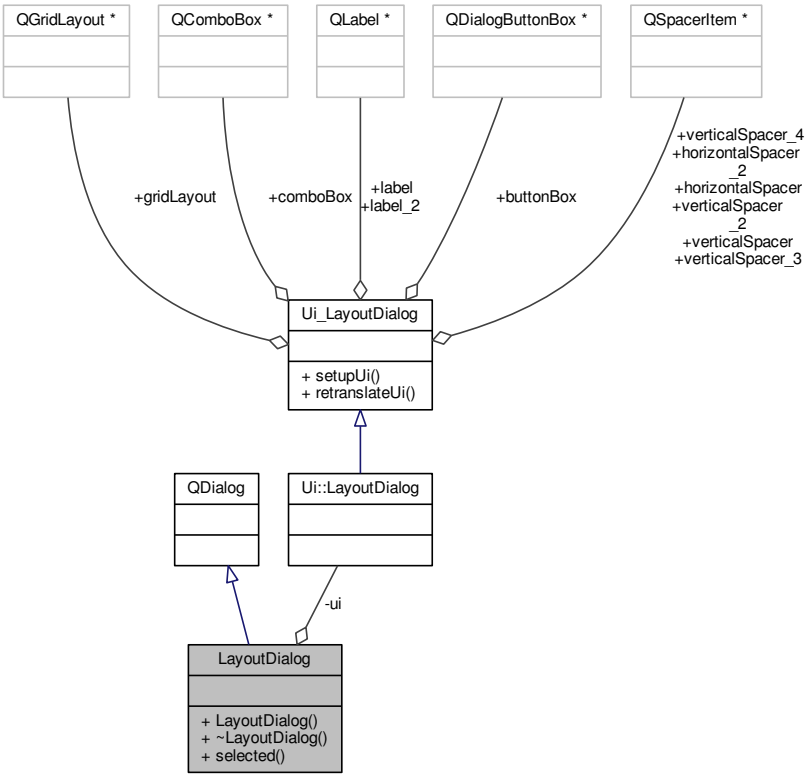
12.47 LayoutDialog Class Reference

`#include <src/ckb/layoutdialog.h>`

Inheritance diagram for LayoutDialog:



Collaboration diagram for LayoutDialog:



Public Member Functions

- LayoutDialog (QWidget *parent=0)
- ~LayoutDialog ()
- KeyMap::Layout selected () const

Private Attributes

- Ui::LayoutDialog * ui

12.47.1 Detailed Description

Definition at line 13 of file layoutdialog.h.

12.47.2 Constructor & Destructor Documentation

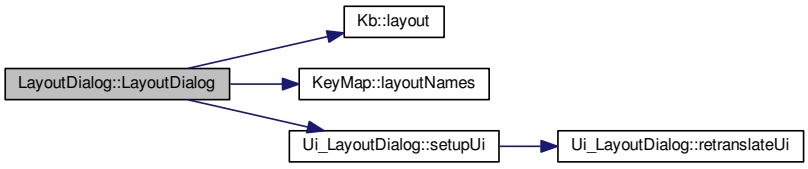
12.47.2.1 LayoutDialog::LayoutDialog (QWidget * *parent* = 0) [explicit]

Definition at line 5 of file layoutdialog.cpp.

References Ui_LayoutDialog::comboBox, Kb::layout(), KeyMap::layoutNames(), Ui_LayoutDialog::setupUi(), and ui.

```
5                                       :
6     QDialog(parent),
7     ui(new Ui::LayoutDialog) {
8     ui->setupUi(this);
9     // Populate layout list
10     ui->comboBox->addItems(KeyMap::layoutNames());
11     ui->comboBox->setCurrentIndex((int)Kb::layout());
12 }
```

Here is the call graph for this function:



12.47.2.2 LayoutDialog::~LayoutDialog ( )

Definition at line 14 of file layoutdialog.cpp.

References ui.

```
14                              {
15     delete ui;
16 }
```

12.47.3 Member Function Documentation

12.47.3.1 KeyMap::Layout LayoutDialog::selected ( ) const

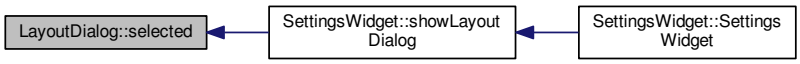
Definition at line 18 of file layoutdialog.cpp.

References Ui_LayoutDialog::comboBox, and ui.

Referenced by SettingsWidget::showLayoutDialog().

```
18                                          {
19     return (KeyMap::Layout)ui->comboBox->currentIndex();
20 }
```

Here is the caller graph for this function:



12.47.4 Field Documentation

12.47.4.1 Ui::LayoutDialog* LayoutDialog::ui [private]

Definition at line 24 of file layoutdialog.h.

Referenced by LayoutDialog(), selected(), and ~LayoutDialog().

The documentation for this class was generated from the following files:

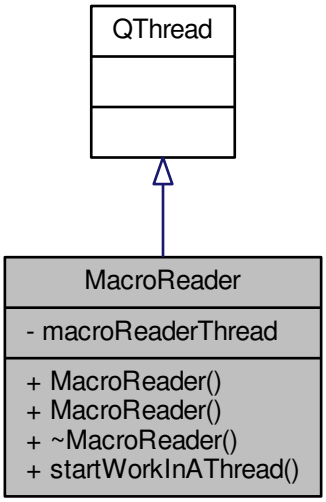
- src/ckb/layoutdialog.h
- src/ckb/layoutdialog.cpp

12.48 MacroReader Class Reference

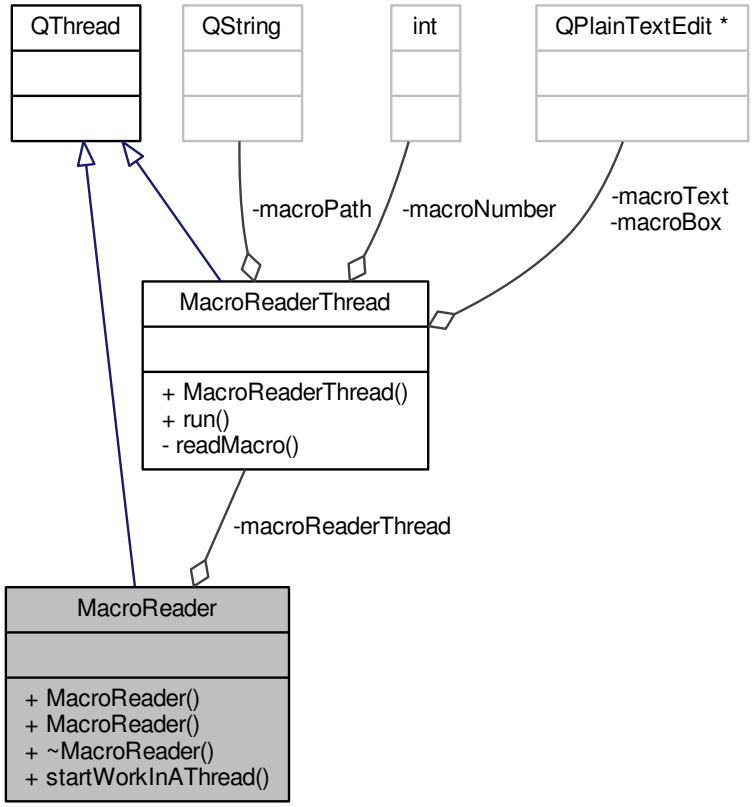
The MacroReader class creates a worker thread object. It does a connect do delayed deletion of thread local variables in the case the worker thread terminates.

`#include <src/ckb/macroreader.h>`

Inheritance diagram for MacroReader:



Collaboration diagram for MacroReader:



Public Member Functions

- MacroReader ()

  MacroReader Calling MacroReader without params is not allowed.
- MacroReader (int macroNumber, QString macroPath, QPlainTextEdit ∗macBox, QPlainTextEdit ∗macText)

  MacroReader This is the only allowed constructor. It only calls startWorkInAThread() with the four params.
- ∼MacroReader ()
- void startWorkInAThread (int macroNumber, QString macroPath, QPlainTextEdit ∗macBox, QPlainTextEdit ∗macText)

  startWorkInAThread This member function creates the new thread object and starts it with the four params.

Private Attributes

- MacroReaderThread ∗ macroReaderThread

12.48.1 Detailed Description

Definition at line 87 of file macroreader.h.

12.48.2 Constructor & Destructor Documentation

12.48.2.1 MacroReader::MacroReader ( )

Definition at line 9 of file macroreader.cpp.

```
9                           {
10     qDebug() << "Calling MacroReader without params is not allowed.";
11 }
```

12.48.2.2 MacroReader::MacroReader (int *macroNumber*, QString *macroPath*, QPlainTextEdit * *macBox*, QPlainTextEdit * *macText*)

Parameters

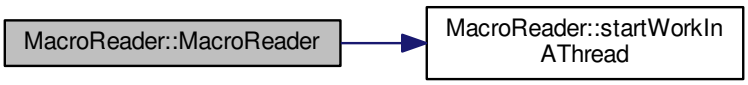
macroNumber	
macroPath	
macBox	
macText	

Definition at line 13 of file macroreader.cpp.

References startWorkInAThread().

```
13
       {
14     startWorkInAThread(macroNumber, macroPath, macBox, macText);
15 }
```

Here is the call graph for this function:



12.48.2.3 MacroReader::~MacroReader ( )

Definition at line 16 of file macroreader.cpp.

```
16 {}
```

12.48.3 Member Function Documentation

12.48.3.1 void MacroReader::startWorkInAThread (int *macroNumber*, QString *macroPath*, QPlainTextEdit * *macBox*, QPlainTextEdit * *macText*)

Parameters

macroNumber	
macroPath	
macBox	
macText	

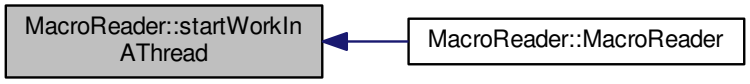
Definition at line 18 of file macroreader.cpp.

References macroReaderThread.

Referenced by MacroReader().

```
18
                {
19     macroReaderThread = new MacroReaderThread(macroNumber, macroPath,
      macBox, macText);
20     connect(macroReaderThread, &MacroReaderThread::finished,
      macroReaderThread, &QObject::deleteLater);
21     macroReaderThread->start();
22 }
```

Here is the caller graph for this function:



12.48.4 Field Documentation

12.48.4.1 MacroReaderThread* MacroReader::macroReaderThread [private]

Definition at line 123 of file macroreader.h.

Referenced by startWorkInAThread().

The documentation for this class was generated from the following files:

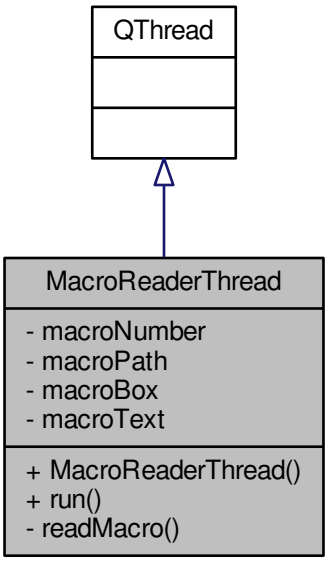
- src/ckb/macroreader.h
- src/ckb/macroreader.cpp

12.49 MacroReaderThread Class Reference

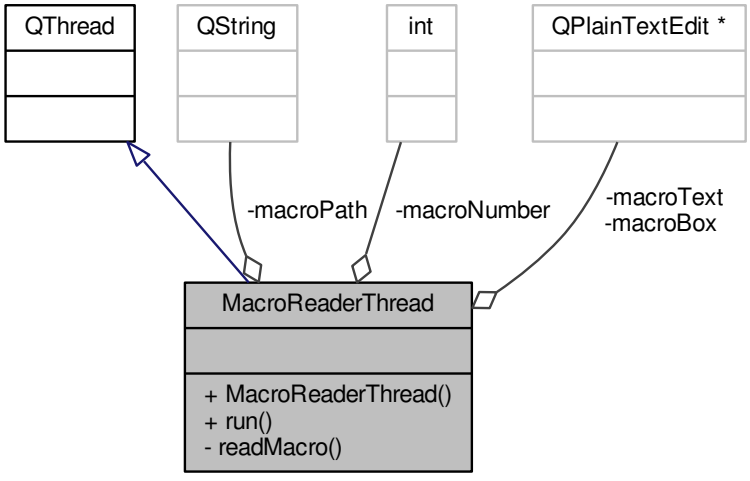
The MacroReaderThread class is responsible for reading Macro Key Values. It is created as a separate thread (worker thread) for reading macro commands from an fresh opened notify channel. Standard notify channel for macro definitions is number 2.

```
#include <src/ckb/macroreader.h>
```

Inheritance diagram for MacroReaderThread:



Collaboration diagram for MacroReaderThread:



Public Member Functions

- MacroReaderThread (int macNum, QString macPath, QPlainTextEdit ∗macBox, QPlainTextEdit ∗macText)

  MacroReaderThread saves the four params to local vars with similar varNames.

- void run () Q_DECL_OVERRIDE

  run is the notification reader main loop.

Private Slots

- void readMacro (QString line)

  readMacro is called for each line received by the worker thread. The method ist called via signal (metaObject) from the worker thread, which reads the keyboard input. Just display the key code in the macroBox Widget without he trailing newline and reposition the cursor in the macro pane.

Private Attributes

- int macroNumber

  macroNumber Filenames of nofity channels have the structure <input-device-path>/ckb1/notify<number> First part is hold in macroPath, the number is hold in macroNumber. macroNumber may range from 0 to 9.
- QString macroPath

  macroPath holds the path for the notify channel
- QPlainTextEdit * macroBox

  macroBox will receive the Macro Key Values sent from the keyboard while defining a new macro.
- QPlainTextEdit * macroText

  macroText is the other textpane used in the UI while typing new macros. That variable is used for setting the focus to that textpane and to set the cursor at EOT.

12.49.1 Detailed Description

While the worker Thread gets input from the keyboard, the lines are sent via signalling (metaobject) to run a member function in the context of the Qt UI manager.

When the notify channel is closed (that's normally done by pressing "Stop"-Button in the UI), the worker thread closes the channelFile and leaves.

See Also

MacroReaderThread(), ~MacroReaderThread(), readMaco(), run()

Definition at line 22 of file macroreader.h.

12.49.2 Constructor & Destructor Documentation

12.49.2.1 MacroReaderThread::MacroReaderThread (int *macNum*, QString *macPath*, QPlainTextEdit * *macBox*, QPlainTextEdit * *macText*) [inline]

Parameters

macNum	
macPath	
macBox	
macText	

Definition at line 55 of file macroreader.h.

References macroBox, macroNumber, macroPath, and macroText.

```
55                                                                                  {
56         macroNumber = macNum;
57         macroPath = macPath;
58         macroBox = macBox;
59         macroText = macText;
60     }
```

12.49.3 Member Function Documentation

12.49.3.1 void MacroReaderThread::readMacro (QString *line*) [private], [slot]

This is used, because the worker thread shouldn't get access to the UI elements (and normally has none, because the pointers macroBox and macroText remain on the stack). That mechanism guarantees, that the UI does not freeze if it happens something magic to the reading function.

Parameters

line	holds the line just got from keyboard

We want to see the keys as they appear in the macroText Widget.

Because it is possible to change the Focus via keyboard, we must set the focus on each call.

Definition at line 27 of file macroreader.cpp.

References macroBox, and macroText.

```
27                                              {
32     macroText->setFocus();
33     QTextCursor c = macroText->textCursor();
34     c.setPosition(macroText->toPlainText().length());
35     macroText->setTextCursor(c);
36     macroBox->appendPlainText(line.left(line.size()-1));
37 }
```

12.49.3.2 void MacroReaderThread::run ( )

MacroReaderThread::run is the standard main function for a thread. Tries to open a file <macroPath><macroNumber> several times (in this case, it should be possible the first time (the code was recycled from kb.cpp).

While the file is open, read lines an signal them via metaObject() to the main thread. When the file is closed by the sender, close it as reader and terminate.

Definition at line 47 of file macroreader.cpp.

References macroBox, macroNumber, macroPath, and macroText.

```
47                              {
48     qDebug() << "MacroReader::run() started with" << macroNumber << "and" <<
       macroPath << "and" << macroBox << "and" << macroText;
49
50     QFile macroFile(macroPath);
51     // Wait a small amount of time for the node to open (100ms) (code reused from kb.cpp)
52     QThread::usleep(100000);
53     if(!macroFile.open(QIODevice::ReadOnly)){
54         // If it's still not open, try again before giving up (1s at a time, 10s total)
55         QThread::usleep(900000);
56         for(int i = 1; i < 10; i++){
57             if(macroFile.open(QIODevice::ReadOnly))
58                 break;
59             QThread::sleep(1);
60         }
61         if(!macroFile.isOpen()) {
62             qDebug() << QString("unable to open macroFile (%1)").arg(macroPath);
63             return;
64         }
65     }
66     // Read data from notification node macroPath
67     // Count time between lines read from the interface
68     QByteArray line;
69     timeval t;
70     gettimeofday(&t, NULL);
71     double tstart = t.tv_sec+(t.tv_usec/1000000.0);
72     bool firstline = true;
73
74     while(macroFile.isOpen() && (line = macroFile.readLine()).length() > 0){
75         QString text = QString::fromUtf8(line);
76         gettimeofday(&t, NULL);
77         double tnow = t.tv_sec+(t.tv_usec/1000000.0);
78
79         // in the first line, there is only a delay "before start". Don't use it.
80         if (!firstline) {
```

```
81              text.prepend ("\n");
82              text.prepend (QString::number ((tnow - tstart) * 1000000.0, 'f', 0));
83              text.prepend ("=");
84          } else firstline = false;
85          tstart = tnow;
86
87          metaObject()->invokeMethod(this, "readMacro", Qt::QueuedConnection, Q_ARG(QString, text));
88      }
89      qDebug() << "MacroReader::run() ends.";
90      macroFile.close();
91      QThread::exit ();
92  }
```

12.49.4 Field Documentation

12.49.4.1 QPlainTextEdit* MacroReaderThread::macroBox [private]

Definition at line 40 of file macroreader.h.

Referenced by MacroReaderThread(), readMacro(), and run().

12.49.4.2 int MacroReaderThread::macroNumber [private]

Definition at line 31 of file macroreader.h.

Referenced by MacroReaderThread(), and run().

12.49.4.3 QString MacroReaderThread::macroPath [private]

See Also

macroNumber

Definition at line 36 of file macroreader.h.

Referenced by MacroReaderThread(), and run().

12.49.4.4 QPlainTextEdit* MacroReaderThread::macroText [private]

Definition at line 45 of file macroreader.h.

Referenced by MacroReaderThread(), readMacro(), and run().

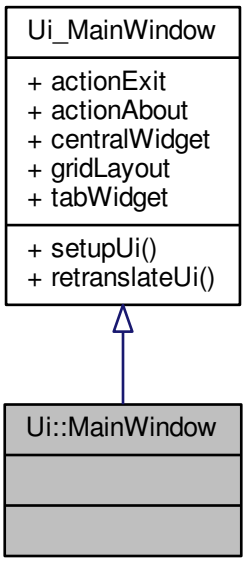
The documentation for this class was generated from the following files:

- src/ckb/macroreader.h
- src/ckb/macroreader.cpp

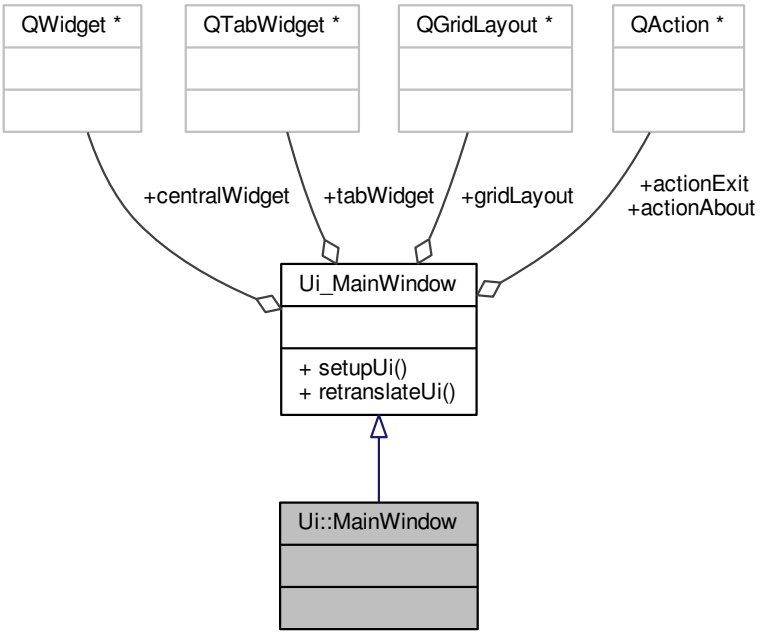
12.50 Ui::MainWindow Class Reference

`#include <src/ckb/ui_mainwindow.h>`

Inheritance diagram for Ui::MainWindow:



Collaboration diagram for Ui::MainWindow:



Public Member Functions

- void setupUi (QMainWindow *MainWindow)
- void retranslateUi (QMainWindow *MainWindow)

Data Fields

- QAction * actionExit
- QAction * actionAbout
- QWidget * centralWidget
- QGridLayout * gridLayout
- QTabWidget * tabWidget

12.50.1 Detailed Description

Definition at line 79 of file ui_mainwindow.h.

12.50.2 Member Function Documentation

12.50.2.1 void Ui_MainWindow::retranslateUi (QMainWindow * *MainWindow*) [inline], [inherited]

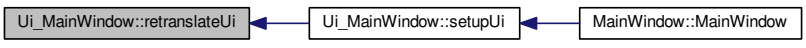
Definition at line 68 of file ui_mainwindow.h.

References Ui_MainWindow::actionAbout, and Ui_MainWindow::actionExit.

Referenced by Ui_MainWindow::setupUi().

```
69     {
70         MainWindow->setWindowTitle(QApplication::translate("MainWindow", "ckb-next", Q_NULLPTR));
71         actionExit->setText(QApplication::translate("MainWindow", "Exit", Q_NULLPTR));
72         actionExit->setShortcut(QApplication::translate("MainWindow", "Ctrl+Q", Q_NULLPTR));
73         actionAbout->setText(QApplication::translate("MainWindow", "About", Q_NULLPTR));
74     } // retranslateUi
```

Here is the caller graph for this function:



12.50.2.2 void Ui_MainWindow::setupUi (QMainWindow * *MainWindow*) [inline], [inherited]

Definition at line 33 of file ui_mainwindow.h.

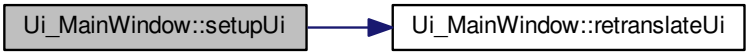
References Ui_MainWindow::actionAbout, Ui_MainWindow::actionExit, Ui_MainWindow::centralWidget, Ui_MainWindow::gridLayout, Ui_MainWindow::retranslateUi(), and Ui_MainWindow::tabWidget.

Referenced by MainWindow::MainWindow().

```
34     {
35         if (MainWindow->objectName().isEmpty())
36             MainWindow->setObjectName(QStringLiteral("MainWindow"));
37         MainWindow->resize(1000, 750);
38         MainWindow->setMinimumSize(QSize(1000, 750));
39         QIcon icon;
40         icon.addFile(QStringLiteral(":/img/ckb-logo.png"), QSize(), QIcon::Normal, QIcon::Off);
```

```
41          MainWindow->setWindowIcon(icon);
42          MainWindow->setUnifiedTitleAndToolBarOnMac(true);
43          actionExit = new QAction(MainWindow);
44          actionExit->setObjectName(QStringLiteral("actionExit"));
45          actionAbout = new QAction(MainWindow);
46          actionAbout->setObjectName(QStringLiteral("actionAbout"));
47          centralWidget = new QWidget(MainWindow);
48          centralWidget->setObjectName(QStringLiteral("centralWidget"));
49          gridLayout = new QGridLayout(centralWidget);
50          gridLayout->setSpacing(6);
51          gridLayout->setContentsMargins(11, 11, 11, 11);
52          gridLayout->setObjectName(QStringLiteral("gridLayout"));
53          tabWidget = new QTabWidget(centralWidget);
54          tabWidget->setObjectName(QStringLiteral("tabWidget"));
55
56          gridLayout->addWidget(tabWidget, 0, 0, 1, 1);
57
58          MainWindow->setCentralWidget(centralWidget);
59
60          retranslateUi(MainWindow);
61
62          tabWidget->setCurrentIndex(-1);
63
64
65          QMetaObject::connectSlotsByName(MainWindow);
66      } // setupUi
```

Here is the call graph for this function:



Here is the caller graph for this function:

12.50.3 Field Documentation

12.50.3.1 QAction* Ui_MainWindow::actionAbout [inherited]

Definition at line 28 of file ui_mainwindow.h.

Referenced by Ui_MainWindow::retranslateUi(), and Ui_MainWindow::setupUi().

12.50.3.2 QAction* Ui_MainWindow::actionExit [inherited]

Definition at line 27 of file ui_mainwindow.h.

Referenced by MainWindow::MainWindow(), Ui_MainWindow::retranslateUi(), and Ui_MainWindow::setupUi().

12.50.3.3 QWidget* Ui_MainWindow::centralWidget [inherited]

Definition at line 29 of file ui_mainwindow.h.

Referenced by Ui_MainWindow::setupUi().

12.50.3.4 QGridLayout* Ui_MainWindow::gridLayout [inherited]

Definition at line 30 of file ui_mainwindow.h.

Referenced by Ui_MainWindow::setupUi().

12.50.3.5 QTabWidget* Ui_MainWindow::tabWidget [inherited]

Definition at line 31 of file ui_mainwindow.h.

Referenced by MainWindow::addDevice(), MainWindow::MainWindow(), MainWindow::removeDevice(), Ui_MainWindow::setupUi(), and MainWindow::showFwUpdateNotification().

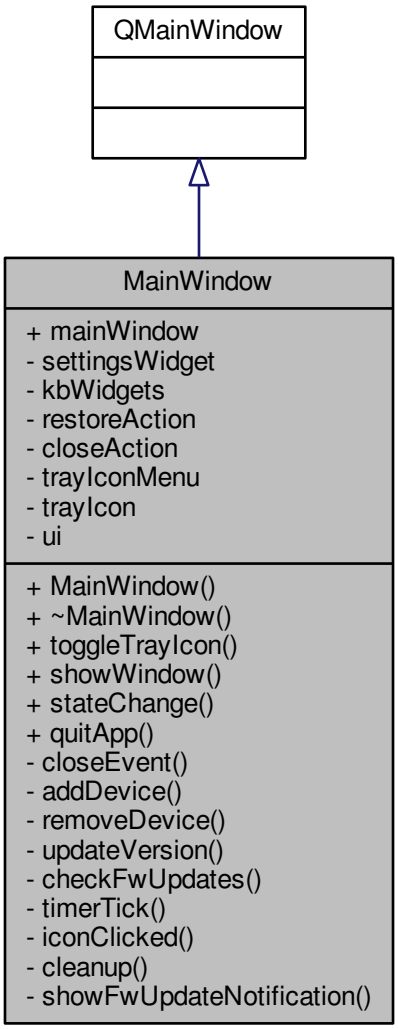
The documentation for this class was generated from the following file:

- src/ckb/ui_mainwindow.h

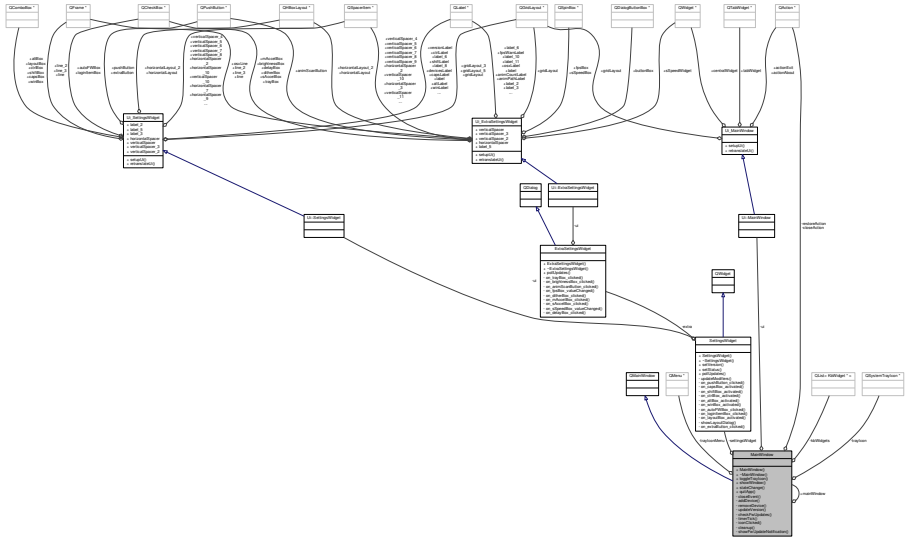
12.51 MainWindow Class Reference

`#include <src/ckb/mainwindow.h>`

Inheritance diagram for MainWindow:



Collaboration diagram for MainWindow:



Public Slots

- void showWindow ()
- void stateChange (Qt::ApplicationState state)
- void quitApp ()

Public Member Functions

- MainWindow (QWidget ∗parent=0)
- ∼MainWindow ()
- void toggleTrayIcon (bool visible)

Static Public Attributes

- static MainWindow ∗ mainWindow = 0

Private Slots

- void addDevice (Kb ∗device)
- void removeDevice (Kb ∗device)
- void updateVersion ()
- void checkFwUpdates ()
- void timerTick ()
- void iconClicked (QSystemTrayIcon::ActivationReason reason)
- void cleanup ()
- void showFwUpdateNotification (QWidget ∗widget, float version)

Private Member Functions

- void closeEvent (QCloseEvent ∗event)

Private Attributes

- SettingsWidget * settingsWidget
- QList< KbWidget * > kbWidgets
- QAction * restoreAction
- QAction * closeAction
- QMenu * trayIconMenu
- QSystemTrayIcon * trayIcon
- Ui::MainWindow * ui

12.51.1 Detailed Description

Definition at line 27 of file mainwindow.h.

12.51.2 Constructor & Destructor Documentation

12.51.2.1 MainWindow::MainWindow (QWidget * *parent* = 0) [explicit]

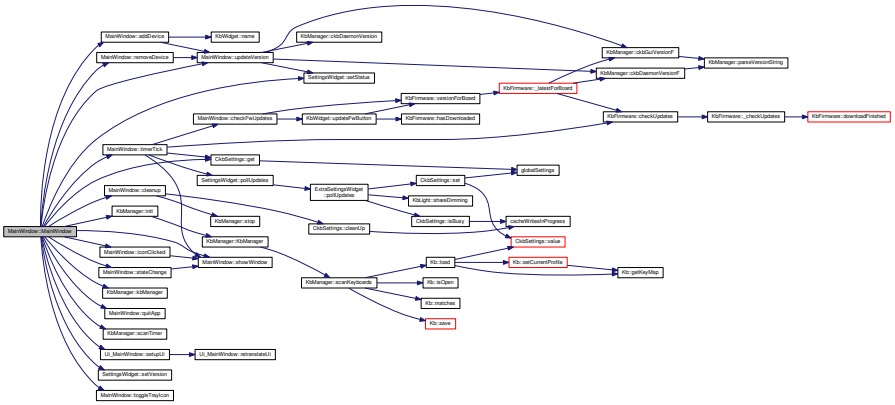
Definition at line 49 of file mainwindow.cpp.

References Ui_MainWindow::actionExit, addDevice(), cleanup(), closeAction, configLabel, daemonDialogText, devpath, CkbSettings::get(), iconClicked(), KbManager::init(), KbManager::kbManager(), mainWindow, quitApp(), removeDevice(), restoreAction, KbManager::scanTimer(), SettingsWidget::setStatus(), settingsWidget, Ui_MainWindow::setupUi(), SettingsWidget::setVersion(), showWindow(), stateChange(), Ui_MainWindow::tabWidget, timerTick(), toggleTrayIcon(), trayIcon, trayIconMenu, ui, and updateVersion().

```
49                                      :
50     QMainWindow(parent),
51     ui(new Ui::MainWindow)
52 {
53     ui->setupUi(this);
54     mainWindow = this;
55
56     // Start device manager
57     KbManager::init(CKB_VERSION_STR);
58     connect(KbManager::kbManager(), SIGNAL(kbConnected(Kb*)), this, SLOT(
      addDevice(Kb*)));
59     connect(KbManager::kbManager(), SIGNAL(kbDisconnected(Kb*)), this, SLOT(
      removeDevice(Kb*)));
60     connect(KbManager::kbManager(), SIGNAL(versionUpdated()), this, SLOT(
      updateVersion()));
61     connect(KbManager::scanTimer(), SIGNAL(timeout()), this, SLOT(
      timerTick()));
62
63     // Set up tray icon
64     restoreAction = new QAction(tr("Restore"), this);
65     closeAction = new QAction(tr("Quit ckb"), this);
66 #ifdef USE_LIBAPPINDICATOR
67     QString desktop = std::getenv("XDG_CURRENT_DESKTOP");
68     unityDesktop = (desktop.toLower() == "unity");
69
70     if(unityDesktop){
71         trayIcon = 0;
72
73         indicatorMenu = gtk_menu_new();
74         indicatorMenuRestoreItem = gtk_menu_item_new_with_label("Restore");
75         indicatorMenuQuitItem = gtk_menu_item_new_with_label("Quit ckb");
76
77         gtk_menu_shell_append(GTK_MENU_SHELL(indicatorMenu), indicatorMenuRestoreItem);
78         gtk_menu_shell_append(GTK_MENU_SHELL(indicatorMenu), indicatorMenuQuitItem);
79
80         g_signal_connect(indicatorMenuQuitItem, "activate",
81             G_CALLBACK(quitIndicator), this);
82         g_signal_connect(indicatorMenuRestoreItem, "activate",
83             G_CALLBACK(restoreIndicator), this);
84
85         gtk_widget_show(indicatorMenuRestoreItem);
86         gtk_widget_show(indicatorMenuQuitItem);
87
88         indicator = app_indicator_new("ckb", "indicator-messages",
      APP_INDICATOR_CATEGORY_APPLICATION_STATUS);
89
```

```
90          app_indicator_set_status(indicator, APP_INDICATOR_STATUS_ACTIVE);
91          app_indicator_set_menu(indicator, GTK_MENU(indicatorMenu));
92          app_indicator_set_icon(indicator, "ckb");
93      } else
94 #endif // USE_LIBAPPINDICATOR
95      {
96          trayIconMenu = new QMenu(this);
97          trayIconMenu->addAction(restoreAction);
98          trayIconMenu->addAction(closeAction);
99          trayIcon = new QSystemTrayIcon(QIcon(":/img/ckb-logo.png"), this);
100          trayIcon->setContextMenu(trayIconMenu);
101          connect(trayIcon, SIGNAL(activated(QSystemTrayIcon::ActivationReason)), this, SLOT(
    iconClicked(QSystemTrayIcon::ActivationReason)));
102      }
103      toggleTrayIcon(!CkbSettings::get("Program/SuppressTrayIcon").toBool());
104
105 #ifdef Q_OS_MACX
106      // Make a custom "Close" menu action for OSX, as the default one brings up the "still running" popup
     unnecessarily
107      QMenuBar* menuBar = new QMenuBar(this);
108      setMenuBar(menuBar);
109      this->menuBar()->addMenu("ckb")->addAction(closeAction);
110 #else
111      // On linux, add a handler for Ctrl+Q
112      new QShortcut(QKeySequence("Ctrl+Q"), this, SLOT(quitApp()));
113 #endif
114
115      connect(ui->actionExit, SIGNAL(triggered()), this, SLOT(quitApp()));
116      connect(closeAction, SIGNAL(triggered()), this, SLOT(quitApp()));
117      connect(restoreAction, SIGNAL(triggered()), this, SLOT(
    showWindow()));
118      connect(qApp, SIGNAL(applicationStateChanged(Qt::ApplicationState)), this, SLOT(
    stateChange(Qt::ApplicationState)));
119
120      connect(qApp, SIGNAL(aboutToQuit()), this, SLOT(cleanup()));
121
122      ui->tabWidget->addTab(settingsWidget = new
    SettingsWidget(this), configLabel);
123      settingsWidget->setVersion("ckb-next " CKB_VERSION_STR);
124
125      // create daemon dialog as a QMessageBox
126      // this will create a focussed dialog, that has to be interacted with,
127      // if the daemon is not running
128      // set the main and informative text to tell the user about the issue
129      QMessageBox dialog;
130      dialog.setText(tr("The ckb-next daemon is not running. This program will <b>not</b> work without it!"))
    ;
131      dialog.setInformativeText(daemonDialogText);
132      dialog.setIcon(QMessageBox::Critical);
133
134      // check, whether daemon is running
135      // the daemon creates the root device path on initialization and thus it
136      // can be assumed, that the daemon is not running if doesn't exist
137      // `.arg(0)` is necessary to interpolate the correct suffix into the path
138      // see `./kbmanager.cpp` for details
139      QFileInfo rootDevPath(devpath.arg(0));
140      if (!rootDevPath.exists()) {
141          // set settings widget's status
142          // show the main window (otherwise only the dialog will be visible)
143          // finally show the dialog
144          settingsWidget->setStatus(tr("The ckb-next daemon is not running."));
145          showWindow();
146          dialog.exec();
147      }
148 }
```

Here is the call graph for this function:



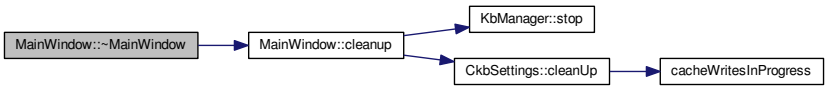
12.51.2.2 MainWindow::~MainWindow ( )

Definition at line 341 of file mainwindow.cpp.

References cleanup(), and ui.

```
341                         {
342     cleanup();
343     delete ui;
344 }
```

Here is the call graph for this function:



12.51.3 Member Function Documentation

12.51.3.1 void MainWindow::addDevice (Kb * *device*) [private], [slot]

Definition at line 159 of file mainwindow.cpp.

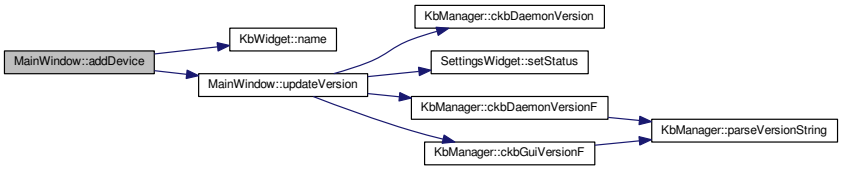
References KbWidget::device, kbWidgets, KbWidget::name(), Ui_MainWindow::tabWidget, ui, and updateVersion().

Referenced by MainWindow().

```
159                                 {
160     // Connected already?
161     foreach(KbWidget* w, kbWidgets){
162         if(w->device == device)
163             return;
164     }
165     // Add the keyboard
166     KbWidget* widget = new KbWidget(this, device);
167     kbWidgets.append(widget);
168     // Add to tabber; switch to this device if the user is on the settings screen
169     int count = ui->tabWidget->count();
170     ui->tabWidget->insertTab(count - 1, widget, widget->name());
171     if(ui->tabWidget->currentIndex() == count)
172         ui->tabWidget->setCurrentIndex(count - 1);
```

```
173     // Update connected device count
174     updateVersion();
175 }
```

Here is the call graph for this function:



Here is the caller graph for this function:

12.51.3.2 void MainWindow::checkFwUpdates ( ) [private], [slot]

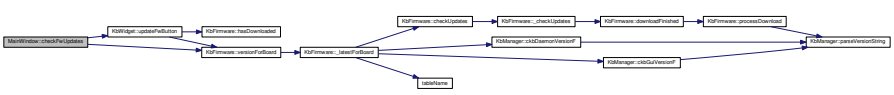
Definition at line 210 of file mainwindow.cpp.

References KbWidget::device, Kb::features, Kb::firmware, KbWidget::hasShownNewFW, kbWidgets, mainWindow, KbWidget::updateFwButton(), and KbFirmware::versionForBoard().

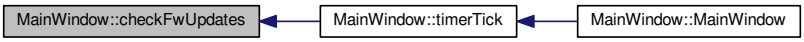
Referenced by timerTick().

```
210                                 {
211     if(!mainWindow->isVisible())
212         return;
213     foreach(KbWidget* w, kbWidgets){
214         // Display firmware upgrade notification if a new version is available
215         float version = KbFirmware::versionForBoard(w->
    device->features);
216         if(version > w->device->firmware.toFloat()){
217             if(w->hasShownNewFW)
218                 continue;
219             w->hasShownNewFW = true;
220             w->updateFwButton();
221             // Don't run this method here because it will lock up the timer and prevent devices from
     working properly
222             // Use a queued invocation instead
223             metaObject()->invokeMethod(this, "showFwUpdateNotification", Qt::QueuedConnection, Q_ARG(
    QWidget*, w), Q_ARG(float, version));
224             // Don't display more than one of these at once
225             return;
226         }
227     }
228 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.51.3.3 void MainWindow::cleanup ( ) [private], [slot]

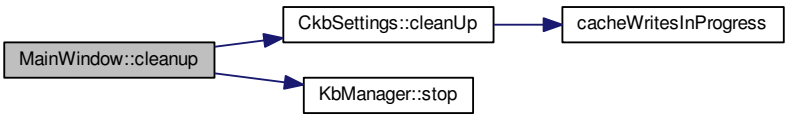
Definition at line 333 of file mainwindow.cpp.

References CkbSettings::cleanUp(), kbWidgets, and KbManager::stop().

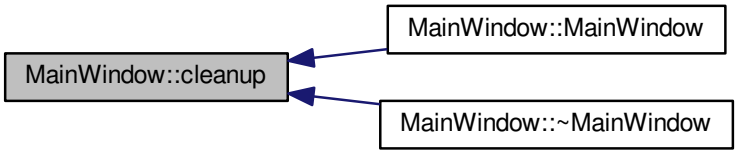
Referenced by MainWindow(), and ~MainWindow().

```
333                         {
334     foreach(KbWidget* w, kbWidgets)
335         delete w;
336     kbWidgets.clear();
337     KbManager::stop();
338     CkbSettings::cleanUp();
339 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



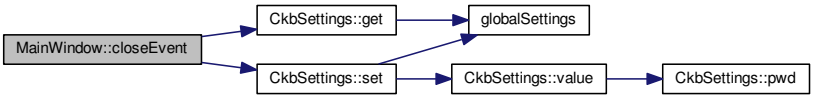
12.51.3.4 void MainWindow::closeEvent (QCloseEvent * *event*) [private]

Definition at line 247 of file mainwindow.cpp.

References CkbSettings::get(), and CkbSettings::set().

```
247                                             {
248     // If the window is hidden already or the event is non-spontaneous (can happen on OSX when using the
      Quit menu), accept it and close
249     if(!event->spontaneous() || isHidden()){
250         event->accept();
251         return;
252     }
253     if(!CkbSettings::get("Popups/BGWarning").toBool()){
254         QMessageBox::information(this, "ckb", "ckb will still run in the background.\nTo close it, choose
      Exit from the tray menu\nor click \"Quit ckb\" on the Settings screen.");
255         CkbSettings::set("Popups/BGWarning", true);
256     }
257     hide();
258     event->ignore();
259 }
```

Here is the call graph for this function:



12.51.3.5 void MainWindow::iconClicked (QSystemTrayIcon::ActivationReason *reason*) [private],[slot]

Definition at line 295 of file mainwindow.cpp.

References showWindow().

Referenced by MainWindow().

```
295                                                                 {
296     // On Linux, hide/show the app when the tray icon is clicked
297     // On OSX this just shows the menu
298 #ifndef Q_OS_MACX
299     if(reason == QSystemTrayIcon::DoubleClick || reason == QSystemTrayIcon::Trigger){
300         if(isVisible())
301             hide();
302         else
303             showWindow();
304     }
305 #endif
306 }
```

Here is the call graph for this function:

Here is the caller graph for this function:

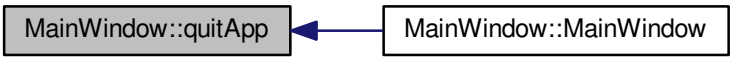
12.51.3.6 void MainWindow::quitApp ( ) [slot]

Definition at line 329 of file mainwindow.cpp.

Referenced by MainWindow().

```
329                          {
330     qApp->quit();
331 }
```

Here is the caller graph for this function:



12.51.3.7 void MainWindow::removeDevice (Kb * *device*) [private], [slot]

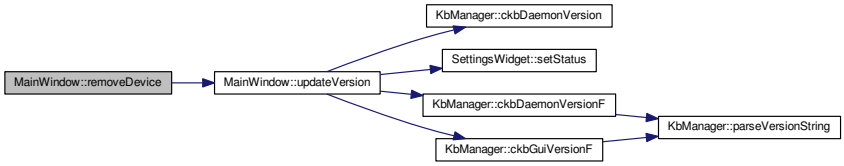
Definition at line 177 of file mainwindow.cpp.

References KbWidget::device, kbWidgets, Ui_MainWindow::tabWidget, ui, and updateVersion().

Referenced by MainWindow().

```
177                                     {
178     foreach(KbWidget* w, kbWidgets){
179         // Remove this device from the UI
180         if(w->device == device){
181             int i = kbWidgets.indexOf(w);
182             ui->tabWidget->removeTab(i);
183             kbWidgets.removeAt(i);
184             w->deleteLater();
185         }
186     }
187     // Update connected device count
188     updateVersion();
189 }
```

Here is the call graph for this function:



Here is the caller graph for this function:

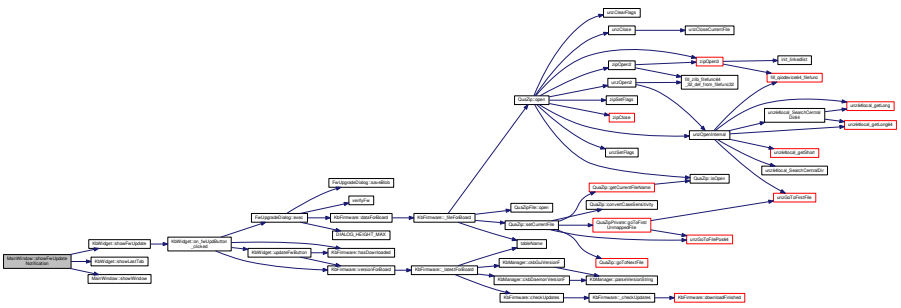
12.51.3.8 void MainWindow::showFwUpdateNotification (QWidget * *widget*, float *version*) [private], [slot]

Definition at line 230 of file mainwindow.cpp.

References KbWidget::device, kbWidgets, KbWidget::showFwUpdate(), KbWidget::showLastTab(), showWindow(), Ui_MainWindow::tabWidget, ui, and Kb::usbModel.

```
230                                                                                 {
231     static bool isShowing = false;
232     if(isShowing)
233         return;
234     isShowing = true;
235     showWindow();
236     KbWidget* w = (KbWidget*)widget;
237     // Ask for update
238     if(QMessageBox::information(this, "Firmware update", tr("A new firmware is available for your %1 (v%2)
      \nWould you like to install it now?").arg(w->device->usbModel, QString::number(version, 'f', 2)
      ), QMessageBox::StandardButtons(QMessageBox::Yes | QMessageBox::No), QMessageBox::Yes) == QMessageBox::Yes){
239         // If accepted, switch to the firmware tab and bring up the update window
240         w->showLastTab();
241         ui->tabWidget->setCurrentIndex(kbWidgets.indexOf(w));
242         w->showFwUpdate();
243     }
244     isShowing = false;
245 }
```

Here is the call graph for this function:



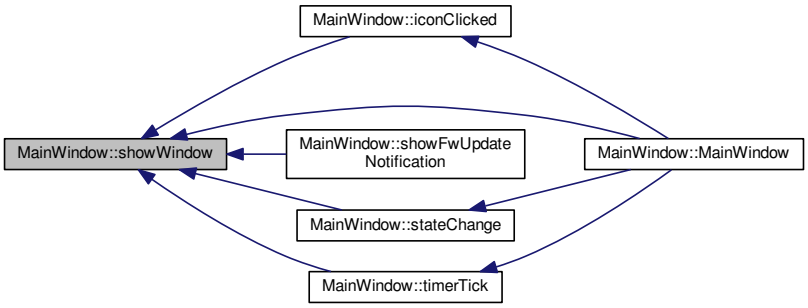
12.51.3.9 void MainWindow::showWindow ( ) [slot]

Definition at line 308 of file mainwindow.cpp.

Referenced by iconClicked(), MainWindow(), showFwUpdateNotification(), stateChange(), and timerTick().

```
308                             {
309     showNormal();
310     raise();
311     activateWindow();
312 }
```

Here is the caller graph for this function:



12.51.3.10 void MainWindow::stateChange (Qt::ApplicationState *state*) [slot]

Definition at line 314 of file mainwindow.cpp.

References showWindow().

Referenced by MainWindow().

```
314                                                 {
315     // On OSX it's possible for the app to be brought to the foreground without the window actually
      reappearing.
316     // We want to make sure it's shown when this happens.
317 #ifdef Q_OS_MAC
318     static quint64 lastStateChange = 0;
319     quint64 now = QDateTime::currentMSecsSinceEpoch();
320     if(state == Qt::ApplicationActive){
321         // This happens once at startup so ignore it. Also don't allow it to be called more than once every
      2s.
322         if(lastStateChange != 0 && now >= lastStateChange + 2 * 1000)
323             showWindow();
324         lastStateChange = now;
325     }
326 #endif
327 }
```

Here is the call graph for this function:

Here is the caller graph for this function:

12.51.3.11 void MainWindow::timerTick ( ) [private], [slot]

Definition at line 261 of file mainwindow.cpp.

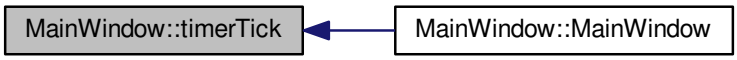
References appShare, checkFwUpdates(), KbFirmware::checkUpdates(), CkbSettings::get(), SettingsWidget::pollUpdates(), settingsWidget, and showWindow().

Referenced by MainWindow().

```
261                         {
262     // Check shared memory for changes
263     if(appShare.lock()){
264         void* data = appShare.data();
265         QStringList commands = QString((const char*)data).split("\n");
266         // Restore PID, remove all other data
267         snprintf((char*)appShare.data(), appShare.size(), "PID %ld", (long)getpid());
268         appShare.unlock();
269         // Parse commands
270         foreach(const QString& line, commands){
271             // Old ckb option line - bring application to foreground
272             if(line == "Open")
273                 showWindow();
274             if(line.startsWith("Option ")){
275                 // New ckb option line
276                 QString option = line.split(" ")[1];
277                 if(option == "Open")
278                     // Bring to foreground
279                     showWindow();
280                 else if(option == "Close")
281                     // Quit application
282                     qApp->quit();
283             }
284         }
285     }
286     // Check for firmware updates (when appropriate)
287     if(!CkbSettings::get("Program/DisableAutoFWCheck").toBool()){
288         KbFirmware::checkUpdates();
289         checkFwUpdates();
290     }
291     // Poll for setting updates
292     settingsWidget->pollUpdates();
293 }
```

Here is the call graph for this function:

Here is the caller graph for this function:



12.51.3.12 void MainWindow::toggleTrayIcon (bool *visible*)

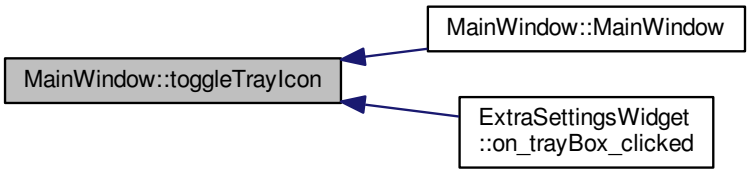
Definition at line 150 of file mainwindow.cpp.

References trayIcon.

Referenced by MainWindow(), and ExtraSettingsWidget::on_trayBox_clicked().

```
150                                          {
151 #ifdef USE_LIBAPPINDICATOR
152     if(unityDesktop)
153         app_indicator_set_status(indicator, visible ? APP_INDICATOR_STATUS_ACTIVE :
    APP_INDICATOR_STATUS_PASSIVE);
154     else
155 #endif // USE_LIBAPPINDICATOR
156         trayIcon->setVisible(visible);
157 }
```

Here is the caller graph for this function:



12.51.3.13 void MainWindow::updateVersion ( ) [private], [slot]

Definition at line 191 of file mainwindow.cpp.

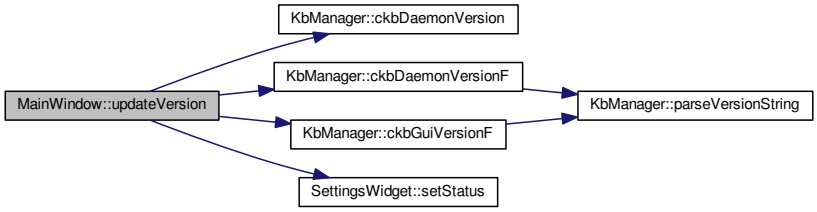
References KbManager::ckbDaemonVersion(), KbManager::ckbDaemonVersionF(), KbManager::ckbGuiVersionF(), DAEMON_UNAVAILABLE_STR, kbWidgets, SettingsWidget::setStatus(), and settingsWidget.

Referenced by addDevice(), MainWindow(), and removeDevice().

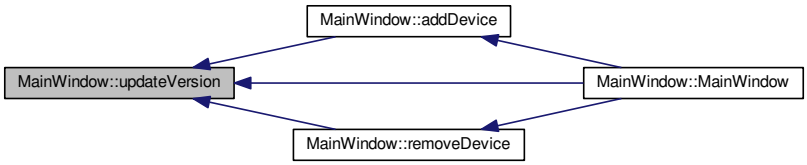
```
191                             {
192     QString daemonVersion = KbManager::ckbDaemonVersion();
193     if(daemonVersion == DAEMON_UNAVAILABLE_STR){
194         settingsWidget->setStatus("Driver inactive");
195         return;
196     }
197     int count = kbWidgets.count();
198     // Warn if the daemon version doesn't match the GUI
```

```
199     QString daemonWarning;
200     if(daemonVersion != CKB_VERSION_STR)
201         daemonWarning = "<br /><br /><b>Warning:</b> Driver version mismatch (" + daemonVersion + ").
       Please upgrade ckb" + QString(KbManager::ckbDaemonVersionF() >
      KbManager::ckbGuiVersionF() ? "" : "-daemon") + ". If the problem persists, try
       rebooting.";
202     if(count == 0)
203         settingsWidget->setStatus("No devices connected" + daemonWarning);
204     else if(count == 1)
205         settingsWidget->setStatus("1 device connected" + daemonWarning);
206     else
207         settingsWidget->setStatus(QString("%1 devices connected").arg(count) +
      daemonWarning);
208 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.51.4 Field Documentation

12.51.4.1 QAction* MainWindow::closeAction [private]

Definition at line 44 of file mainwindow.h.

Referenced by MainWindow().

12.51.4.2 QList<KbWidget*> MainWindow::kbWidgets [private]

Definition at line 41 of file mainwindow.h.

Referenced by addDevice(), checkFwUpdates(), cleanup(), removeDevice(), showFwUpdateNotification(), and updateVersion().

12.51.4.3 MainWindow * MainWindow::mainWindow = 0 [static]

Definition at line 35 of file mainwindow.h.

Referenced by checkFwUpdates(), MainWindow(), and ExtraSettingsWidget::on_trayBox_clicked().

12.51.4.4 QAction* MainWindow::restoreAction [private]

Definition at line 43 of file mainwindow.h.

Referenced by MainWindow().

12.51.4.5 SettingsWidget* MainWindow::settingsWidget [private]

Definition at line 40 of file mainwindow.h.

Referenced by MainWindow(), timerTick(), and updateVersion().

12.51.4.6 QSystemTrayIcon* MainWindow::trayIcon [private]

Definition at line 54 of file mainwindow.h.

Referenced by MainWindow(), and toggleTrayIcon().

12.51.4.7 QMenu* MainWindow::trayIconMenu [private]

Definition at line 53 of file mainwindow.h.

Referenced by MainWindow().

12.51.4.8 Ui::MainWindow* MainWindow::ui [private]

Definition at line 75 of file mainwindow.h.

Referenced by addDevice(), MainWindow(), removeDevice(), showFwUpdateNotification(), and ~MainWindow().

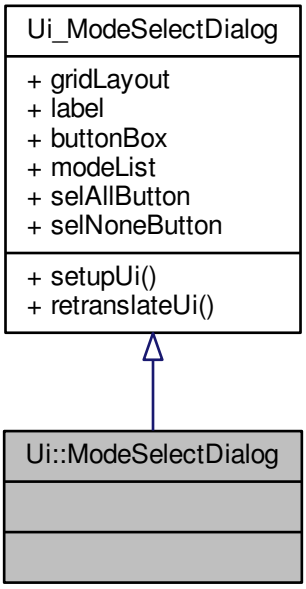
The documentation for this class was generated from the following files:

- src/ckb/mainwindow.h
- src/ckb/mainwindow.cpp

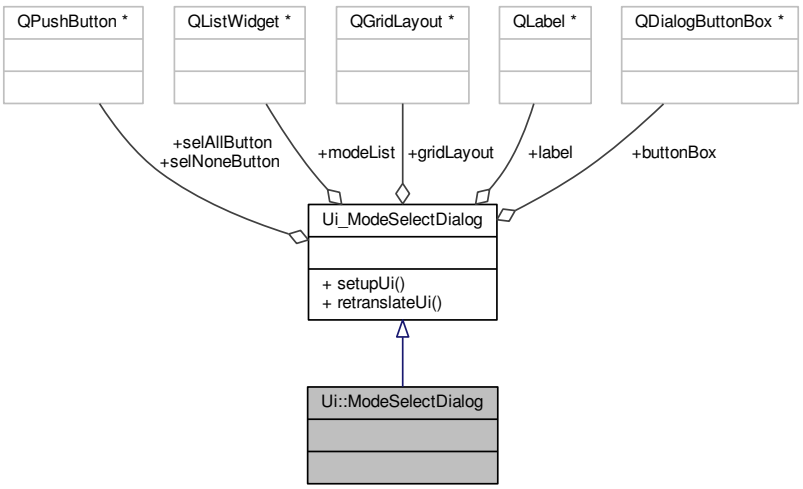
12.52 Ui::ModeSelectDialog Class Reference

`#include <src/ckb/ui_modeselectdialog.h>`

Inheritance diagram for Ui::ModeSelectDialog:



Collaboration diagram for Ui::ModeSelectDialog:



Public Member Functions

- void setupUi (QDialog *ModeSelectDialog)
- void retranslateUi (QDialog *ModeSelectDialog)

Data Fields

- QGridLayout * gridLayout
- QLabel * label
- QDialogButtonBox * buttonBox
- QListWidget * modeList
- QPushButton * selAllButton
- QPushButton * selNoneButton

12.52.1 Detailed Description

Definition at line 95 of file ui_modeselectdialog.h.

12.52.2 Member Function Documentation

12.52.2.1 void Ui_ModeSelectDialog::retranslateUi (QDialog * *ModeSelectDialog*) [inline], [inherited]

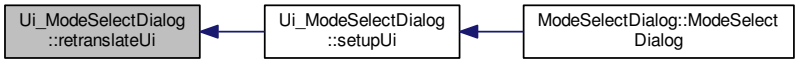
Definition at line 84 of file ui_modeselectdialog.h.

References Ui_ModeSelectDialog::label, Ui_ModeSelectDialog::selAllButton, and Ui_ModeSelectDialog::selNoneButton.

Referenced by Ui_ModeSelectDialog::setupUi().

```
85     {
86         ModeSelectDialog->setWindowTitle(QApplication::translate("ModeSelectDialog", "Select Modes",
      Q_NULLPTR));
87         label->setText(QApplication::translate("ModeSelectDialog", "TextLabel", Q_NULLPTR));
88         selAllButton->setText(QApplication::translate("ModeSelectDialog", "Select All",
      Q_NULLPTR));
89         selNoneButton->setText(QApplication::translate("ModeSelectDialog", "Select None",
      Q_NULLPTR));
90     } // retranslateUi
```

Here is the caller graph for this function:



12.52.2.2 void Ui_ModeSelectDialog::setupUi (QDialog * *ModeSelectDialog*) [inline], [inherited]

Definition at line 36 of file ui_modeselectdialog.h.

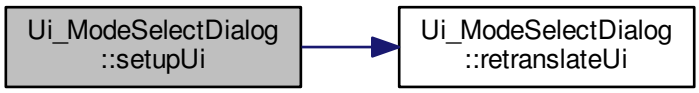
References Ui_ModeSelectDialog::buttonBox, Ui_ModeSelectDialog::gridLayout, Ui_ModeSelectDialog::label, Ui_ModeSelectDialog::modeList, Ui_ModeSelectDialog::retranslateUi(), Ui_ModeSelectDialog::selAllButton, and Ui_ModeSelectDialog::selNoneButton.

Referenced by ModeSelectDialog::ModeSelectDialog().

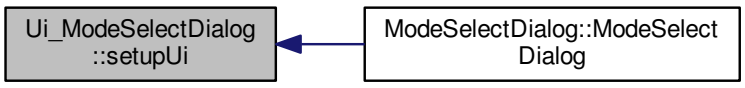
```
37     {
38         if (ModeSelectDialog->objectName().isEmpty())
39             ModeSelectDialog->setObjectName(QStringLiteral("ModeSelectDialog"));
40         ModeSelectDialog->resize(320, 320);
41         gridLayout = new QGridLayout(ModeSelectDialog);
42         gridLayout->setObjectName(QStringLiteral("gridLayout"));
43         label = new QLabel(ModeSelectDialog);
```

```
44          label->setObjectName(QStringLiteral("label"));
45          label->setAlignment(Qt::AlignLeading|Qt::AlignLeft|Qt::AlignTop);
46
47          gridLayout->addWidget(label, 0, 0, 1, 2);
48
49          buttonBox = new QDialogButtonBox(ModeSelectDialog);
50          buttonBox->setObjectName(QStringLiteral("buttonBox"));
51          buttonBox->setOrientation(Qt::Vertical);
52          buttonBox->setStandardButtons(QDialogButtonBox::Cancel|QDialogButtonBox::Ok);
53          buttonBox->setCenterButtons(false);
54
55          gridLayout->addWidget(buttonBox, 0, 2, 3, 1);
56
57          modeList = new QListWidget(ModeSelectDialog);
58          modeList->setObjectName(QStringLiteral("modeList"));
59          modeList->setFrameShadow(QFrame::Plain);
60          modeList->setSelectionMode(QAbstractItemView::ExtendedSelection);
61
62          gridLayout->addWidget(modeList, 1, 0, 1, 2);
63
64          selAllButton = new QPushButton(ModeSelectDialog);
65          selAllButton->setObjectName(QStringLiteral("selAllButton"));
66
67          gridLayout->addWidget(selAllButton, 2, 0, 1, 1);
68
69          selNoneButton = new QPushButton(ModeSelectDialog);
70          selNoneButton->setObjectName(QStringLiteral("selNoneButton"));
71
72          gridLayout->addWidget(selNoneButton, 2, 1, 1, 1);
73
74          QWidget::setTabOrder(modeList, selAllButton);
75          QWidget::setTabOrder(selAllButton, selNoneButton);
76
77          retranslateUi(ModeSelectDialog);
78          QObject::connect(buttonBox, SIGNAL(accepted()), ModeSelectDialog, SLOT(accept()));
79          QObject::connect(buttonBox, SIGNAL(rejected()), ModeSelectDialog, SLOT(reject()));
80
81          QMetaObject::connectSlotsByName(ModeSelectDialog);
82      } // setupUi
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.52.3 Field Documentation

12.52.3.1 QDialogButtonBox* Ui_ModeSelectDialog::buttonBox [inherited]

Definition at line 31 of file ui_modeselectdialog.h.

Referenced by Ui_ModeSelectDialog::setupUi().

12.52.3.2 QGridLayout* Ui_ModeSelectDialog::gridLayout [inherited]

Definition at line 29 of file ui_modeselectdialog.h.

Referenced by Ui_ModeSelectDialog::setupUi().

12.52.3.3 QLabel* Ui_ModeSelectDialog::label [inherited]

Definition at line 30 of file ui_modeselectdialog.h.

Referenced by ModeSelectDialog::ModeSelectDialog(), Ui_ModeSelectDialog::retranslateUi(), and Ui_ModeSelectDialog::setupUi().

12.52.3.4 QListWidget* Ui_ModeSelectDialog::modeList [inherited]

Definition at line 32 of file ui_modeselectdialog.h.

Referenced by ModeSelectDialog::ModeSelectDialog(), ModeSelectDialog::on_selAllButton_clicked(), ModeSelectDialog::on_selNoneButton_clicked(), ModeSelectDialog::selection(), and Ui_ModeSelectDialog::setupUi().

12.52.3.5 QPushButton* Ui_ModeSelectDialog::selAllButton [inherited]

Definition at line 33 of file ui_modeselectdialog.h.

Referenced by Ui_ModeSelectDialog::retranslateUi(), and Ui_ModeSelectDialog::setupUi().

12.52.3.6 QPushButton* Ui_ModeSelectDialog::selNoneButton [inherited]

Definition at line 34 of file ui_modeselectdialog.h.

Referenced by Ui_ModeSelectDialog::retranslateUi(), and Ui_ModeSelectDialog::setupUi().

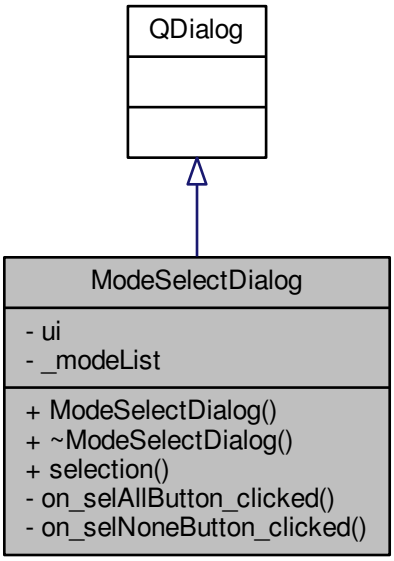
The documentation for this class was generated from the following file:

- src/ckb/ui_modeselectdialog.h

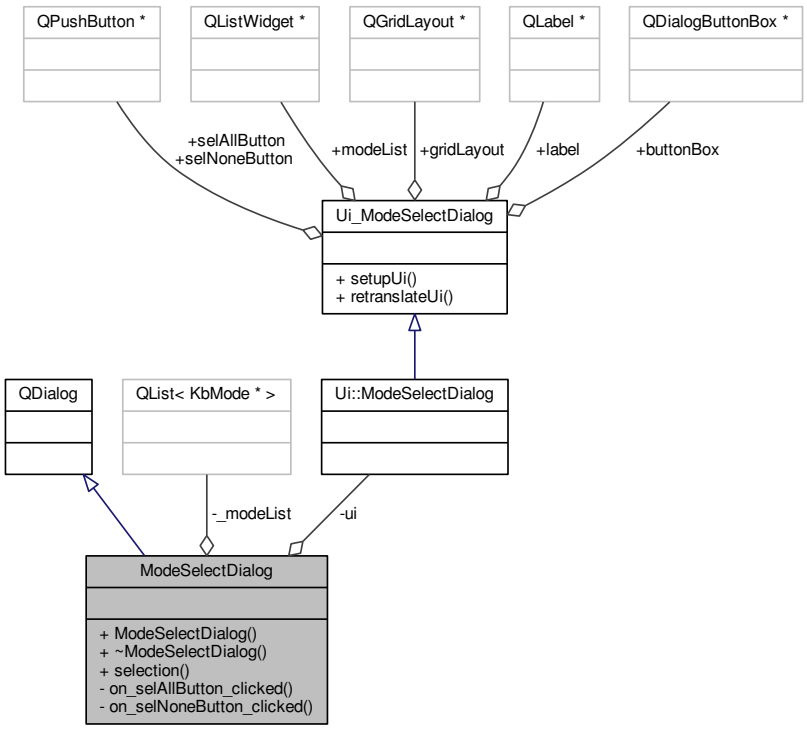
12.53 ModeSelectDialog Class Reference

`#include <src/ckb/modeselectdialog.h>`

Inheritance diagram for ModeSelectDialog:



Collaboration diagram for ModeSelectDialog:



Public Member Functions

- ModeSelectDialog (QWidget *parent, KbMode *currentMode, QList< KbMode * > modeList, const QString &textLabel)
- ~ModeSelectDialog ()
- QList< KbMode * > selection ()

Private Slots

- void on_selAllButton_clicked ()
- void on_selNoneButton_clicked ()

Private Attributes

- Ui::ModeSelectDialog * ui
- QList< KbMode * > _modeList

12.53.1 Detailed Description

Definition at line 13 of file modeselectdialog.h.

12.53.2 Constructor & Destructor Documentation

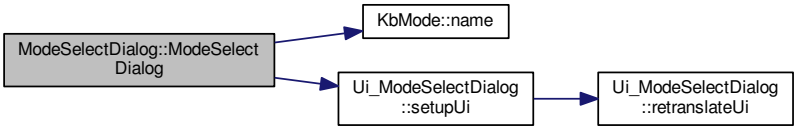
12.53.2.1 ModeSelectDialog::ModeSelectDialog (QWidget * *parent*, KbMode * *currentMode*, QList< KbMode * > *modeList*, const QString & *textLabel*)

Definition at line 4 of file modeselectdialog.cpp.

References _modeList, Ui_ModeSelectDialog::label, Ui_ModeSelectDialog::modeList, KbMode::name(), Ui_ModeSelectDialog::setupUi(), and ui.

```
4
                    :
5     QDialog(parent), ui(new Ui::ModeSelectDialog),
6     _modeList(modeList)
7 {
8     ui->setupUi(this);
9     _modeList.removeAll(currentMode);
10     ui->label->setText(textLabel);
11     // Populate list
12     foreach(KbMode* mode, _modeList)
13         ui->modeList->addItem(mode->name());
14 }
```

Here is the call graph for this function:



12.53.2.2 ModeSelectDialog::~ModeSelectDialog ( )

Definition at line 16 of file modeselectdialog.cpp.

References ui.

```
16                                      {
17     delete ui;
18 }
```

12.53.3 Member Function Documentation

12.53.3.1 void ModeSelectDialog::on_selAllButton_clicked ( ) [private], [slot]

Definition at line 30 of file modeselectdialog.cpp.

References Ui_ModeSelectDialog::modeList, and ui.

```
30                                                {
31     int count = ui->modeList->count();
32     for(int i = 0; i < count; i++)
33         ui->modeList->item(i)->setSelected(true);
34 }
```

12.53.3.2 void ModeSelectDialog::on_selNoneButton_clicked ( ) [private], [slot]

Definition at line 36 of file modeselectdialog.cpp.

References Ui_ModeSelectDialog::modeList, and ui.

```
36                                                 {
37     int count = ui->modeList->count();
38     for(int i = 0; i < count; i++)
39         ui->modeList->item(i)->setSelected(false);
40 }
```

12.53.3.3 QList< KbMode * > ModeSelectDialog::selection ( )

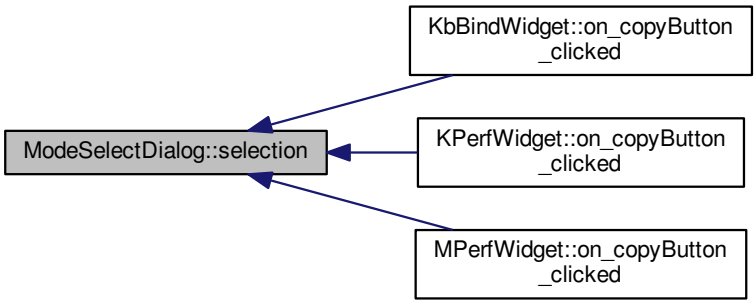
Definition at line 20 of file modeselectdialog.cpp.

References _modeList, Ui_ModeSelectDialog::modeList, and ui.

Referenced by KbBindWidget::on_copyButton_clicked(), KPerfWidget::on_copyButton_clicked(), and MPerfWidget::on_copyButton_clicked().

```
20                                          {
21     QList<KbMode*> res;
22     int count = ui->modeList->count();
23     for(int i = 0; i < count; i++){
24         if(ui->modeList->item(i)->isSelected())
25             res.append(_modeList[i]);
26     }
27     return res;
28 }
```

Here is the caller graph for this function:



12.53.4 Field Documentation

12.53.4.1 QList<KbMode*> ModeSelectDialog::_modeList [private]

Definition at line 31 of file modeselectdialog.h.

Referenced by ModeSelectDialog(), and selection().

12.53.4.2 Ui::ModeSelectDialog* ModeSelectDialog::ui [private]

Definition at line 30 of file modeselectdialog.h.

Referenced by ModeSelectDialog(), on_selAllButton_clicked(), on_selNoneButton_clicked(), selection(), and ~ModeSelectDialog().

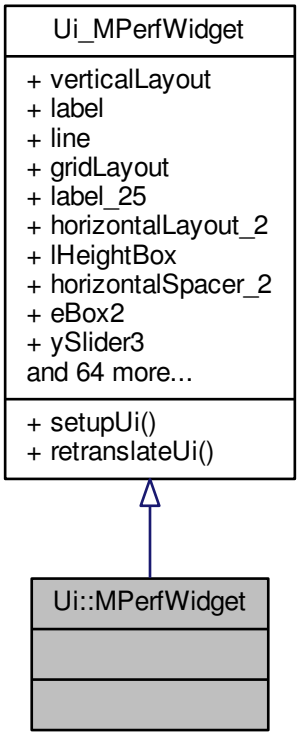
The documentation for this class was generated from the following files:

- src/ckb/modeselectdialog.h
- src/ckb/modeselectdialog.cpp

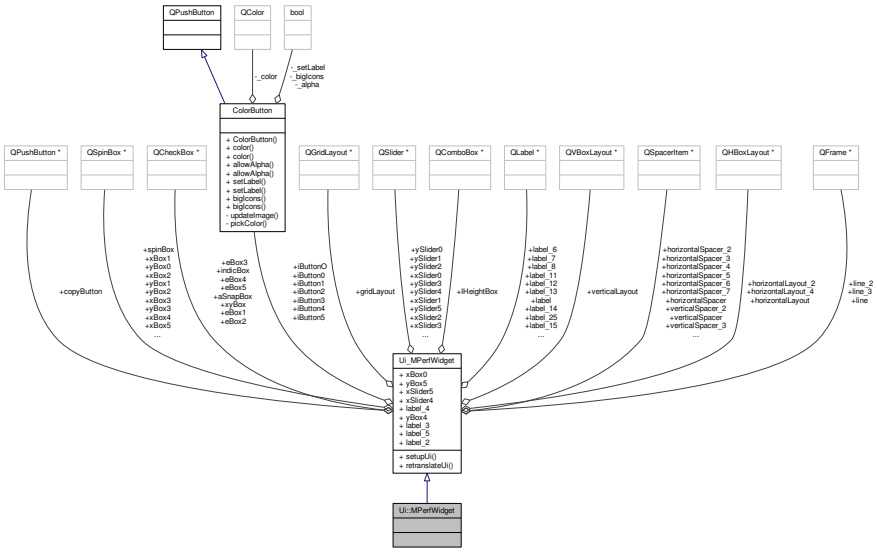
12.54 Ui::MPerfWidget Class Reference

#include <src/ckb/ui_mperfwidget.h>

Inheritance diagram for Ui::MPerfWidget:



Collaboration diagram for Ui::MPerfWidget:



Public Member Functions

- void setupUi (QWidget *MPerfWidget)
- void retranslateUi (QWidget *MPerfWidget)

Data Fields

- QVBoxLayout * verticalLayout
- QLabel * label
- QFrame * line
- QGridLayout * gridLayout
- QLabel * label_25
- QHBoxLayout * horizontalLayout_2
- QComboBox * lHeightBox
- QSpacerItem * horizontalSpacer_2
- QCheckBox * eBox2
- QSlider * ySlider3
- QSlider * xSlider2
- QSpinBox * xBox3
- QSpinBox * yBox0
- QSlider * ySlider0
- ColorButton * iButton4
- ColorButton * iButton5
- QSpacerItem * horizontalSpacer_4
- QSpacerItem * verticalSpacer_3
- QSpacerItem * horizontalSpacer_5
- ColorButton * iButton3
- QLabel * label_12
- QSpinBox * xBox4
- QSpinBox * xBox0
- QSpinBox * yBox5
- QCheckBox * eBox5
- QSlider * xSlider5
- QSpinBox * xBox1
- QSpinBox * yBox2
- QSpinBox * yBox3
- QSlider * xSlider4
- QLabel * label_4
- QSlider * xSlider0
- ColorButton * iButton2
- QFrame * line_3
- ColorButton * iButtonO
- QSpinBox * yBox1
- QSlider * ySlider2
- QSpinBox * xBox2
- QSlider * xSlider1
- QSpacerItem * horizontalSpacer
- QSpinBox * yBox4
- ColorButton * iButton1
- QSlider * ySlider4
- QSlider * ySlider1
- ColorButton * iButton0
- QSlider * xSlider3
- QSpinBox * xBox5

- QLabel * label_3
- QLabel * label_6
- QSlider * ySlider5
- QLabel * label_7
- QLabel * label_8
- QLabel * label_5
- QLabel * label_15
- QLabel * label_13
- QLabel * label_14
- QCheckBox * eBox4
- QLabel * label_11
- QCheckBox * eBox3
- QCheckBox * eBox1
- QLabel * label_2
- QSpacerItem * verticalSpacer
- QCheckBox * aSnapBox
- QFrame * line_2
- QHBoxLayout * horizontalLayout_4
- QCheckBox * indicBox
- QSpacerItem * horizontalSpacer_3
- QSpinBox * spinBox
- QSpacerItem * horizontalSpacer_7
- QCheckBox * xyBox
- QSpacerItem * verticalSpacer_2
- QHBoxLayout * horizontalLayout
- QPushButton * copyButton
- QSpacerItem * horizontalSpacer_6

12.54.1 Detailed Description

Definition at line 768 of file ui_mperfwidget.h.

12.54.2 Member Function Documentation

12.54.2.1 void Ui_MPerfWidget::retranslateUi (QWidget * *MPerfWidget*) [inline], [inherited]

Definition at line 682 of file ui_mperfwidget.h.

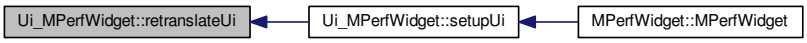
References Ui_MPerfWidget::aSnapBox, Ui_MPerfWidget::copyButton, Ui_MPerfWidget::eBox1, Ui_MPerfWidget::eBox2, Ui_MPerfWidget::eBox3, Ui_MPerfWidget::eBox4, Ui_MPerfWidget::eBox5, Ui_MPerfWidget::iButton0, Ui_MPerfWidget::iButton1, Ui_MPerfWidget::iButton2, Ui_MPerfWidget::iButton3, Ui_MPerfWidget::iButton4, Ui_MPerfWidget::iButton5, Ui_MPerfWidget::iButtonO, Ui_MPerfWidget::indicBox, Ui_MPerfWidget::label, Ui_MPerfWidget::label_11, Ui_MPerfWidget::label_12, Ui_MPerfWidget::label_13, Ui_MPerfWidget::label_14, Ui_MPerfWidget::label_15, Ui_MPerfWidget::label_2, Ui_MPerfWidget::label_25, Ui_MPerfWidget::label_3, Ui_MPerfWidget::label_4, Ui_MPerfWidget::label_5, Ui_MPerfWidget::label_6, Ui_MPerfWidget::label_7, Ui_MPerfWidget::label_8, Ui_MPerfWidget::lHeightBox, Ui_MPerfWidget::spinBox, and Ui_MPerfWidget::xyBox.

Referenced by Ui_MPerfWidget::setupUi().

```
683     {
684         MPerfWidget->setWindowTitle(QApplication::translate("MPerfWidget", "Form", Q_NULLPTR));
685         label->setText(QApplication::translate("MPerfWidget", "DPI", Q_NULLPTR));
686         label_25->setText(QApplication::translate("MPerfWidget", "Lift height:", Q_NULLPTR));
687         lHeightBox->clear();
688         lHeightBox->insertItems(0, QStringList()
689          << QApplication::translate("MPerfWidget", "Low", Q_NULLPTR)
690          << QApplication::translate("MPerfWidget", "Medium-low", Q_NULLPTR)
691          << QApplication::translate("MPerfWidget", "Medium", Q_NULLPTR)
692          << QApplication::translate("MPerfWidget", "Medium-high", Q_NULLPTR)
```

```
693          << QApplication::translate("MPerfWidget", "High", Q_NULLPTR)
694         );
695 #ifndef QT_NO_TOOLTIP
696         eBox2->setToolTip(QApplication::translate("MPerfWidget", "Uncheck this box to skip the stage
    while moving up or down.", Q_NULLPTR));
697 #endif // QT_NO_TOOLTIP
698         eBox2->setText(QString());
699 #ifndef QT_NO_TOOLTIP
700         iButton4->setToolTip(QApplication::translate("MPerfWidget", "Tip: alt+click to set all
    colors at once", Q_NULLPTR));
701 #endif // QT_NO_TOOLTIP
702         iButton4->setText(QString());
703 #ifndef QT_NO_TOOLTIP
704         iButton5->setToolTip(QApplication::translate("MPerfWidget", "Tip: alt+click to set all
    colors at once", Q_NULLPTR));
705 #endif // QT_NO_TOOLTIP
706         iButton5->setText(QString());
707 #ifndef QT_NO_TOOLTIP
708         iButton3->setToolTip(QApplication::translate("MPerfWidget", "Tip: alt+click to set all
    colors at once", Q_NULLPTR));
709 #endif // QT_NO_TOOLTIP
710         iButton3->setText(QString());
711         label_12->setText(QApplication::translate("MPerfWidget", "X Value:", Q_NULLPTR));
712 #ifndef QT_NO_TOOLTIP
713         eBox5->setToolTip(QApplication::translate("MPerfWidget", "Uncheck this box to skip the stage
    while moving up or down.", Q_NULLPTR));
714 #endif // QT_NO_TOOLTIP
715         eBox5->setText(QString());
716         label_4->setText(QApplication::translate("MPerfWidget", "Y Value:", Q_NULLPTR));
717 #ifndef QT_NO_TOOLTIP
718         iButton2->setToolTip(QApplication::translate("MPerfWidget", "Tip: alt+click to set all
    colors at once", Q_NULLPTR));
719 #endif // QT_NO_TOOLTIP
720         iButton2->setText(QString());
721 #ifndef QT_NO_TOOLTIP
722         iButtonO->setToolTip(QApplication::translate("MPerfWidget", "Tip: alt+click to set all
    colors at once", Q_NULLPTR));
723 #endif // QT_NO_TOOLTIP
724         iButtonO->setText(QString());
725 #ifndef QT_NO_TOOLTIP
726         iButton1->setToolTip(QApplication::translate("MPerfWidget", "Tip: alt+click to set all
    colors at once", Q_NULLPTR));
727 #endif // QT_NO_TOOLTIP
728         iButton1->setText(QString());
729 #ifndef QT_NO_TOOLTIP
730         iButton0->setToolTip(QApplication::translate("MPerfWidget", "Tip: alt+click to set all
    colors at once", Q_NULLPTR));
731 #endif // QT_NO_TOOLTIP
732         iButton0->setText(QString());
733         label_3->setText(QApplication::translate("MPerfWidget", "You can assign other DPIs in the
    Binding tab.", Q_NULLPTR));
734         label_6->setText(QApplication::translate("MPerfWidget", "2:", Q_NULLPTR));
735         label_7->setText(QApplication::translate("MPerfWidget", "3:", Q_NULLPTR));
736         label_8->setText(QApplication::translate("MPerfWidget", "4:", Q_NULLPTR));
737         label_5->setText(QApplication::translate("MPerfWidget", "1:", Q_NULLPTR));
738         label_15->setText(QApplication::translate("MPerfWidget", "Other:", Q_NULLPTR));
739         label_13->setText(QApplication::translate("MPerfWidget", "5:", Q_NULLPTR));
740         label_14->setText(QApplication::translate("MPerfWidget", "Sniper:", Q_NULLPTR));
741 #ifndef QT_NO_TOOLTIP
742         eBox4->setToolTip(QApplication::translate("MPerfWidget", "Uncheck this box to skip the stage
    while moving up or down.", Q_NULLPTR));
743 #endif // QT_NO_TOOLTIP
744         eBox4->setText(QString());
745         label_11->setText(QApplication::translate("MPerfWidget", "DPI Stage:", Q_NULLPTR));
746 #ifndef QT_NO_TOOLTIP
747         eBox3->setToolTip(QApplication::translate("MPerfWidget", "Uncheck this box to skip the stage
    while moving up or down.", Q_NULLPTR));
748 #endif // QT_NO_TOOLTIP
749         eBox3->setText(QString());
750 #ifndef QT_NO_TOOLTIP
751         eBox1->setToolTip(QApplication::translate("MPerfWidget", "Uncheck this box to skip the stage
    while moving up or down.", Q_NULLPTR));
752 #endif // QT_NO_TOOLTIP
753         eBox1->setText(QString());
754         label_2->setText(QApplication::translate("MPerfWidget", "Movement", Q_NULLPTR));
755         aSnapBox->setText(QApplication::translate("MPerfWidget", "Angle snap", Q_NULLPTR));
756         indicBox->setText(QApplication::translate("MPerfWidget", "Use DPI indicator:", Q_NULLPTR));
757 #ifndef QT_NO_TOOLTIP
758         spinBox->setToolTip(QApplication::translate("MPerfWidget", "Indicator intensity", Q_NULLPTR)
    );
759 #endif // QT_NO_TOOLTIP
760         spinBox->setSuffix(QApplication::translate("MPerfWidget", "%", Q_NULLPTR));
761         xyBox->setText(QApplication::translate("MPerfWidget", "Independent X/Y states", Q_NULLPTR));
762         copyButton->setText(QApplication::translate("MPerfWidget", "Copy to mode...", Q_NULLPTR))
    ;
763     } // retranslateUi
```

Here is the caller graph for this function:



12.54.2.2 void Ui_MPerfWidget::setupUi (QWidget * *MPerfWidget*) [inline],[inherited]

Definition at line 111 of file ui_mperfwidget.h.

References Ui_MPerfWidget::aSnapBox, Ui_MPerfWidget::copyButton, Ui_MPerfWidget::eBox1, Ui_MPerfWidget::eBox2, Ui_MPerfWidget::eBox3, Ui_MPerfWidget::eBox4, Ui_MPerfWidget::eBox5, Ui_MPerfWidget::gridLayout, Ui_MPerfWidget::horizontalLayout, Ui_MPerfWidget::horizontalLayout_2, Ui_MPerfWidget::horizontalLayout_4, Ui_MPerfWidget::horizontalSpacer, Ui_MPerfWidget::horizontalSpacer_2, Ui_MPerfWidget::horizontalSpacer_3, Ui_MPerfWidget::horizontalSpacer_4, Ui_MPerfWidget::horizontalSpacer_5, Ui_MPerfWidget::horizontalSpacer_6, Ui_MPerfWidget::horizontalSpacer_7, Ui_MPerfWidget::iButton0, Ui_MPerfWidget::iButton1, Ui_MPerfWidget::iButton2, Ui_MPerfWidget::iButton3, Ui_MPerfWidget::iButton4, Ui_MPerfWidget::iButton5, Ui_MPerfWidget::iButtonO, Ui_MPerfWidget::indicBox, Ui_MPerfWidget::label, Ui_MPerfWidget::label_11, Ui_MPerfWidget::label_12, Ui_MPerfWidget::label_13, Ui_MPerfWidget::label_14, Ui_MPerfWidget::label_15, Ui_MPerfWidget::label_2, Ui_MPerfWidget::label_25, Ui_MPerfWidget::label_3, Ui_MPerfWidget::label_4, Ui_MPerfWidget::label_5, Ui_MPerfWidget::label_6, Ui_MPerfWidget::label_7, Ui_MPerfWidget::label_8, Ui_MPerfWidget::lHeightBox, Ui_MPerfWidget::line, Ui_MPerfWidget::line_2, Ui_MPerfWidget::line_3, Ui_MPerfWidget::retranslateUi(), Ui_MPerfWidget::spinBox, Ui_MPerfWidget::verticalLayout, Ui_MPerfWidget::verticalSpacer, Ui_MPerfWidget::verticalSpacer_2, Ui_MPerfWidget::verticalSpacer_3, Ui_MPerfWidget::xBox0, Ui_MPerfWidget::xBox1, Ui_MPerfWidget::xBox2, Ui_MPerfWidget::xBox3, Ui_MPerfWidget::xBox4, Ui_MPerfWidget::xBox5, Ui_MPerfWidget::xSlider0, Ui_MPerfWidget::xSlider1, Ui_MPerfWidget::xSlider2, Ui_MPerfWidget::xSlider3, Ui_MPerfWidget::xSlider4, Ui_MPerfWidget::xSlider5, Ui_MPerfWidget::xyBox, Ui_MPerfWidget::yBox0, Ui_MPerfWidget::yBox1, Ui_MPerfWidget::yBox2, Ui_MPerfWidget::yBox3, Ui_MPerfWidget::yBox4, Ui_MPerfWidget::yBox5, Ui_MPerfWidget::ySlider0, Ui_MPerfWidget::ySlider1, Ui_MPerfWidget::ySlider2, Ui_MPerfWidget::ySlider3, Ui_MPerfWidget::ySlider4, and Ui_MPerfWidget::ySlider5.

Referenced by MPerfWidget::MPerfWidget().

```
112     {
113         if (MPerfWidget->objectName().isEmpty())
114             MPerfWidget->setObjectName(QStringLiteral("MPerfWidget"));
115         MPerfWidget->resize(932, 637);
116         verticalLayout = new QVBoxLayout(MPerfWidget);
117         verticalLayout->setObjectName(QStringLiteral("verticalLayout"));
118         label = new QLabel(MPerfWidget);
119         label->setObjectName(QStringLiteral("label"));
120         QFont font;
121         font.setBold(true);
122         font.setWeight(75);
123         label->setFont(font);
124 
125         verticalLayout->addWidget(label);
126 
127         line = new QFrame(MPerfWidget);
128         line->setObjectName(QStringLiteral("line"));
129         line->setFrameShape(QFrame::HLine);
130         line->setFrameShadow(QFrame::Sunken);
131 
132         verticalLayout->addWidget(line);
133 
134         gridLayout = new QGridLayout();
135         gridLayout->setObjectName(QStringLiteral("gridLayout"));
136         label_25 = new QLabel(MPerfWidget);
137         label_25->setObjectName(QStringLiteral("label_25"));
138         label_25->setAlignment(Qt::AlignRight|Qt::AlignTrailing|Qt::AlignVCenter);
139 
140         gridLayout->addWidget(label_25, 16, 5, 1, 2);
141 
142         horizontalLayout_2 = new QHBoxLayout();
143         horizontalLayout_2->setObjectName(QStringLiteral("horizontalLayout_2"));
```

```
144         lHeightBox = new QComboBox(MPerfWidget);
145         lHeightBox->setObjectName(QStringLiteral("lHeightBox"));
146 
147         horizontalLayout_2->addWidget(lHeightBox);
148 
149         horizontalSpacer_2 = new QSpacerItem(40, 20, QSizePolicy::Expanding,
    QSizePolicy::Minimum);
150 
151         horizontalLayout_2->addItem(horizontalSpacer_2);
152 
153 
154         gridLayout->addLayout(horizontalLayout_2, 16, 8, 1, 3);
155 
156         eBox2 = new QCheckBox(MPerfWidget);
157         eBox2->setObjectName(QStringLiteral("eBox2"));
158 
159         gridLayout->addWidget(eBox2, 7, 1, 1, 1);
160 
161         ySlider3 = new QSlider(MPerfWidget);
162         ySlider3->setObjectName(QStringLiteral("ySlider3"));
163         ySlider3->setMinimum(0);
164         ySlider3->setMaximum(1000);
165         ySlider3->setSingleStep(10);
166         ySlider3->setPageStep(50);
167         ySlider3->setValue(0);
168         ySlider3->setOrientation(Qt::Horizontal);
169 
170         gridLayout->addWidget(ySlider3, 8, 8, 1, 1);
171 
172         xSlider2 = new QSlider(MPerfWidget);
173         xSlider2->setObjectName(QStringLiteral("xSlider2"));
174         xSlider2->setMinimum(0);
175         xSlider2->setMaximum(1000);
176         xSlider2->setSingleStep(10);
177         xSlider2->setPageStep(50);
178         xSlider2->setValue(0);
179         xSlider2->setOrientation(Qt::Horizontal);
180 
181         gridLayout->addWidget(xSlider2, 7, 5, 1, 1);
182 
183         xBox3 = new QSpinBox(MPerfWidget);
184         xBox3->setObjectName(QStringLiteral("xBox3"));
185         xBox3->setMinimum(100);
186         xBox3->setMaximum(12000);
187         xBox3->setSingleStep(50);
188 
189         gridLayout->addWidget(xBox3, 8, 6, 1, 1);
190 
191         yBox0 = new QSpinBox(MPerfWidget);
192         yBox0->setObjectName(QStringLiteral("yBox0"));
193         yBox0->setMinimum(100);
194         yBox0->setMaximum(12000);
195         yBox0->setSingleStep(50);
196 
197         gridLayout->addWidget(yBox0, 11, 9, 1, 1);
198 
199         ySlider0 = new QSlider(MPerfWidget);
200         ySlider0->setObjectName(QStringLiteral("ySlider0"));
201         ySlider0->setMinimum(0);
202         ySlider0->setMaximum(1000);
203         ySlider0->setSingleStep(10);
204         ySlider0->setPageStep(50);
205         ySlider0->setValue(0);
206         ySlider0->setOrientation(Qt::Horizontal);
207 
208         gridLayout->addWidget(ySlider0, 11, 8, 1, 1);
209 
210         iButton4 = new ColorButton(MPerfWidget);
211         iButton4->setObjectName(QStringLiteral("iButton4"));
212         QSizePolicy sizePolicy(QSizePolicy::Fixed, QSizePolicy::Fixed);
213         sizePolicy.setHorizontalStretch(0);
214         sizePolicy.setVerticalStretch(0);
215         sizePolicy.setHeightForWidth(iButton4->sizePolicy().hasHeightForWidth());
216         iButton4->setSizePolicy(sizePolicy);
217         iButton4->setMinimumSize(QSize(0, 40));
218 
219         gridLayout->addWidget(iButton4, 9, 2, 1, 1);
220 
221         iButton5 = new ColorButton(MPerfWidget);
222         iButton5->setObjectName(QStringLiteral("iButton5"));
223         sizePolicy.setHeightForWidth(iButton5->sizePolicy().hasHeightForWidth());
224         iButton5->setSizePolicy(sizePolicy);
225         iButton5->setMinimumSize(QSize(0, 40));
226 
227         gridLayout->addWidget(iButton5, 10, 2, 1, 1);
228 
229         horizontalSpacer_4 = new QSpacerItem(5, 20, QSizePolicy::Fixed,
```

```
    QSizePolicy::Minimum);
230
231         gridLayout->addItem(horizontalSpacer_4, 4, 4, 1, 1);
232
233         verticalSpacer_3 = new QSpacerItem(20, 15, QSizePolicy::Minimum, QSizePolicy::Fixed
    );
234
235         gridLayout->addItem(verticalSpacer_3, 3, 2, 1, 1);
236
237         horizontalSpacer_5 = new QSpacerItem(5, 20, QSizePolicy::Fixed,
    QSizePolicy::Minimum);
238
239         gridLayout->addItem(horizontalSpacer_5, 4, 7, 1, 1);
240
241         iButton3 = new ColorButton(MPerfWidget);
242         iButton3->setObjectName(QStringLiteral("iButton3"));
243         sizePolicy.setHeightForWidth(iButton3->sizePolicy().hasHeightForWidth());
244         iButton3->setSizePolicy(sizePolicy);
245         iButton3->setMinimumSize(QSize(0, 40));
246
247         gridLayout->addWidget(iButton3, 8, 2, 1, 1);
248
249         label_12 = new QLabel(MPerfWidget);
250         label_12->setObjectName(QStringLiteral("label_12"));
251         QSizePolicy sizePolicy1(QSizePolicy::Expanding, QSizePolicy::Preferred);
252         sizePolicy1.setHorizontalStretch(0);
253         sizePolicy1.setVerticalStretch(0);
254         sizePolicy1.setHeightForWidth(label_12->sizePolicy().hasHeightForWidth());
255         label_12->setSizePolicy(sizePolicy1);
256
257         gridLayout->addWidget(label_12, 4, 5, 1, 1);
258
259         xBox4 = new QSpinBox(MPerfWidget);
260         xBox4->setObjectName(QStringLiteral("xBox4"));
261         xBox4->setMinimum(100);
262         xBox4->setMaximum(12000);
263         xBox4->setSingleStep(50);
264
265         gridLayout->addWidget(xBox4, 9, 6, 1, 1);
266
267         xBox0 = new QSpinBox(MPerfWidget);
268         xBox0->setObjectName(QStringLiteral("xBox0"));
269         xBox0->setMinimum(100);
270         xBox0->setMaximum(12000);
271         xBox0->setSingleStep(50);
272
273         gridLayout->addWidget(xBox0, 11, 6, 1, 1);
274
275         yBox5 = new QSpinBox(MPerfWidget);
276         yBox5->setObjectName(QStringLiteral("yBox5"));
277         yBox5->setMinimum(100);
278         yBox5->setMaximum(12000);
279         yBox5->setSingleStep(50);
280
281         gridLayout->addWidget(yBox5, 10, 9, 1, 1);
282
283         eBox5 = new QCheckBox(MPerfWidget);
284         eBox5->setObjectName(QStringLiteral("eBox5"));
285
286         gridLayout->addWidget(eBox5, 10, 1, 1, 1);
287
288         xSlider5 = new QSlider(MPerfWidget);
289         xSlider5->setObjectName(QStringLiteral("xSlider5"));
290         xSlider5->setMinimum(0);
291         xSlider5->setMaximum(1000);
292         xSlider5->setSingleStep(10);
293         xSlider5->setPageStep(50);
294         xSlider5->setValue(0);
295         xSlider5->setOrientation(Qt::Horizontal);
296
297         gridLayout->addWidget(xSlider5, 10, 5, 1, 1);
298
299         xBox1 = new QSpinBox(MPerfWidget);
300         xBox1->setObjectName(QStringLiteral("xBox1"));
301         sizePolicy.setHeightForWidth(xBox1->sizePolicy().hasHeightForWidth());
302         xBox1->setSizePolicy(sizePolicy);
303         xBox1->setMinimum(100);
304         xBox1->setMaximum(12000);
305         xBox1->setSingleStep(50);
306
307         gridLayout->addWidget(xBox1, 6, 6, 1, 1);
308
309         yBox2 = new QSpinBox(MPerfWidget);
310         yBox2->setObjectName(QStringLiteral("yBox2"));
311         yBox2->setMinimum(100);
312         yBox2->setMaximum(12000);
313         yBox2->setSingleStep(50);
```

```
314
315         gridLayout->addWidget(yBox2, 7, 9, 1, 1);
316
317         yBox3 = new QSpinBox(MPerfWidget);
318         yBox3->setObjectName(QStringLiteral("yBox3"));
319         yBox3->setMinimum(100);
320         yBox3->setMaximum(12000);
321         yBox3->setSingleStep(50);
322
323         gridLayout->addWidget(yBox3, 8, 9, 1, 1);
324
325         xSlider4 = new QSlider(MPerfWidget);
326         xSlider4->setObjectName(QStringLiteral("xSlider4"));
327         xSlider4->setMinimum(0);
328         xSlider4->setMaximum(1000);
329         xSlider4->setSingleStep(10);
330         xSlider4->setPageStep(50);
331         xSlider4->setValue(0);
332         xSlider4->setOrientation(Qt::Horizontal);
333
334         gridLayout->addWidget(xSlider4, 9, 5, 1, 1);
335
336         label_4 = new QLabel(MPerfWidget);
337         label_4->setObjectName(QStringLiteral("label_4"));
338         sizePolicy1.setHeightForWidth(label_4->sizePolicy().hasHeightForWidth());
339         label_4->setSizePolicy(sizePolicy1);
340
341         gridLayout->addWidget(label_4, 4, 8, 1, 1);
342
343         xSlider0 = new QSlider(MPerfWidget);
344         xSlider0->setObjectName(QStringLiteral("xSlider0"));
345         xSlider0->setMinimum(0);
346         xSlider0->setMaximum(1000);
347         xSlider0->setSingleStep(10);
348         xSlider0->setPageStep(50);
349         xSlider0->setValue(0);
350         xSlider0->setOrientation(Qt::Horizontal);
351
352         gridLayout->addWidget(xSlider0, 11, 5, 1, 1);
353
354         iButton2 = new ColorButton(MPerfWidget);
355         iButton2->setObjectName(QStringLiteral("iButton2"));
356         sizePolicy.setHeightForWidth(iButton2->sizePolicy().hasHeightForWidth());
357         iButton2->setSizePolicy(sizePolicy);
358         iButton2->setMinimumSize(QSize(0, 40));
359
360         gridLayout->addWidget(iButton2, 7, 2, 1, 1);
361
362         line_3 = new QFrame(MPerfWidget);
363         line_3->setObjectName(QStringLiteral("line_3"));
364         line_3->setFrameShape(QFrame::HLine);
365         line_3->setFrameShadow(QFrame::Sunken);
366
367         gridLayout->addWidget(line_3, 5, 0, 1, 11);
368
369         iButtonO = new ColorButton(MPerfWidget);
370         iButtonO->setObjectName(QStringLiteral("iButtonO"));
371         sizePolicy.setHeightForWidth(iButtonO->sizePolicy().hasHeightForWidth());
372         iButtonO->setSizePolicy(sizePolicy);
373         iButtonO->setMinimumSize(QSize(0, 40));
374
375         gridLayout->addWidget(iButtonO, 12, 2, 1, 1);
376
377         yBox1 = new QSpinBox(MPerfWidget);
378         yBox1->setObjectName(QStringLiteral("yBox1"));
379         sizePolicy.setHeightForWidth(yBox1->sizePolicy().hasHeightForWidth());
380         yBox1->setSizePolicy(sizePolicy);
381         yBox1->setMinimum(100);
382         yBox1->setMaximum(12000);
383         yBox1->setSingleStep(50);
384
385         gridLayout->addWidget(yBox1, 6, 9, 1, 1);
386
387         ySlider2 = new QSlider(MPerfWidget);
388         ySlider2->setObjectName(QStringLiteral("ySlider2"));
389         ySlider2->setMinimum(0);
390         ySlider2->setMaximum(1000);
391         ySlider2->setSingleStep(10);
392         ySlider2->setPageStep(50);
393         ySlider2->setValue(0);
394         ySlider2->setOrientation(Qt::Horizontal);
395
396         gridLayout->addWidget(ySlider2, 7, 8, 1, 1);
397
398         xBox2 = new QSpinBox(MPerfWidget);
399         xBox2->setObjectName(QStringLiteral("xBox2"));
400         xBox2->setMinimum(100);
```

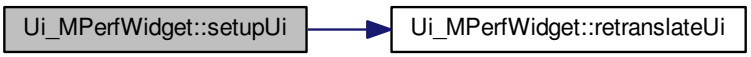
```
401         xBox2->setMaximum(12000);
402         xBox2->setSingleStep(50);
403 
404         gridLayout->addWidget(xBox2, 7, 6, 1, 1);
405 
406         xSlider1 = new QSlider(MPerfWidget);
407         xSlider1->setObjectName(QStringLiteral("xSlider1"));
408         QSizePolicy sizePolicy2(QSizePolicy::Expanding, QSizePolicy::Fixed);
409         sizePolicy2.setHorizontalStretch(0);
410         sizePolicy2.setVerticalStretch(0);
411         sizePolicy2.setHeightForWidth(xSlider1->sizePolicy().hasHeightForWidth());
412         xSlider1->setSizePolicy(sizePolicy2);
413         xSlider1->setMinimum(0);
414         xSlider1->setMaximum(1000);
415         xSlider1->setSingleStep(10);
416         xSlider1->setPageStep(50);
417         xSlider1->setValue(0);
418         xSlider1->setOrientation(Qt::Horizontal);
419 
420         gridLayout->addWidget(xSlider1, 6, 5, 1, 1);
421 
422         horizontalSpacer = new QSpacerItem(5, 20, QSizePolicy::Fixed, QSizePolicy::Minimum)
    ;
423 
424         gridLayout->addItem(horizontalSpacer, 0, 0, 1, 1);
425 
426         yBox4 = new QSpinBox(MPerfWidget);
427         yBox4->setObjectName(QStringLiteral("yBox4"));
428         yBox4->setMinimum(100);
429         yBox4->setMaximum(12000);
430         yBox4->setSingleStep(50);
431 
432         gridLayout->addWidget(yBox4, 9, 9, 1, 1);
433 
434         iButton1 = new ColorButton(MPerfWidget);
435         iButton1->setObjectName(QStringLiteral("iButton1"));
436         sizePolicy.setHeightForWidth(iButton1->sizePolicy().hasHeightForWidth());
437         iButton1->setSizePolicy(sizePolicy);
438         iButton1->setMinimumSize(QSize(0, 40));
439 
440         gridLayout->addWidget(iButton1, 6, 2, 1, 1);
441 
442         ySlider4 = new QSlider(MPerfWidget);
443         ySlider4->setObjectName(QStringLiteral("ySlider4"));
444         ySlider4->setMinimum(0);
445         ySlider4->setMaximum(1000);
446         ySlider4->setSingleStep(10);
447         ySlider4->setPageStep(50);
448         ySlider4->setValue(0);
449         ySlider4->setOrientation(Qt::Horizontal);
450 
451         gridLayout->addWidget(ySlider4, 9, 8, 1, 1);
452 
453         ySlider1 = new QSlider(MPerfWidget);
454         ySlider1->setObjectName(QStringLiteral("ySlider1"));
455         ySlider1->setMinimum(0);
456         ySlider1->setMaximum(1000);
457         ySlider1->setSingleStep(10);
458         ySlider1->setPageStep(50);
459         ySlider1->setValue(0);
460         ySlider1->setOrientation(Qt::Horizontal);
461 
462         gridLayout->addWidget(ySlider1, 6, 8, 1, 1);
463 
464         iButton0 = new ColorButton(MPerfWidget);
465         iButton0->setObjectName(QStringLiteral("iButton0"));
466         sizePolicy.setHeightForWidth(iButton0->sizePolicy().hasHeightForWidth());
467         iButton0->setSizePolicy(sizePolicy);
468         iButton0->setMinimumSize(QSize(0, 40));
469 
470         gridLayout->addWidget(iButton0, 11, 2, 1, 1);
471 
472         xSlider3 = new QSlider(MPerfWidget);
473         xSlider3->setObjectName(QStringLiteral("xSlider3"));
474         xSlider3->setMinimum(0);
475         xSlider3->setMaximum(1000);
476         xSlider3->setSingleStep(10);
477         xSlider3->setPageStep(50);
478         xSlider3->setValue(0);
479         xSlider3->setOrientation(Qt::Horizontal);
480 
481         gridLayout->addWidget(xSlider3, 8, 5, 1, 1);
482 
483         xBox5 = new QSpinBox(MPerfWidget);
484         xBox5->setObjectName(QStringLiteral("xBox5"));
485         xBox5->setMinimum(100);
486         xBox5->setMaximum(12000);
```

```
487         xBox5->setSingleStep(50);
488
489         gridLayout->addWidget(xBox5, 10, 6, 1, 1);
490
491         label_3 = new QLabel(MPerfWidget);
492         label_3->setObjectName(QStringLiteral("label_3"));
493
494         gridLayout->addWidget(label_3, 12, 5, 1, 5);
495
496         label_6 = new QLabel(MPerfWidget);
497         label_6->setObjectName(QStringLiteral("label_6"));
498
499         gridLayout->addWidget(label_6, 7, 3, 1, 1);
500
501         ySlider5 = new QSlider(MPerfWidget);
502         ySlider5->setObjectName(QStringLiteral("ySlider5"));
503         ySlider5->setMinimum(0);
504         ySlider5->setMaximum(1000);
505         ySlider5->setSingleStep(10);
506         ySlider5->setPageStep(50);
507         ySlider5->setValue(0);
508         ySlider5->setOrientation(Qt::Horizontal);
509
510         gridLayout->addWidget(ySlider5, 10, 8, 1, 1);
511
512         label_7 = new QLabel(MPerfWidget);
513         label_7->setObjectName(QStringLiteral("label_7"));
514
515         gridLayout->addWidget(label_7, 8, 3, 1, 1);
516
517         label_8 = new QLabel(MPerfWidget);
518         label_8->setObjectName(QStringLiteral("label_8"));
519
520         gridLayout->addWidget(label_8, 9, 3, 1, 1);
521
522         label_5 = new QLabel(MPerfWidget);
523         label_5->setObjectName(QStringLiteral("label_5"));
524
525         gridLayout->addWidget(label_5, 6, 3, 1, 1);
526
527         label_15 = new QLabel(MPerfWidget);
528         label_15->setObjectName(QStringLiteral("label_15"));
529
530         gridLayout->addWidget(label_15, 12, 3, 1, 1);
531
532         label_13 = new QLabel(MPerfWidget);
533         label_13->setObjectName(QStringLiteral("label_13"));
534
535         gridLayout->addWidget(label_13, 10, 3, 1, 1);
536
537         label_14 = new QLabel(MPerfWidget);
538         label_14->setObjectName(QStringLiteral("label_14"));
539
540         gridLayout->addWidget(label_14, 11, 3, 1, 1);
541
542         eBox4 = new QCheckBox(MPerfWidget);
543         eBox4->setObjectName(QStringLiteral("eBox4"));
544
545         gridLayout->addWidget(eBox4, 9, 1, 1, 1);
546
547         label_11 = new QLabel(MPerfWidget);
548         label_11->setObjectName(QStringLiteral("label_11"));
549
550         gridLayout->addWidget(label_11, 4, 2, 1, 2);
551
552         eBox3 = new QCheckBox(MPerfWidget);
553         eBox3->setObjectName(QStringLiteral("eBox3"));
554
555         gridLayout->addWidget(eBox3, 8, 1, 1, 1);
556
557         eBox1 = new QCheckBox(MPerfWidget);
558         eBox1->setObjectName(QStringLiteral("eBox1"));
559
560         gridLayout->addWidget(eBox1, 6, 1, 1, 1);
561
562         label_2 = new QLabel(MPerfWidget);
563         label_2->setObjectName(QStringLiteral("label_2"));
564         label_2->setFont(font);
565
566         gridLayout->addWidget(label_2, 14, 0, 1, 10);
567
568         verticalSpacer = new QSpacerItem(20, 30, QSizePolicy::Minimum, QSizePolicy::Fixed);
569
570         gridLayout->addItem(verticalSpacer, 13, 2, 1, 1);
571
572         aSnapBox = new QCheckBox(MPerfWidget);
573         aSnapBox->setObjectName(QStringLiteral("aSnapBox"));
```

```
574
575         gridLayout->addWidget(aSnapBox, 16, 1, 1, 4);
576
577         line_2 = new QFrame(MPerfWidget);
578         line_2->setObjectName(QStringLiteral("line_2"));
579         line_2->setFrameShape(QFrame::HLine);
580         line_2->setFrameShadow(QFrame::Sunken);
581
582         gridLayout->addWidget(line_2, 15, 0, 1, 10);
583
584         horizontalLayout_4 = new QHBoxLayout();
585         horizontalLayout_4->setObjectName(QStringLiteral("horizontalLayout_4"));
586         indicBox = new QCheckBox(MPerfWidget);
587         indicBox->setObjectName(QStringLiteral("indicBox"));
588
589         horizontalLayout_4->addWidget(indicBox);
590
591         horizontalSpacer_3 = new QSpacerItem(5, 20, QSizePolicy::Fixed,
    QSizePolicy::Minimum);
592
593         horizontalLayout_4->addItem(horizontalSpacer_3);
594
595         spinBox = new QSpinBox(MPerfWidget);
596         spinBox->setObjectName(QStringLiteral("spinBox"));
597         spinBox->setMaximum(100);
598         spinBox->setValue(100);
599
600         horizontalLayout_4->addWidget(spinBox);
601
602         horizontalSpacer_7 = new QSpacerItem(40, 20, QSizePolicy::Expanding,
    QSizePolicy::Minimum);
603
604         horizontalLayout_4->addItem(horizontalSpacer_7);
605
606
607         gridLayout->addLayout(horizontalLayout_4, 0, 1, 1, 9);
608
609         xyBox = new QCheckBox(MPerfWidget);
610         xyBox->setObjectName(QStringLiteral("xyBox"));
611
612         gridLayout->addWidget(xyBox, 2, 1, 1, 9);
613
614
615         verticalLayout->addLayout(gridLayout);
616
617         verticalSpacer_2 = new QSpacerItem(20, 40, QSizePolicy::Minimum,
    QSizePolicy::Expanding);
618
619         verticalLayout->addItem(verticalSpacer_2);
620
621         horizontalLayout = new QHBoxLayout();
622         horizontalLayout->setObjectName(QStringLiteral("horizontalLayout"));
623         copyButton = new QPushButton(MPerfWidget);
624         copyButton->setObjectName(QStringLiteral("copyButton"));
625
626         horizontalLayout->addWidget(copyButton);
627
628         horizontalSpacer_6 = new QSpacerItem(40, 20, QSizePolicy::Expanding,
    QSizePolicy::Minimum);
629
630         horizontalLayout->addItem(horizontalSpacer_6);
631
632
633         verticalLayout->addLayout(horizontalLayout);
634
635         QWidget::setTabOrder(indicBox, spinBox);
636         QWidget::setTabOrder(spinBox, xyBox);
637         QWidget::setTabOrder(xyBox, eBox1);
638         QWidget::setTabOrder(eBox1, eBox2);
639         QWidget::setTabOrder(eBox2, eBox3);
640         QWidget::setTabOrder(eBox3, eBox4);
641         QWidget::setTabOrder(eBox4, eBox5);
642         QWidget::setTabOrder(eBox5, iButton1);
643         QWidget::setTabOrder(iButton1, iButton2);
644         QWidget::setTabOrder(iButton2, iButton3);
645         QWidget::setTabOrder(iButton3, iButton4);
646         QWidget::setTabOrder(iButton4, iButton5);
647         QWidget::setTabOrder(iButton5, iButton0);
648         QWidget::setTabOrder(iButton0, iButtonO);
649         QWidget::setTabOrder(iButtonO, xSlider1);
650         QWidget::setTabOrder(xSlider1, xSlider2);
651         QWidget::setTabOrder(xSlider2, xSlider3);
652         QWidget::setTabOrder(xSlider3, xSlider4);
653         QWidget::setTabOrder(xSlider4, xSlider5);
654         QWidget::setTabOrder(xSlider5, xSlider0);
655         QWidget::setTabOrder(xSlider0, xBox1);
656         QWidget::setTabOrder(xBox1, xBox2);
```

```
657         QWidget::setTabOrder(xBox2, xBox3);
658         QWidget::setTabOrder(xBox3, xBox4);
659         QWidget::setTabOrder(xBox4, xBox5);
660         QWidget::setTabOrder(xBox5, xBox0);
661         QWidget::setTabOrder(xBox0, ySlider1);
662         QWidget::setTabOrder(ySlider1, ySlider2);
663         QWidget::setTabOrder(ySlider2, ySlider3);
664         QWidget::setTabOrder(ySlider3, ySlider4);
665         QWidget::setTabOrder(ySlider4, ySlider5);
666         QWidget::setTabOrder(ySlider5, ySlider0);
667         QWidget::setTabOrder(ySlider0, yBox1);
668         QWidget::setTabOrder(yBox1, yBox2);
669         QWidget::setTabOrder(yBox2, yBox3);
670         QWidget::setTabOrder(yBox3, yBox4);
671         QWidget::setTabOrder(yBox4, yBox5);
672         QWidget::setTabOrder(yBox5, yBox0);
673         QWidget::setTabOrder(yBox0, aSnapBox);
674         QWidget::setTabOrder(aSnapBox, lHeightBox);
675         QWidget::setTabOrder(lHeightBox, copyButton);
676 
677         retranslateUi(MPerfWidget);
678 
679         QMetaObject::connectSlotsByName(MPerfWidget);
680     } // setupUi
```

Here is the call graph for this function:



Here is the caller graph for this function:

12.54.3 Field Documentation

12.54.3.1 QCheckBox* Ui_MPerfWidget::aSnapBox [inherited]

Definition at line 98 of file ui_mperfwidget.h.

Referenced by Ui_MPerfWidget::retranslateUi(), MPerfWidget::setPerf(), and Ui_MPerfWidget::setupUi().

12.54.3.2 QPushButton* Ui_MPerfWidget::copyButton [inherited]

Definition at line 108 of file ui_mperfwidget.h.

Referenced by Ui_MPerfWidget::retranslateUi(), and Ui_MPerfWidget::setupUi().

12.54.3.3 QCheckBox* Ui_MPerfWidget::eBox1 [inherited]

Definition at line 95 of file ui_mperfwidget.h.

Referenced by MPerfWidget::MPerfWidget(), Ui_MPerfWidget::retranslateUi(), and Ui_MPerfWidget::setupUi().

12.54.3.4 QCheckBox* Ui_MPerfWidget::eBox2 [inherited]

Definition at line 44 of file ui_mperfwidget.h.

Referenced by MPerfWidget::MPerfWidget(), Ui_MPerfWidget::retranslateUi(), and Ui_MPerfWidget::setupUi().

12.54.3.5 QCheckBox* Ui_MPerfWidget::eBox3 [inherited]

Definition at line 94 of file ui_mperfwidget.h.

Referenced by MPerfWidget::MPerfWidget(), Ui_MPerfWidget::retranslateUi(), and Ui_MPerfWidget::setupUi().

12.54.3.6 QCheckBox* Ui_MPerfWidget::eBox4 [inherited]

Definition at line 92 of file ui_mperfwidget.h.

Referenced by MPerfWidget::MPerfWidget(), Ui_MPerfWidget::retranslateUi(), and Ui_MPerfWidget::setupUi().

12.54.3.7 QCheckBox* Ui_MPerfWidget::eBox5 [inherited]

Definition at line 60 of file ui_mperfwidget.h.

Referenced by MPerfWidget::MPerfWidget(), Ui_MPerfWidget::retranslateUi(), and Ui_MPerfWidget::setupUi().

12.54.3.8 QGridLayout* Ui_MPerfWidget::gridLayout [inherited]

Definition at line 39 of file ui_mperfwidget.h.

Referenced by Ui_MPerfWidget::setupUi().

12.54.3.9 QHBoxLayout* Ui_MPerfWidget::horizontalLayout [inherited]

Definition at line 107 of file ui_mperfwidget.h.

Referenced by Ui_MPerfWidget::setupUi().

12.54.3.10 QHBoxLayout* Ui_MPerfWidget::horizontalLayout_2 [inherited]

Definition at line 41 of file ui_mperfwidget.h.

Referenced by Ui_MPerfWidget::setupUi().

12.54.3.11 QHBoxLayout* Ui_MPerfWidget::horizontalLayout_4 [inherited]

Definition at line 100 of file ui_mperfwidget.h.

Referenced by Ui_MPerfWidget::setupUi().

12.54.3.12 QSpacerItem* Ui_MPerfWidget::horizontalSpacer [inherited]

Definition at line 75 of file ui_mperfwidget.h.

Referenced by Ui_MPerfWidget::setupUi().

12.54.3.13 QSpacerItem* Ui_MPerfWidget::horizontalSpacer_2 [inherited]

Definition at line 43 of file ui_mperfwidget.h.

Referenced by Ui_MPerfWidget::setupUi().

12.54.3.14 QSpacerItem* Ui_MPerfWidget::horizontalSpacer_3 [inherited]

Definition at line 102 of file ui_mperfwidget.h.

Referenced by Ui_MPerfWidget::setupUi().

12.54.3.15 QSpacerItem* Ui_MPerfWidget::horizontalSpacer_4 [inherited]

Definition at line 52 of file ui_mperfwidget.h.

Referenced by Ui_MPerfWidget::setupUi().

12.54.3.16 QSpacerItem* Ui_MPerfWidget::horizontalSpacer_5 [inherited]

Definition at line 54 of file ui_mperfwidget.h.

Referenced by Ui_MPerfWidget::setupUi().

12.54.3.17 QSpacerItem* Ui_MPerfWidget::horizontalSpacer_6 [inherited]

Definition at line 109 of file ui_mperfwidget.h.

Referenced by Ui_MPerfWidget::setupUi().

12.54.3.18 QSpacerItem* Ui_MPerfWidget::horizontalSpacer_7 [inherited]

Definition at line 104 of file ui_mperfwidget.h.

Referenced by Ui_MPerfWidget::setupUi().

12.54.3.19 ColorButton* Ui_MPerfWidget::iButton0 [inherited]

Definition at line 80 of file ui_mperfwidget.h.

Referenced by MPerfWidget::MPerfWidget(), Ui_MPerfWidget::retranslateUi(), and Ui_MPerfWidget::setupUi().

12.54.3.20 ColorButton* Ui_MPerfWidget::iButton1 [inherited]

Definition at line 77 of file ui_mperfwidget.h.

Referenced by MPerfWidget::MPerfWidget(), Ui_MPerfWidget::retranslateUi(), and Ui_MPerfWidget::setupUi().

12.54.3.21 ColorButton* Ui_MPerfWidget::iButton2 [inherited]

Definition at line 68 of file ui_mperfwidget.h.

Referenced by MPerfWidget::MPerfWidget(), Ui_MPerfWidget::retranslateUi(), and Ui_MPerfWidget::setupUi().

12.54.3.22 ColorButton* Ui_MPerfWidget::iButton3 [inherited]

Definition at line 55 of file ui_mperfwidget.h.

Referenced by MPerfWidget::MPerfWidget(), Ui_MPerfWidget::retranslateUi(), and Ui_MPerfWidget::setupUi().

12.54.3.23 ColorButton* Ui_MPerfWidget::iButton4 [inherited]

Definition at line 50 of file ui_mperfwidget.h.

Referenced by MPerfWidget::MPerfWidget(), Ui_MPerfWidget::retranslateUi(), and Ui_MPerfWidget::setupUi().

12.54.3.24 ColorButton* Ui_MPerfWidget::iButton5 [inherited]

Definition at line 51 of file ui_mperfwidget.h.

Referenced by MPerfWidget::MPerfWidget(), Ui_MPerfWidget::retranslateUi(), and Ui_MPerfWidget::setupUi().

12.54.3.25 ColorButton* Ui_MPerfWidget::iButtonO [inherited]

Definition at line 70 of file ui_mperfwidget.h.

Referenced by MPerfWidget::colorChanged(), MPerfWidget::MPerfWidget(), Ui_MPerfWidget::retranslateUi(), MPerfWidget::setPerf(), and Ui_MPerfWidget::setupUi().

12.54.3.26 QCheckBox* Ui_MPerfWidget::indicBox [inherited]

Definition at line 101 of file ui_mperfwidget.h.

Referenced by Ui_MPerfWidget::retranslateUi(), MPerfWidget::setPerf(), and Ui_MPerfWidget::setupUi().

12.54.3.27 QLabel* Ui_MPerfWidget::label [inherited]

Definition at line 37 of file ui_mperfwidget.h.

Referenced by Ui_MPerfWidget::retranslateUi(), and Ui_MPerfWidget::setupUi().

12.54.3.28 QLabel* Ui_MPerfWidget::label_11 [inherited]

Definition at line 93 of file ui_mperfwidget.h.

Referenced by Ui_MPerfWidget::retranslateUi(), and Ui_MPerfWidget::setupUi().

12.54.3.29 QLabel* Ui_MPerfWidget::label_12 [inherited]

Definition at line 56 of file ui_mperfwidget.h.

Referenced by Ui_MPerfWidget::retranslateUi(), and Ui_MPerfWidget::setupUi().

12.54.3.30 QLabel* Ui_MPerfWidget::label_13 [inherited]

Definition at line 90 of file ui_mperfwidget.h.

Referenced by Ui_MPerfWidget::retranslateUi(), and Ui_MPerfWidget::setupUi().

12.54.3.31 QLabel* Ui_MPerfWidget::label_14 [inherited]

Definition at line 91 of file ui_mperfwidget.h.

Referenced by Ui_MPerfWidget::retranslateUi(), and Ui_MPerfWidget::setupUi().

12.54.3.32 QLabel* Ui_MPerfWidget::label_15 [inherited]

Definition at line 89 of file ui_mperfwidget.h.

Referenced by Ui_MPerfWidget::retranslateUi(), and Ui_MPerfWidget::setupUi().

12.54.3.33 QLabel* Ui_MPerfWidget::label_2 [inherited]

Definition at line 96 of file ui_mperfwidget.h.

Referenced by Ui_MPerfWidget::retranslateUi(), and Ui_MPerfWidget::setupUi().

12.54.3.34 QLabel* Ui_MPerfWidget::label_25 [inherited]

Definition at line 40 of file ui_mperfwidget.h.

Referenced by Ui_MPerfWidget::retranslateUi(), and Ui_MPerfWidget::setupUi().

12.54.3.35 QLabel* Ui_MPerfWidget::label_3 [inherited]

Definition at line 83 of file ui_mperfwidget.h.

Referenced by Ui_MPerfWidget::retranslateUi(), and Ui_MPerfWidget::setupUi().

12.54.3.36 QLabel* Ui_MPerfWidget::label_4 [inherited]

Definition at line 66 of file ui_mperfwidget.h.

Referenced by Ui_MPerfWidget::retranslateUi(), and Ui_MPerfWidget::setupUi().

12.54.3.37 QLabel* Ui_MPerfWidget::label_5 [inherited]

Definition at line 88 of file ui_mperfwidget.h.

Referenced by Ui_MPerfWidget::retranslateUi(), and Ui_MPerfWidget::setupUi().

12.54.3.38 QLabel* Ui_MPerfWidget::label_6 [inherited]

Definition at line 84 of file ui_mperfwidget.h.

Referenced by Ui_MPerfWidget::retranslateUi(), and Ui_MPerfWidget::setupUi().

12.54.3.39 QLabel* Ui_MPerfWidget::label_7 [inherited]

Definition at line 86 of file ui_mperfwidget.h.

Referenced by Ui_MPerfWidget::retranslateUi(), and Ui_MPerfWidget::setupUi().

12.54.3.40 QLabel* Ui_MPerfWidget::label_8 [inherited]

Definition at line 87 of file ui_mperfwidget.h.

Referenced by Ui_MPerfWidget::retranslateUi(), and Ui_MPerfWidget::setupUi().

12.54.3.41 QComboBox* Ui_MPerfWidget::lHeightBox [inherited]

Definition at line 42 of file ui_mperfwidget.h.

Referenced by Ui_MPerfWidget::retranslateUi(), MPerfWidget::setPerf(), and Ui_MPerfWidget::setupUi().

12.54.3.42 QFrame* Ui_MPerfWidget::line [inherited]

Definition at line 38 of file ui_mperfwidget.h.

Referenced by Ui_MPerfWidget::setupUi().

12.54.3.43 QFrame* Ui_MPerfWidget::line_2 [inherited]

Definition at line 99 of file ui_mperfwidget.h.

Referenced by Ui_MPerfWidget::setupUi().

12.54.3.44 QFrame* Ui_MPerfWidget::line_3 [inherited]

Definition at line 69 of file ui_mperfwidget.h.

Referenced by Ui_MPerfWidget::setupUi().

12.54.3.45 QSpinBox* Ui_MPerfWidget::spinBox [inherited]

Definition at line 103 of file ui_mperfwidget.h.

Referenced by Ui_MPerfWidget::retranslateUi(), MPerfWidget::setPerf(), and Ui_MPerfWidget::setupUi().

12.54.3.46 QVBoxLayout* Ui_MPerfWidget::verticalLayout [inherited]

Definition at line 36 of file ui_mperfwidget.h.

Referenced by Ui_MPerfWidget::setupUi().

12.54.3.47 QSpacerItem* Ui_MPerfWidget::verticalSpacer [inherited]

Definition at line 97 of file ui_mperfwidget.h.

Referenced by Ui_MPerfWidget::setupUi().

12.54.3.48 QSpacerItem* Ui_MPerfWidget::verticalSpacer_2 [inherited]

Definition at line 106 of file ui_mperfwidget.h.

Referenced by Ui_MPerfWidget::setupUi().

12.54.3.49 QSpacerItem* Ui_MPerfWidget::verticalSpacer_3 [inherited]

Definition at line 53 of file ui_mperfwidget.h.

Referenced by Ui_MPerfWidget::setupUi().

12.54.3.50 QSpinBox* Ui_MPerfWidget::xBox0 [inherited]

Definition at line 58 of file ui_mperfwidget.h.

Referenced by MPerfWidget::MPerfWidget(), and Ui_MPerfWidget::setupUi().

12.54.3.51 QSpinBox* Ui_MPerfWidget::xBox1 [inherited]

Definition at line 62 of file ui_mperfwidget.h.

Referenced by MPerfWidget::MPerfWidget(), and Ui_MPerfWidget::setupUi().

12.54.3.52 QSpinBox* Ui_MPerfWidget::xBox2 [inherited]

Definition at line 73 of file ui_mperfwidget.h.

Referenced by MPerfWidget::MPerfWidget(), and Ui_MPerfWidget::setupUi().

12.54.3.53 QSpinBox* Ui_MPerfWidget::xBox3 [inherited]

Definition at line 47 of file ui_mperfwidget.h.

Referenced by MPerfWidget::MPerfWidget(), and Ui_MPerfWidget::setupUi().

12.54.3.54 QSpinBox* Ui_MPerfWidget::xBox4 [inherited]

Definition at line 57 of file ui_mperfwidget.h.

Referenced by MPerfWidget::MPerfWidget(), and Ui_MPerfWidget::setupUi().

12.54.3.55 QSpinBox* Ui_MPerfWidget::xBox5 [inherited]

Definition at line 82 of file ui_mperfwidget.h.

Referenced by MPerfWidget::MPerfWidget(), and Ui_MPerfWidget::setupUi().

12.54.3.56 QSlider* Ui_MPerfWidget::xSlider0 [inherited]

Definition at line 67 of file ui_mperfwidget.h.

Referenced by MPerfWidget::MPerfWidget(), and Ui_MPerfWidget::setupUi().

12.54.3.57 QSlider* Ui_MPerfWidget::xSlider1 [inherited]

Definition at line 74 of file ui_mperfwidget.h.

Referenced by MPerfWidget::MPerfWidget(), and Ui_MPerfWidget::setupUi().

12.54.3.58 QSlider* Ui_MPerfWidget::xSlider2 [inherited]

Definition at line 46 of file ui_mperfwidget.h.

Referenced by MPerfWidget::MPerfWidget(), and Ui_MPerfWidget::setupUi().

12.54.3.59 QSlider* Ui_MPerfWidget::xSlider3 [inherited]

Definition at line 81 of file ui_mperfwidget.h.

Referenced by MPerfWidget::MPerfWidget(), and Ui_MPerfWidget::setupUi().

12.54.3.60 QSlider* Ui_MPerfWidget::xSlider4 [inherited]

Definition at line 65 of file ui_mperfwidget.h.

Referenced by MPerfWidget::MPerfWidget(), and Ui_MPerfWidget::setupUi().

12.54.3.61 QSlider* Ui_MPerfWidget::xSlider5 [inherited]

Definition at line 61 of file ui_mperfwidget.h.

Referenced by MPerfWidget::MPerfWidget(), and Ui_MPerfWidget::setupUi().

12.54.3.62 QCheckBox* Ui_MPerfWidget::xyBox [inherited]

Definition at line 105 of file ui_mperfwidget.h.

Referenced by MPerfWidget::MPerfWidget(), Ui_MPerfWidget::retranslateUi(), Ui_MPerfWidget::setupUi(), and MPerfWidget::sliderYMoved().

12.54.3.63 QSpinBox* Ui_MPerfWidget::yBox0 [inherited]

Definition at line 48 of file ui_mperfwidget.h.

Referenced by MPerfWidget::MPerfWidget(), and Ui_MPerfWidget::setupUi().

12.54.3.64 QSpinBox* Ui_MPerfWidget::yBox1 [inherited]

Definition at line 71 of file ui_mperfwidget.h.

Referenced by MPerfWidget::MPerfWidget(), and Ui_MPerfWidget::setupUi().

12.54.3.65 QSpinBox* Ui_MPerfWidget::yBox2 [inherited]

Definition at line 63 of file ui_mperfwidget.h.

Referenced by MPerfWidget::MPerfWidget(), and Ui_MPerfWidget::setupUi().

12.54.3.66 QSpinBox* Ui_MPerfWidget::yBox3 [inherited]

Definition at line 64 of file ui_mperfwidget.h.

Referenced by MPerfWidget::MPerfWidget(), and Ui_MPerfWidget::setupUi().

12.54.3.67 QSpinBox* Ui_MPerfWidget::yBox4 [inherited]

Definition at line 76 of file ui_mperfwidget.h.

Referenced by MPerfWidget::MPerfWidget(), and Ui_MPerfWidget::setupUi().

12.54.3.68 QSpinBox* Ui_MPerfWidget::yBox5 [inherited]

Definition at line 59 of file ui_mperfwidget.h.

Referenced by MPerfWidget::MPerfWidget(), and Ui_MPerfWidget::setupUi().

12.54.3.69 QSlider* Ui_MPerfWidget::ySlider0 [inherited]

Definition at line 49 of file ui_mperfwidget.h.

Referenced by MPerfWidget::MPerfWidget(), and Ui_MPerfWidget::setupUi().

12.54.3.70 QSlider* Ui_MPerfWidget::ySlider1 [inherited]

Definition at line 79 of file ui_mperfwidget.h.

Referenced by MPerfWidget::MPerfWidget(), and Ui_MPerfWidget::setupUi().

12.54.3.71 QSlider* Ui_MPerfWidget::ySlider2 [inherited]

Definition at line 72 of file ui_mperfwidget.h.

Referenced by MPerfWidget::MPerfWidget(), and Ui_MPerfWidget::setupUi().

12.54.3.72 QSlider* Ui_MPerfWidget::ySlider3 [inherited]

Definition at line 45 of file ui_mperfwidget.h.

Referenced by MPerfWidget::MPerfWidget(), and Ui_MPerfWidget::setupUi().

12.54.3.73 QSlider* Ui_MPerfWidget::ySlider4 [inherited]

Definition at line 78 of file ui_mperfwidget.h.

Referenced by MPerfWidget::MPerfWidget(), and Ui_MPerfWidget::setupUi().

12.54.3.74 QSlider* Ui_MPerfWidget::ySlider5 [inherited]

Definition at line 85 of file ui_mperfwidget.h.

Referenced by MPerfWidget::MPerfWidget(), and Ui_MPerfWidget::setupUi().

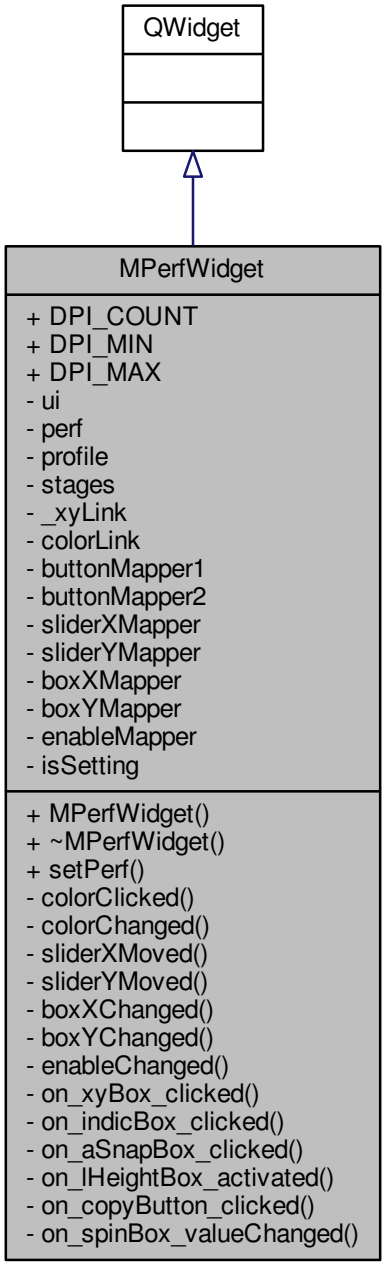
The documentation for this class was generated from the following file:

- src/ckb/ui_mperfwidget.h

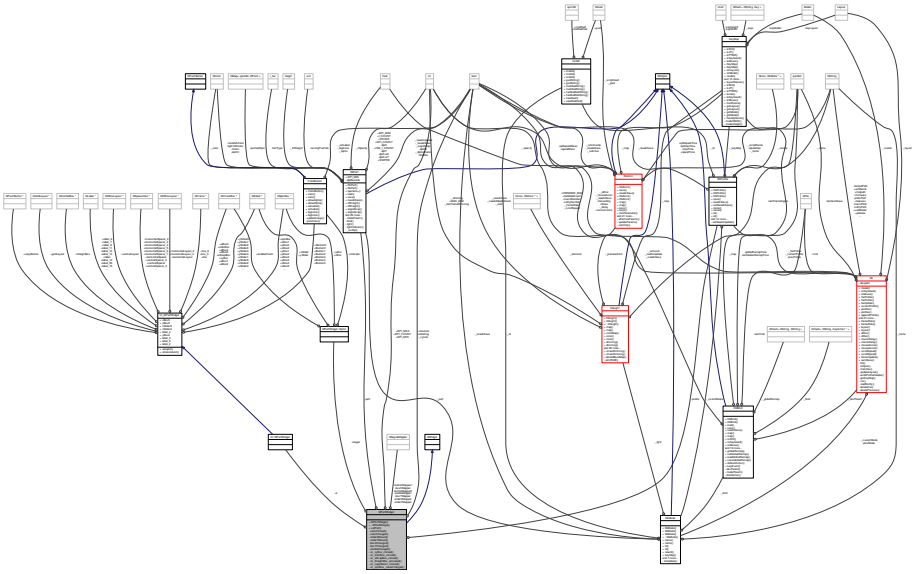
12.55 MPerfWidget Class Reference

`#include <src/ckb/mperfwidget.h>`

Inheritance diagram for MPerfWidget:



Collaboration diagram for MPerfWidget:



Data Structures

- struct DpiUi

Public Member Functions

- MPerfWidget (QWidget *parent=0)
- ~MPerfWidget ()
- void setPerf (KbPerf *newPerf, KbProfile *newProfile)

Static Public Attributes

- static const int DPI_COUNT = KbPerf::DPI_COUNT
- static const int DPI_MIN = KbPerf::DPI_MIN
- static const int DPI_MAX = KbPerf::DPI_MAX

Private Slots

- void colorClicked (int index)
- void colorChanged (int index)
- void sliderXMoved (int index)
- void sliderYMoved (int index)
- void boxXChanged (int index)
- void boxYChanged (int index)
- void enableChanged (int index)
- void on_xyBox_clicked (bool checked)
- void on_indicBox_clicked (bool checked)
- void on_aSnapBox_clicked (bool checked)
- void on_lHeightBox_activated (int index)
- void on_copyButton_clicked ()
- void on_spinBox_valueChanged (int arg1)

Private Attributes

- Ui::MPerfWidget * ui
- KbPerf * perf
- KbProfile * profile
- DpiUi stages [DPI_COUNT]
- bool _xyLink
- bool colorLink
- QSignalMapper buttonMapper1
- QSignalMapper buttonMapper2
- QSignalMapper sliderXMapper
- QSignalMapper sliderYMapper
- QSignalMapper boxXMapper
- QSignalMapper boxYMapper
- QSignalMapper enableMapper
- bool isSetting

12.55.1 Detailed Description

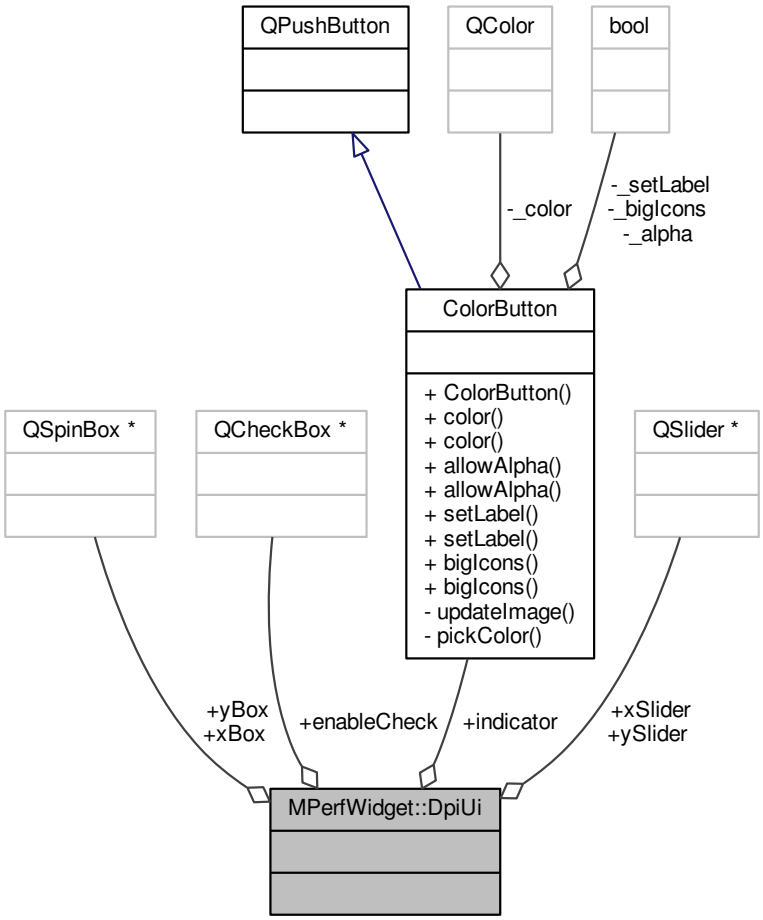
Definition at line 18 of file mperfwidget.h.

12.55.2 Data Structure Documentation

12.55.2.1 struct MPerfWidget::DpiUi

Definition at line 37 of file mperfwidget.h.

Collaboration diagram for MPerfWidget::DpiUi:



Data Fields

QCheckBox *	enableCheck	
ColorButton *	indicator	
QSpinBox *	xBox	
QSlider *	xSlider	
QSpinBox *	yBox	
QSlider *	ySlider	

12.55.3 Constructor & Destructor Documentation

12.55.3.1 MPerfWidget::MPerfWidget (QWidget * *parent* = 0) [explicit]

Definition at line 8 of file mperfwidget.cpp.

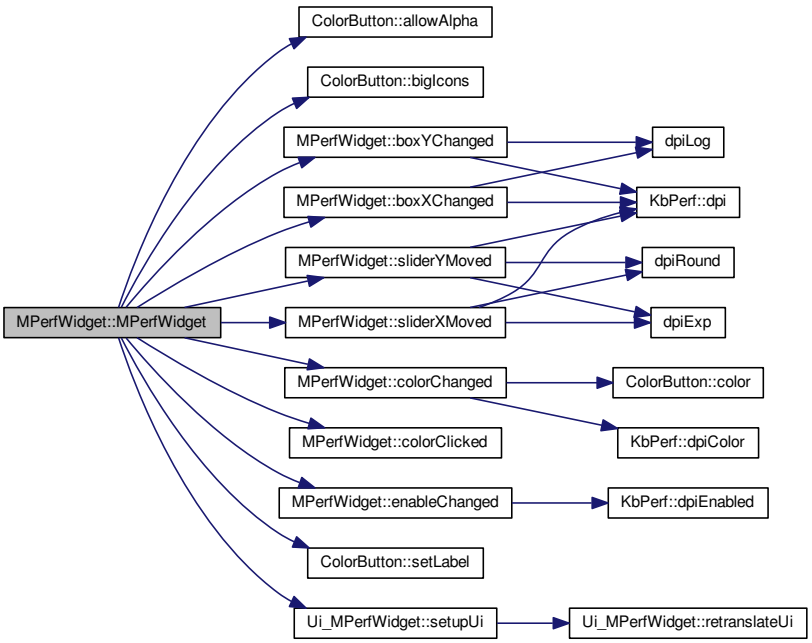
References _xyLink, ColorButton::allowAlpha(), ColorButton::bigIcons(), boxXChanged(), boxXMapper, boxYChanged(), boxYMapper, buttonMapper1, buttonMapper2, colorChanged(), colorClicked(), DPI_COUNT, Ui_MPerfWidget::eBox1, Ui_MPerfWidget::eBox2, Ui_MPerfWidget::eBox3, Ui_MPerfWidget::eBox4, Ui_MPerfWidget::eBox5, enableChanged(), MPerfWidget::DpiUi::enableCheck, enableMapper, Ui_MPerfWidget::iButton0,

Ui_MPerfWidget::iButton1, Ui_MPerfWidget::iButton2, Ui_MPerfWidget::iButton3, Ui_MPerfWidget::iButton4, Ui_MPerfWidget::iButton5, Ui_MPerfWidget::iButtonO, MPerfWidget::DpiUi::indicator, ColorButton::setLabel(), Ui_MPerfWidget::setupUi(), sliderXMapper, sliderXMoved(), sliderYMapper, sliderYMoved(), stages, ui, MPerfWidget::DpiUi::xBox, Ui_MPerfWidget::xBox0, Ui_MPerfWidget::xBox1, Ui_MPerfWidget::xBox2, Ui_MPerfWidget::xBox3, Ui_MPerfWidget::xBox4, Ui_MPerfWidget::xBox5, MPerfWidget::DpiUi::xSlider, Ui_MPerfWidget::xSlider0, Ui_MPerfWidget::xSlider1, Ui_MPerfWidget::xSlider2, Ui_MPerfWidget::xSlider3, Ui_MPerfWidget::xSlider4, Ui_MPerfWidget::xSlider5, Ui_MPerfWidget::xyBox, MPerfWidget::DpiUi::yBox, Ui_MPerfWidget::yBox0, Ui_MPerfWidget::yBox1, Ui_MPerfWidget::yBox2, Ui_MPerfWidget::yBox3, Ui_MPerfWidget::yBox4, Ui_MPerfWidget::yBox5, MPerfWidget::DpiUi::ySlider, Ui_MPerfWidget::ySlider0, Ui_MPerfWidget::ySlider1, Ui_MPerfWidget::ySlider2, Ui_MPerfWidget::ySlider3, Ui_MPerfWidget::ySlider4, and Ui_MPerfWidget::ySlider5.

```
8                                       :
9     QWidget(parent), ui(new Ui::MPerfWidget), perf(0),
      profile(0), _xyLink(!CkbSettings::get(xyLinkPath).toBool()),
      colorLink(false), isSetting(false) {
10     ui->setupUi(this);
11     ui->xyBox->setChecked(!_xyLink);
12     // Set up DPI stages
13     stages[0].indicator = ui->iButton0; stages[0].
      xSlider = ui->xSlider0; stages[0].ySlider = ui->
      ySlider0; stages[0].xBox = ui->xBox0; stages[0].
      yBox = ui->yBox0; stages[0].enableCheck = 0;
14     stages[1].indicator = ui->iButton1; stages[1].
      xSlider = ui->xSlider1; stages[1].ySlider = ui->
      ySlider1; stages[1].xBox = ui->xBox1; stages[1].
      yBox = ui->yBox1; stages[1].enableCheck = ui->
      eBox1;
15     stages[2].indicator = ui->iButton2; stages[2].
      xSlider = ui->xSlider2; stages[2].ySlider = ui->
      ySlider2; stages[2].xBox = ui->xBox2; stages[2].
      yBox = ui->yBox2; stages[2].enableCheck = ui->
      eBox2;
16     stages[3].indicator = ui->iButton3; stages[3].
      xSlider = ui->xSlider3; stages[3].ySlider = ui->
      ySlider3; stages[3].xBox = ui->xBox3; stages[3].
      yBox = ui->yBox3; stages[3].enableCheck = ui->
      eBox3;
17     stages[4].indicator = ui->iButton4; stages[4].
      xSlider = ui->xSlider4; stages[4].ySlider = ui->
      ySlider4; stages[4].xBox = ui->xBox4; stages[4].
      yBox = ui->yBox4; stages[4].enableCheck = ui->
      eBox4;
18     stages[5].indicator = ui->iButton5; stages[5].
      xSlider = ui->xSlider5; stages[5].ySlider = ui->
      ySlider5; stages[5].xBox = ui->xBox5; stages[5].
      yBox = ui->yBox5; stages[5].enableCheck = ui->
      eBox5;
19     ui->iButtonO->setLabel(false);
20     ui->iButtonO->bigIcons(true);
21     ui->iButtonO->allowAlpha(true);
22     for(int i = 0; i < DPI_COUNT; i++){
23         stages[i].indicator->setLabel(false);
24         stages[i].indicator->bigIcons(true);
25         stages[i].indicator->allowAlpha(true);
26         // Map signals
27         connect(stages[i].indicator, SIGNAL(clicked(bool)), &buttonMapper1, SLOT(map()))
      ;
28         connect(stages[i].indicator, SIGNAL(colorChanged(QColor)), &
      buttonMapper2, SLOT(map()));
29         connect(stages[i].xSlider, SIGNAL(valueChanged(int)), &
      sliderXMapper, SLOT(map()));
30         connect(stages[i].ySlider, SIGNAL(valueChanged(int)), &
      sliderYMapper, SLOT(map()));
31         connect(stages[i].xBox, SIGNAL(valueChanged(int)), &boxXMapper, SLOT(map()));
32         connect(stages[i].yBox, SIGNAL(valueChanged(int)), &boxYMapper, SLOT(map()));
33         if(stages[i].enableCheck)
34             // Sniper has no enable
35             connect(stages[i].enableCheck, SIGNAL(stateChanged(int)), &
      enableMapper, SLOT(map()));
36         // Set names
37         buttonMapper1.setMapping(stages[i].indicator, i);
38         buttonMapper2.setMapping(stages[i].indicator, i);
39         sliderXMapper.setMapping(stages[i].xSlider, i);
40         sliderYMapper.setMapping(stages[i].ySlider, i);
41         boxXMapper.setMapping(stages[i].xBox, i);
42         boxYMapper.setMapping(stages[i].yBox, i);
43         if(stages[i].enableCheck)
44             enableMapper.setMapping(stages[i].enableCheck, i);
45     }
46     // Connect to slots
47     connect(&buttonMapper1, SIGNAL(mapped(int)), this, SLOT(
      colorClicked(int)));
```

```
48     connect(&buttonMapper2, SIGNAL(mapped(int)), this, SLOT(
       colorChanged(int)));
49     connect(&sliderXMapper, SIGNAL(mapped(int)), this, SLOT(
       sliderXMoved(int)));
50     connect(&sliderYMapper, SIGNAL(mapped(int)), this, SLOT(
       sliderYMoved(int)));
51     connect(&boxXMapper, SIGNAL(mapped(int)), this, SLOT(boxXChanged(int)));
52     connect(&boxYMapper, SIGNAL(mapped(int)), this, SLOT(boxYChanged(int)));
53     connect(&enableMapper, SIGNAL(mapped(int)), this, SLOT(
       enableChanged(int)));
54 }
```

Here is the call graph for this function:



12.55.3.2 MPerfWidget::~MPerfWidget ( )

Definition at line 56 of file mperfwidget.cpp.

References ui.

```
56                          {
57     delete ui;
58 }
```

12.55.4 Member Function Documentation

12.55.4.1 void MPerfWidget::boxXChanged (int *index*) [private], [slot]

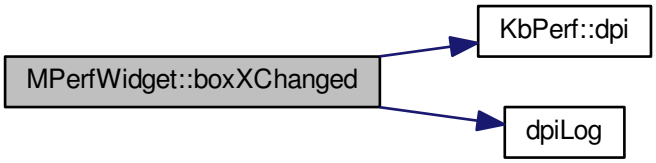
Definition at line 157 of file mperfwidget.cpp.

References _xyLink, KbPerf::dpi(), dpiLog(), perf, SET_END, SET_START, stages, MPerfWidget::DpiUi::xBox, MPerfWidget::DpiUi::xSlider, and MPerfWidget::DpiUi::yBox.

Referenced by MPerfWidget().

```
157                                     {
158     SET_START;
159     QSpinBox* box = stages[index].xBox;
160     QSlider* slider = stages[index].xSlider;
161     int value = box->value();
162     slider->setValue(dpiLog(value, slider->minimum(), slider->maximum()));
163     if(_xyLink)
164         perf->dpi(index, value);
165     else
166         perf->dpi(index, QPoint(value, perf->dpi(index).y()));
167     SET_END;
168     if(_xyLink)
169         stages[index].yBox->setValue(value);
170 }
```

Here is the call graph for this function:



Here is the caller graph for this function:

12.55.4.2 void MPerfWidget::boxYChanged (int *index*) [private],[slot]

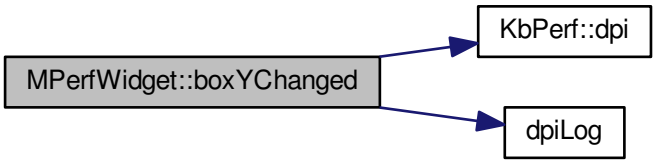
Definition at line 172 of file mperfwidget.cpp.

References _xyLink, KbPerf::dpi(), dpiLog(), perf, SET_END, SET_START, stages, MPerfWidget::DpiUi::xBox, MPerfWidget::DpiUi::yBox, and MPerfWidget::DpiUi::ySlider.

Referenced by MPerfWidget().

```
172                                     {
173     SET_START;
174     QSpinBox* box = stages[index].yBox;
175     QSlider* slider = stages[index].ySlider;
176     int value = box->value();
177     slider->setValue(dpiLog(value, slider->minimum(), slider->maximum()));
178     if(_xyLink)
179         perf->dpi(index, value);
180     else
181         perf->dpi(index, QPoint(perf->dpi(index).x(), value));
182     SET_END;
183     if(_xyLink)
184         stages[index].xBox->setValue(value);
185 }
```

Here is the call graph for this function:



Here is the caller graph for this function:

12.55.4.3 void MPerfWidget::colorChanged (int *index*) [private],[slot]

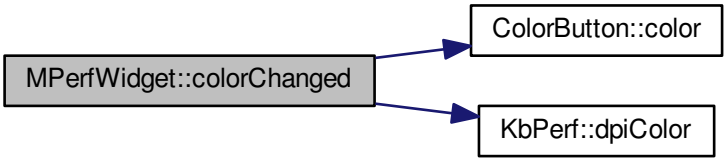
Definition at line 86 of file mperfwidget.cpp.

References ColorButton::color(), colorLink, DPI_COUNT, KbPerf::dpiColor(), Ui_MPerfWidget::iButtonO, MPerfWidget::DpiUi::indicator, KbPerf::OTHER, perf, stages, and ui.

Referenced by MPerfWidget().

```
86                                          {
87     QColor color = stages[index].indicator->color();
88     if(colorLink){
89         // Alt was held down - set all
90         for(int i = 0; i < DPI_COUNT; i++){
91             stages[i].indicator->color(color);
92             perf->dpiColor(i, color);
93         }
94         ui->iButtonO->color(color);
95         perf->dpiColor(KbPerf::OTHER, color);
96     } else {
97         // Set one
98         perf->dpiColor(index, color);
99     }
100     colorLink = false;
101 }
```

Here is the call graph for this function:



Here is the caller graph for this function:

12.55.4.4 void MPerfWidget::colorClicked (int *index*) [private],[slot]

Definition at line 81 of file mperfwidget.cpp.

References colorLink.

Referenced by MPerfWidget().

```
81                                      {
82     // Set all colors at once if Alt is held down
83     colorLink = !!(qApp->keyboardModifiers() & Qt::AltModifier);
84 }
```

Here is the caller graph for this function:

12.55.4.5 void MPerfWidget::enableChanged (int *index*) [private],[slot]

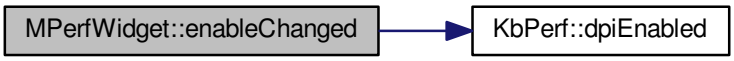
Definition at line 187 of file mperfwidget.cpp.

References KbPerf::dpiEnabled(), perf, and stages.

Referenced by MPerfWidget().

```
187                                         {
188     perf->dpiEnabled(index, stages[index].enableCheck->isChecked());
189 }
```

Here is the call graph for this function:



Here is the caller graph for this function:

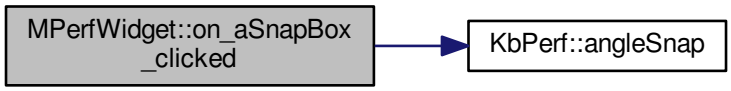
12.55.4.6 void MPerfWidget::on_aSnapBox_clicked (bool *checked*) [private], [slot]

Definition at line 201 of file mperfwidget.cpp.

References KbPerf::angleSnap(), and perf.

```
201                                                 {
202     perf->angleSnap(checked);
203 }
```

Here is the call graph for this function:



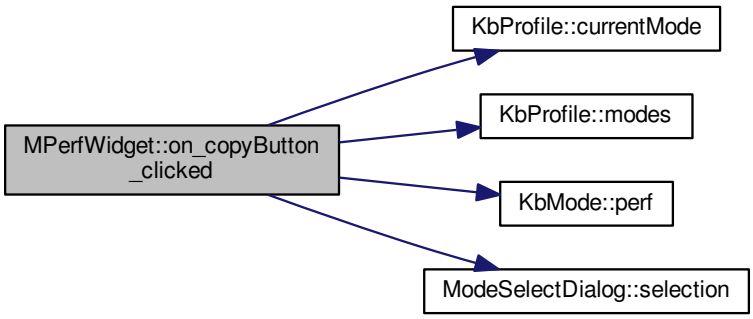
12.55.4.7 void MPerfWidget::on_copyButton_clicked ( ) [private], [slot]

Definition at line 209 of file mperfwidget.cpp.

References KbProfile::currentMode(), KbProfile::modes(), perf, KbMode::perf(), profile, and ModeSelectDialog::selection().

```
209                                     {
210     ModeSelectDialog dialog(this, profile->currentMode(),
      profile->modes(), "Copy performance settings to:");
211     if(dialog.exec() != QDialog::Accepted)
212         return;
213     QList<KbMode*> selectedModes = dialog.selection();
214     foreach(KbMode* mode, selectedModes){
215         *mode->perf() = *perf;
216     }
217 }
```

Here is the call graph for this function:



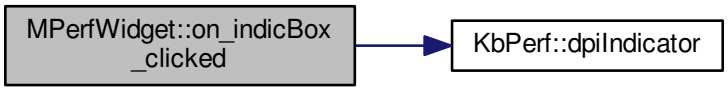
12.55.4.8 void MPerfWidget::on_indicBox_clicked (bool *checked*) [private], [slot]

Definition at line 197 of file mperfwidget.cpp.

References KbPerf::dpiIndicator(), and perf.

```
197                                                 {
198     perf->dpiIndicator(checked);
199 }
```

Here is the call graph for this function:



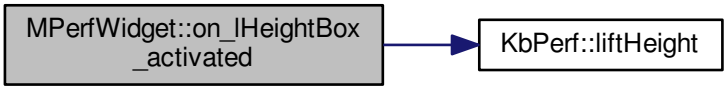
12.55.4.9 void MPerfWidget::on_lHeightBox_activated (int *index*) [private], [slot]

Definition at line 205 of file mperfwidget.cpp.

References KbPerf::liftHeight(), and perf.

```
205                                                 {
206     perf->liftHeight((KbPerf::height)(index + 1));
207 }
```

Here is the call graph for this function:



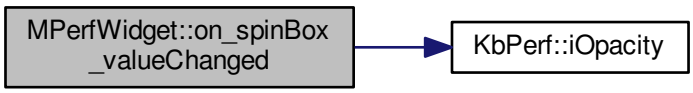
12.55.4.10 void MPerfWidget::on_spinBox_valueChanged (int *arg1*) [private], [slot]

Definition at line 219 of file mperfwidget.cpp.

References KbPerf::iOpacity(), and perf.

```
219                                                 {
220     if(!perf)
221         return;
222     perf->iOpacity(arg1 / 100.f);
223 }
```

Here is the call graph for this function:



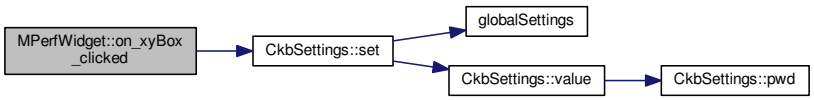
12.55.4.11 void MPerfWidget::on_xyBox_clicked (bool *checked*) [private], [slot]

Definition at line 191 of file mperfwidget.cpp.

References _xyLink, CkbSettings::set(), and xyLinkPath.

```
191                                                 {
192     _xyLink = !checked;
193     CkbSettings::set(xyLinkPath, checked);
194     // FIXME: update other MPerfWidgets, since this is a global setting
195 }
```

Here is the call graph for this function:



12.55.4.12 void MPerfWidget::setPerf (KbPerf * *newPerf,* KbProfile * *newProfile*)

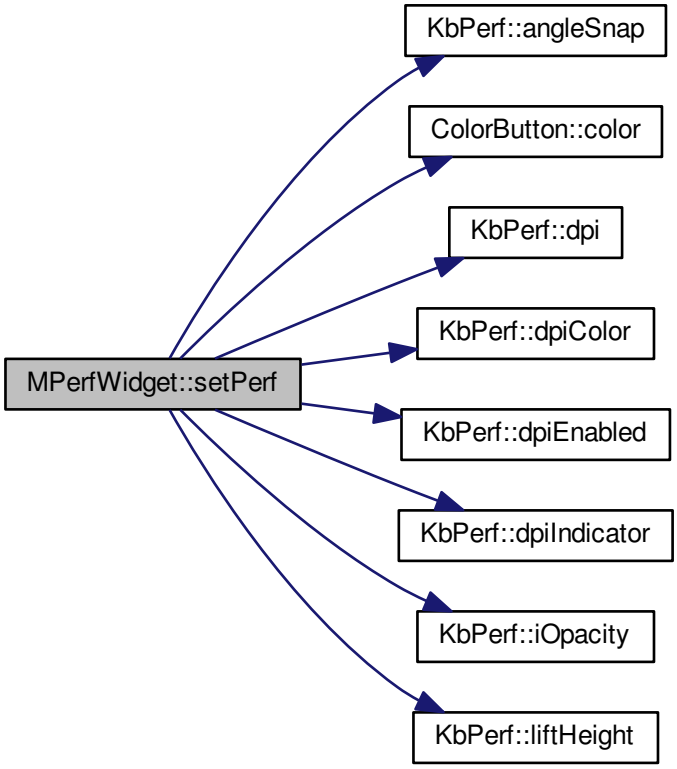
Definition at line 60 of file mperfwidget.cpp.

References _xyLink, KbPerf::angleSnap(), Ui_MPerfWidget::aSnapBox, ColorButton::color(), KbPerf::dpi(), DPI_COUNT, KbPerf::dpiColor(), KbPerf::dpiEnabled(), KbPerf::dpiIndicator(), MPerfWidget::DpiUi::enableCheck, Ui_MPerfWidget::iButtonO, MPerfWidget::DpiUi::indicator, Ui_MPerfWidget::indicBox, KbPerf::iOpacity(), Ui_MPerfWidget::lHeightBox, KbPerf::liftHeight(), KbPerf::OTHER, perf, profile, Ui_MPerfWidget::spinBox, stages, ui, MPerfWidget::DpiUi::xBox, and MPerfWidget::DpiUi::yBox.

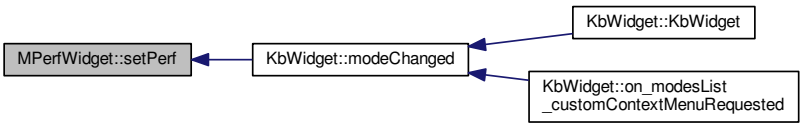
Referenced by KbWidget::modeChanged().

```
60                                                          {
61     perf = newPerf;
62     profile = newProfile;
63     ui->spinBox->setValue(round(perf->iOpacity() * 100.f));
64     for(int i = 0; i < DPI_COUNT; i++){
65         stages[i].indicator->color(perf->dpiColor(i));
66         bool oldLink = _xyLink;
67         // Don't force X/Y to the same value if they were set differently in the past
68         _xyLink = false;
69         stages[i].xBox->setValue(perf->dpi(i).x());
70         stages[i].yBox->setValue(perf->dpi(i).y());
71         _xyLink = oldLink;
72         if(stages[i].enableCheck)
73             stages[i].enableCheck->setChecked(perf->
      dpiEnabled(i));
74     }
75     ui->iButtonO->color(perf->dpiColor(KbPerf::OTHER));
76     ui->aSnapBox->setChecked(perf->angleSnap());
77     ui->lHeightBox->setCurrentIndex(perf->liftHeight() - 1);
78     ui->indicBox->setChecked(perf->dpiIndicator());
79 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.55.4.13 void MPerfWidget::sliderXMoved (int *index*) [private], [slot]

Definition at line 128 of file mperfwidget.cpp.

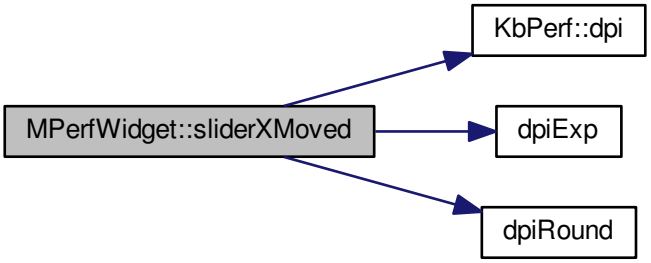
References _xyLink, KbPerf::dpi(), dpiExp(), dpiRound(), perf, SET_END, SET_START, stages, MPerfWidget::DpiUi::xBox, MPerfWidget::DpiUi::xSlider, and MPerfWidget::DpiUi::ySlider.

Referenced by MPerfWidget().

```
128                                    {
129     SET_START;
```

```
130     QSlider* slider = stages[index].xSlider;
131     int value = dpiRound(dpiExp(slider->value(), slider->minimum(), slider->maximum()));
132     stages[index].xBox->setValue(value);
133     if(_xyLink)
134         perf->dpi(index, value);
135     else
136         perf->dpi(index, QPoint(value, perf->dpi(index).y()));
137     SET_END;
138     if(_xyLink)
139         stages[index].ySlider->setValue(slider->value());
140 }
```

Here is the call graph for this function:



Here is the caller graph for this function:

12.55.4.14 void MPerfWidget::sliderYMoved (int *index*) [private],[slot]

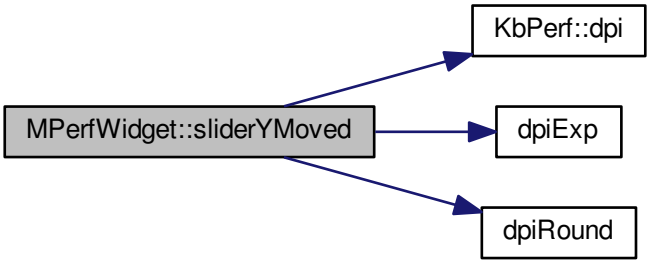
Definition at line 142 of file mperfwidget.cpp.

References _xyLink, KbPerf::dpi(), dpiExp(), dpiRound(), perf, SET_END, SET_START, stages, ui, Ui_MPerfWidget::xyBox, MPerfWidget::DpiUi::yBox, and MPerfWidget::DpiUi::ySlider.

Referenced by MPerfWidget().

```
142                                         {
143     SET_START;
144     QSlider* slider = stages[index].ySlider;
145     int value = dpiRound(dpiExp(slider->value(), slider->minimum(), slider->maximum()));
146     stages[index].yBox->setValue(value);
147     if(_xyLink)
148         perf->dpi(index, value);
149     else
150         perf->dpi(index, QPoint(perf->dpi(index).x(), value));
151     SET_END;
152     if(!ui->xyBox->isChecked())
153         // X/Y linked?
154         stages[index].xSlider->setValue(slider->value());
155 }
```

Here is the call graph for this function:



Here is the caller graph for this function:

12.55.5 Field Documentation

12.55.5.1 bool MPerfWidget::_xyLink [private]

Definition at line 44 of file mperfwidget.h.

Referenced by boxXChanged(), boxYChanged(), MPerfWidget(), on_xyBox_clicked(), setPerf(), sliderXMoved(), and sliderYMoved().

12.55.5.2 QSignalMapper MPerfWidget::boxXMapper [private]

Definition at line 49 of file mperfwidget.h.

Referenced by MPerfWidget().

12.55.5.3 QSignalMapper MPerfWidget::boxYMapper [private]

Definition at line 49 of file mperfwidget.h.

Referenced by MPerfWidget().

12.55.5.4 QSignalMapper MPerfWidget::buttonMapper1 [private]

Definition at line 47 of file mperfwidget.h.

Referenced by MPerfWidget().

12.55.5.5 QSignalMapper MPerfWidget::buttonMapper2 [private]

Definition at line 47 of file mperfwidget.h.

Referenced by MPerfWidget().

12.55.5.6 bool MPerfWidget::colorLink [private]

Definition at line 45 of file mperfwidget.h.

Referenced by colorChanged(), and colorClicked().

12.55.5.7 const int MPerfWidget::DPI_COUNT = KbPerf::DPI_COUNT [static]

Definition at line 28 of file mperfwidget.h.

Referenced by colorChanged(), MPerfWidget(), and setPerf().

12.55.5.8 const int MPerfWidget::DPI_MAX = KbPerf::DPI_MAX [static]

Definition at line 29 of file mperfwidget.h.

12.55.5.9 const int MPerfWidget::DPI_MIN = KbPerf::DPI_MIN [static]

Definition at line 29 of file mperfwidget.h.

12.55.5.10 QSignalMapper MPerfWidget::enableMapper [private]

Definition at line 50 of file mperfwidget.h.

Referenced by MPerfWidget().

12.55.5.11 bool MPerfWidget::isSetting [private]

Definition at line 53 of file mperfwidget.h.

12.55.5.12 KbPerf* MPerfWidget::perf [private]

Definition at line 34 of file mperfwidget.h.

Referenced by boxXChanged(), boxYChanged(), colorChanged(), enableChanged(), on_aSnapBox_clicked(), on_copyButton_clicked(), on_indicBox_clicked(), on_lHeightBox_activated(), on_spinBox_valueChanged(), setPerf(), sliderXMoved(), and sliderYMoved().

12.55.5.13 KbProfile* MPerfWidget::profile [private]

Definition at line 35 of file mperfwidget.h.

Referenced by on_copyButton_clicked(), and setPerf().

12.55.5.14 QSignalMapper MPerfWidget::sliderXMapper [private]

Definition at line 48 of file mperfwidget.h.

Referenced by MPerfWidget().

12.55.5.15 QSignalMapper MPerfWidget::sliderYMapper [private]

Definition at line 48 of file mperfwidget.h.

Referenced by MPerfWidget().

12.55.5.16 DpiUi MPerfWidget::stages[DPI_COUNT] [private]

Definition at line 43 of file mperfwidget.h.

Referenced by boxXChanged(), boxYChanged(), colorChanged(), enableChanged(), MPerfWidget(), setPerf(), sliderXMoved(), and sliderYMoved().

12.55.5.17 Ui::MPerfWidget* MPerfWidget::ui [private]

Definition at line 32 of file mperfwidget.h.

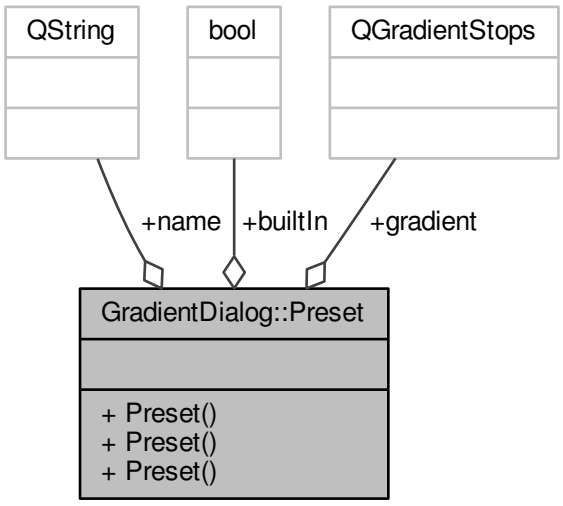
Referenced by colorChanged(), MPerfWidget(), setPerf(), sliderYMoved(), and ~MPerfWidget().

The documentation for this class was generated from the following files:

- src/ckb/mperfwidget.h
- src/ckb/mperfwidget.cpp

12.56 GradientDialog::Preset Struct Reference

Collaboration diagram for GradientDialog::Preset:



Public Member Functions

- Preset (QString _name, bool _builtIn=false)
- Preset (QString _name, const QGradientStops &_gradient)

- Preset ()

Data Fields

- QString name
- QGradientStops gradient
- bool builtIn

12.56.1 Detailed Description

Definition at line 23 of file gradientdialog.h.

12.56.2 Constructor & Destructor Documentation

12.56.2.1 GradientDialog::Preset::Preset (QString *_name,* bool *_builtIn* = false) [inline]

Definition at line 27 of file gradientdialog.h.

```
27 : name(_name), builtIn(_builtIn) {}
```

12.56.2.2 GradientDialog::Preset::Preset (QString *_name,* const QGradientStops & *_gradient*) [inline]

Definition at line 28 of file gradientdialog.h.

```
28 : name(_name), gradient(_gradient), builtIn(false) {}
```

12.56.2.3 GradientDialog::Preset::Preset ( ) [inline]

Definition at line 29 of file gradientdialog.h.

```
29 : builtIn(false) {}
```

12.56.3 Field Documentation

12.56.3.1 bool GradientDialog::Preset::builtIn

Definition at line 26 of file gradientdialog.h.

Referenced by GradientDialog::on_presetDelete_clicked(), GradientDialog::on_presetSave_clicked(), GradientDialog::setPreset(), and GradientDialog::~GradientDialog().

12.56.3.2 QGradientStops GradientDialog::Preset::gradient

Definition at line 25 of file gradientdialog.h.

Referenced by GradientDialog::GradientDialog(), GradientDialog::makeIcon(), GradientDialog::setPreset(), and GradientDialog::~GradientDialog().

12.56.3.3 QString GradientDialog::Preset::name

Definition at line 24 of file gradientdialog.h.

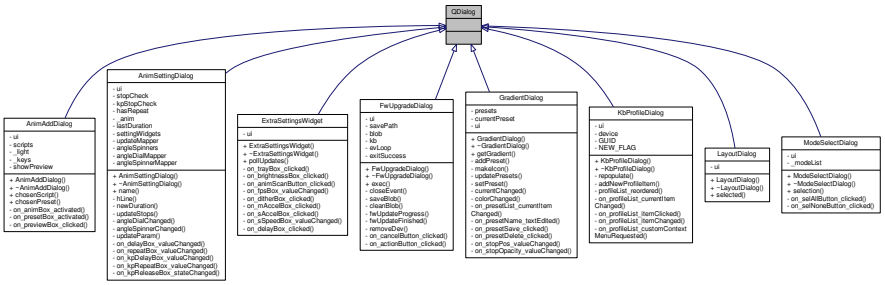
Referenced by GradientDialog::addPreset(), GradientDialog::on_presetDelete_clicked(), GradientDialog::on_presetSave_clicked(), GradientDialog::setPreset(), GradientDialog::updatePresets(), and GradientDialog::~GradientDialog().

The documentation for this struct was generated from the following file:

- src/ckb/gradientdialog.h

12.57 QDialog Class Reference

Inheritance diagram for QDialog:



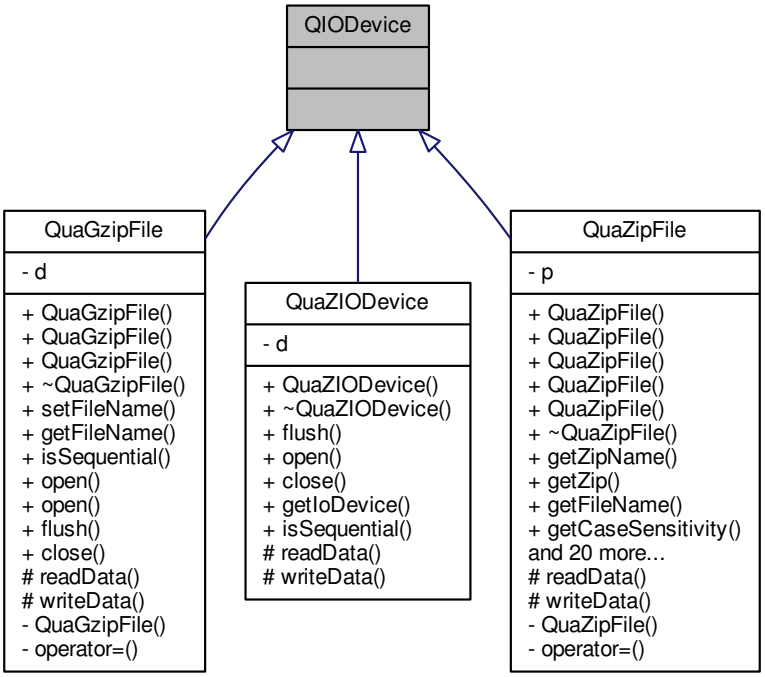
Collaboration diagram for QDialog:

The documentation for this class was generated from the following file:

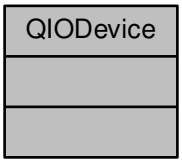
- src/ckb/extrasettingswidget.h

12.58 QIODevice Class Reference

Inheritance diagram for QIODevice:



Collaboration diagram for QIODevice:

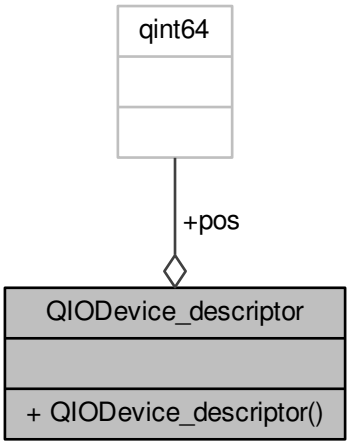


The documentation for this class was generated from the following file:

- src/ckb/quazip/quazipfile.h

12.59 QIODevice_descriptor Struct Reference

Collaboration diagram for QIODevice_descriptor:



Public Member Functions

- QIODevice_descriptor ()

Data Fields

- qint64 pos

12.59.1 Detailed Description

Definition at line 78 of file qioapi.cpp.

12.59.2 Constructor & Destructor Documentation

12.59.2.1 QIODevice_descriptor::QIODevice_descriptor ( ) [inline]

Definition at line 81 of file qioapi.cpp.

```
81                                  :
82          pos(0)
83      {}
```

12.59.3 Field Documentation

12.59.3.1 qint64 QIODevice_descriptor::pos

Definition at line 80 of file qioapi.cpp.

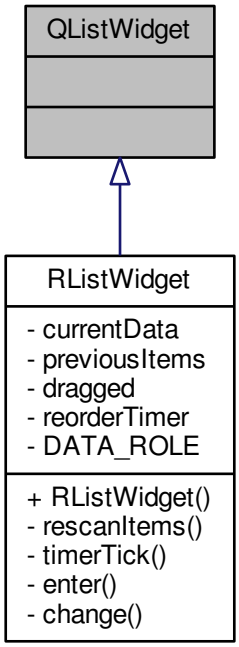
Referenced by qiodevice64_tell_file_func(), qiodevice_open_file_func(), qiodevice_read_file_func(), qiodevice_tell_file_func(), and qiodevice_write_file_func().

The documentation for this struct was generated from the following file:

- src/ckb/quazip/qioapi.cpp

12.60 QListWidget Class Reference

Inheritance diagram for QListWidget:



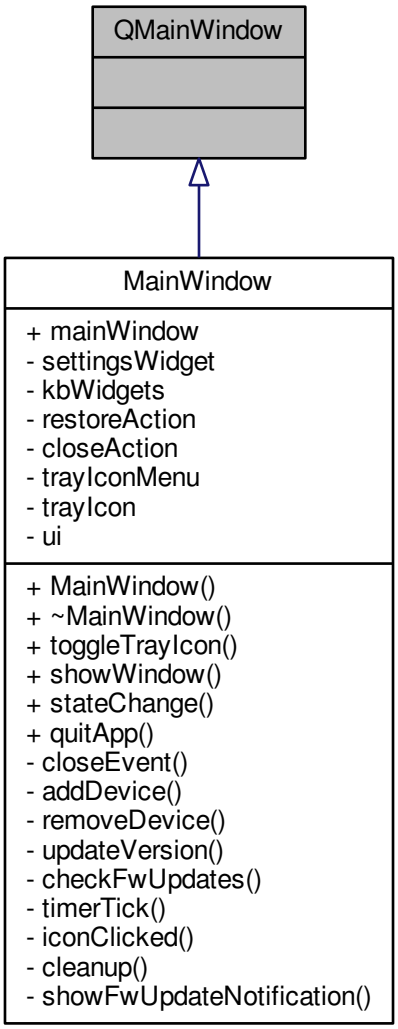
Collaboration diagram for QListWidget:

The documentation for this class was generated from the following file:

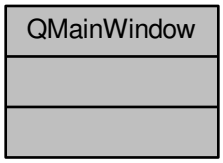
- src/ckb/rlistwidget.h

12.61 QMainWindow Class Reference

Inheritance diagram for QMainWindow:



Collaboration diagram for QMainWindow:

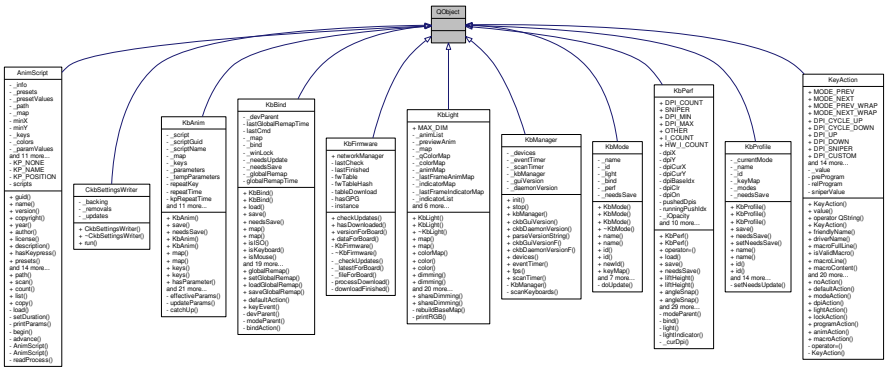


The documentation for this class was generated from the following file:

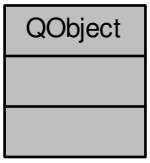
- src/ckb/mainwindow.h

12.62 QObject Class Reference

Inheritance diagram for QObject:



Collaboration diagram for QObject:

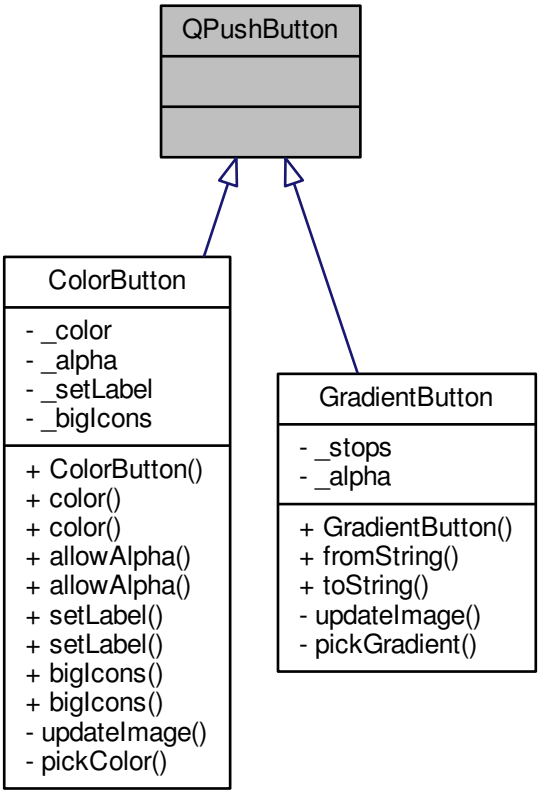


The documentation for this class was generated from the following file:

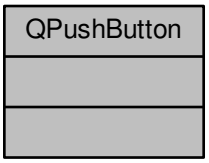
- src/ckb/kbprofile.h

12.63 QPushButton Class Reference

Inheritance diagram for QPushButton:



Collaboration diagram for QPushButton:

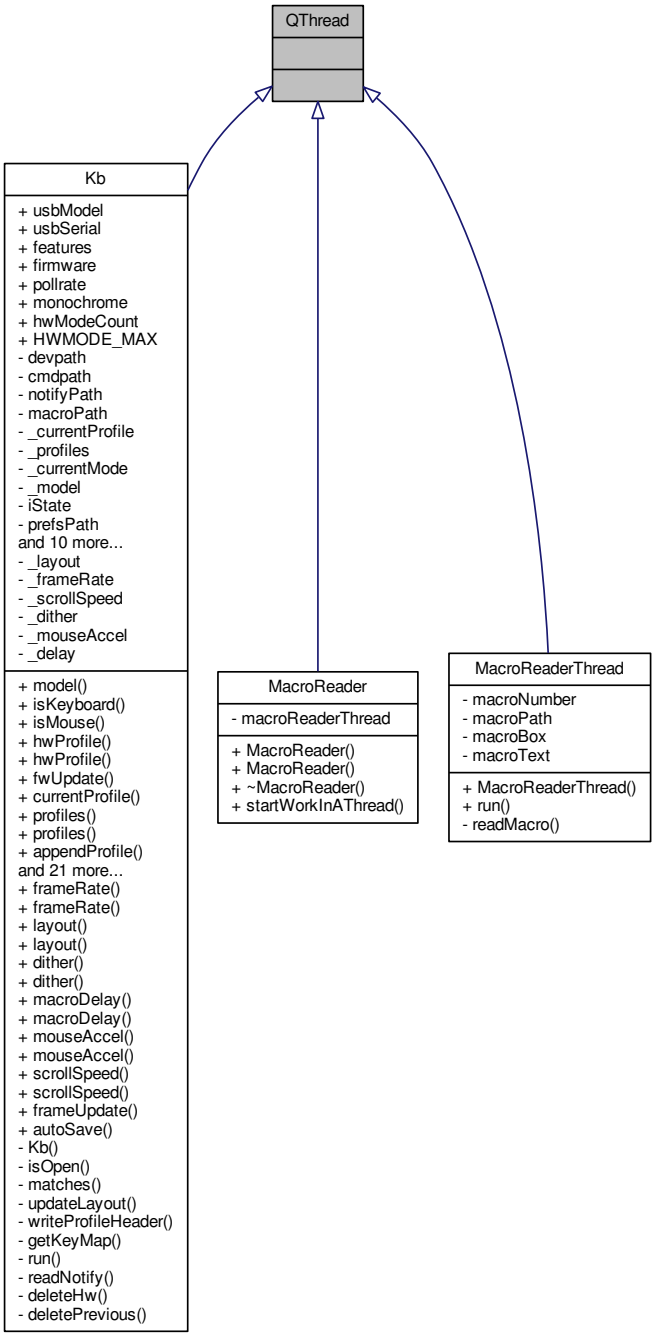


The documentation for this class was generated from the following file:

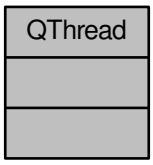
- src/ckb/colorbutton.h

12.64 QThread Class Reference

Inheritance diagram for QThread:



Collaboration diagram for QThread:



The documentation for this class was generated from the following file:

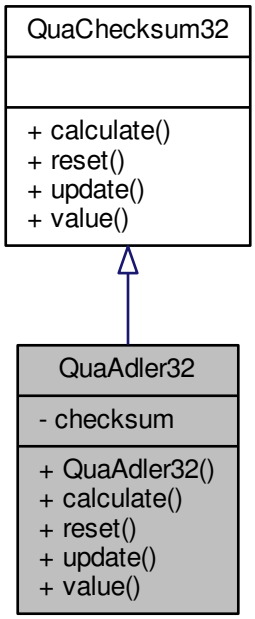
- src/ckb/macroreader.h

12.65 QuaAdler32 Class Reference

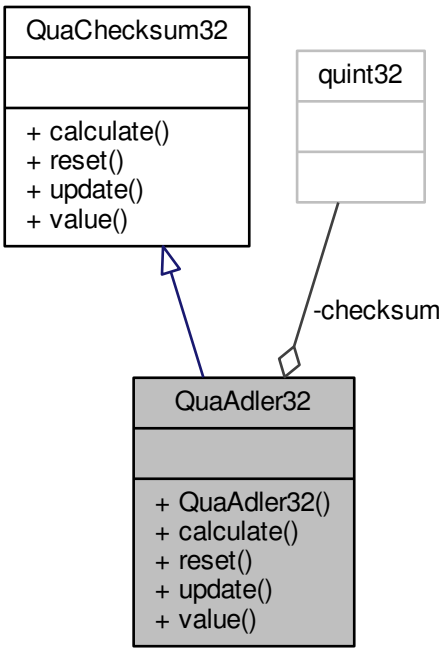
Adler32 checksum.

`#include <quazip/quaadler32.h>`

Inheritance diagram for QuaAdler32:



Collaboration diagram for QuaAdler32:



Public Member Functions

- QuaAdler32 ()
- quint32 calculate (const QByteArray &data)

  Calculates the checksum for data.
- void reset ()

  Resets the calculation on a checksun for a stream.
- void update (const QByteArray &buf)

  Updates the calculated checksum for the stream.
- quint32 value ()

  Value of the checksum calculated for the stream passed throw update().

Private Attributes

- quint32 checksum

12.65.1 Detailed Description

This class wrappers the adler32 function with the QuaChecksum32 interface. See QuaChecksum32 for more info.

Definition at line 38 of file quaadler32.h.

12.65.2 Constructor & Destructor Documentation

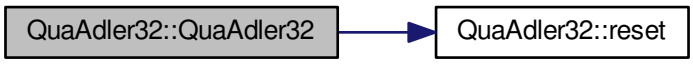
12.65.2.1 QuaAdler32::QuaAdler32 ( )

Definition at line 30 of file quaadler32.cpp.

References reset().

```
31 {
32     reset();
33 }
```

Here is the call graph for this function:



12.65.3 Member Function Documentation

12.65.3.1 quint32 QuaAdler32::calculate (const QByteArray & *data*) [virtual]

data source data

Returns

data checksum

This function has no efect on the value returned by value().

Implements QuaChecksum32.

Definition at line 35 of file quaadler32.cpp.

```
36 {
37     return adler32( adler32(0L, Z_NULL, 0), (const Bytef*)data.data(), data.size() );
38 }
```

12.65.3.2 void QuaAdler32::reset ( ) [virtual]

Implements QuaChecksum32.

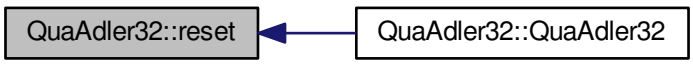
Definition at line 40 of file quaadler32.cpp.

References checksum.

Referenced by QuaAdler32().

```
41 {
42     checksum = adler32(0L, Z_NULL, 0);
43 }
```

Here is the caller graph for this function:



12.65.3.3 void QuaAdler32::update (const QByteArray & *buf*) [virtual]

buf next portion of data from the stream

Implements QuaChecksum32.

Definition at line 45 of file quaadler32.cpp.

References checksum.

```
46 {
47     checksum = adler32( checksum, (const Bytef*)buf.data(), buf.size() );
48 }
```

12.65.3.4 quint32 QuaAdler32::value ( ) [virtual]

Returns

checksum

Implements QuaChecksum32.

Definition at line 50 of file quaadler32.cpp.

References checksum.

```
51 {
52     return checksum;
53 }
```

12.65.4 Field Documentation

12.65.4.1 quint32 QuaAdler32::checksum [private]

Definition at line 51 of file quaadler32.h.

Referenced by reset(), update(), and value().

The documentation for this class was generated from the following files:

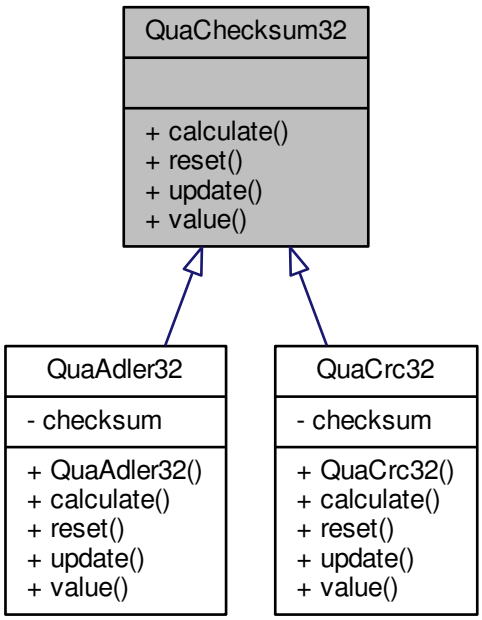
- src/ckb/quazip/quaadler32.h
- src/ckb/quazip/quaadler32.cpp

12.66 QuaChecksum32 Class Reference

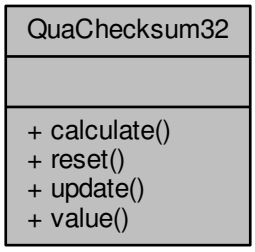
Checksum interface.

`#include <quazip/quachecksum32.h>`

Inheritance diagram for QuaChecksum32:



Collaboration diagram for QuaChecksum32:



Public Member Functions

- virtual quint32 calculate (const QByteArray &data)=0

  Calculates the checksum for data.
- virtual void reset ()=0

  Resets the calculation on a checksun for a stream.
- virtual void update (const QByteArray &buf)=0

Updates the calculated checksum for the stream.

- virtual quint32 value ()=0

  Value of the checksum calculated for the stream passed throw update().

12.66.1 Detailed Description

This is an interface for 32 bit checksums. Classes implementing this interface can calcunate a certin checksum in a single step:

```
QChecksum32 *crc32 = new QuaCrc32();
rasoult = crc32->calculate(data);
```

or by streaming the data:

```
QChecksum32 *crc32 = new QuaCrc32();
while(!fileA.atEnd())
    crc32->update(fileA.read(bufSize));
resoultA = crc32->value();
crc32->reset();
while(!fileB.atEnd())
    crc32->update(fileB.read(bufSize));
resoultB = crc32->value();
```

Definition at line 52 of file quachecksum32.h.

12.66.2 Member Function Documentation

12.66.2.1 virtual quint32 QuaChecksum32::calculate (const QByteArray & *data*) [pure virtual]

data source data

Returns

data checksum

This function has no efect on the value returned by value().

Implemented in QuaAdler32, and QuaCrc32.

12.66.2.2 virtual void QuaChecksum32::reset ( ) [pure virtual]

Implemented in QuaAdler32, and QuaCrc32.

12.66.2.3 virtual void QuaChecksum32::update (const QByteArray & *buf*) [pure virtual]

buf next portion of data from the stream

Implemented in QuaAdler32, and QuaCrc32.

12.66.2.4 virtual quint32 QuaChecksum32::value ( ) [pure virtual]

Returns

checksum

Implemented in QuaAdler32, and QuaCrc32.

The documentation for this class was generated from the following file:

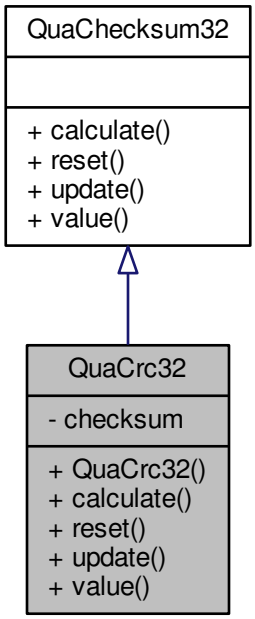
- src/ckb/quazip/quachecksum32.h

12.67 QuaCrc32 Class Reference

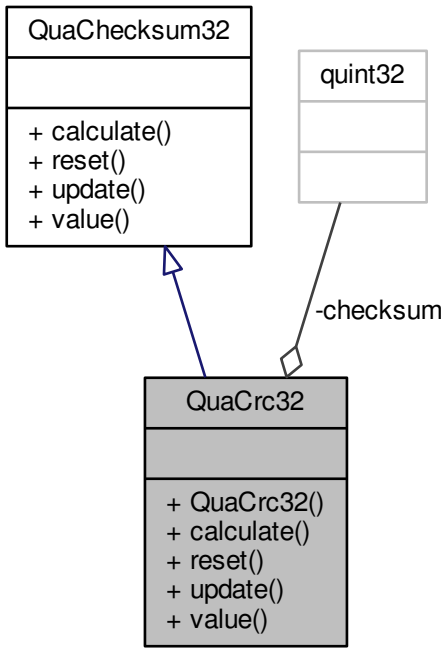
CRC32 checksum.

`#include <quazip/quacrc32.h>`

Inheritance diagram for QuaCrc32:



Collaboration diagram for QuaCrc32:



Public Member Functions

- QuaCrc32 ()
- quint32 calculate (const QByteArray &data)

  Calculates the checksum for data.
- void reset ()

  Resets the calculation on a checksun for a stream.
- void update (const QByteArray &buf)

  Updates the calculated checksum for the stream.
- quint32 value ()

  Value of the checksum calculated for the stream passed throw update().

Private Attributes

- quint32 checksum

12.67.1 Detailed Description

This class wrappers the crc32 function with the QuaChecksum32 interface. See QuaChecksum32 for more info.

Definition at line 35 of file quacrc32.h.

12.67.2 Constructor & Destructor Documentation

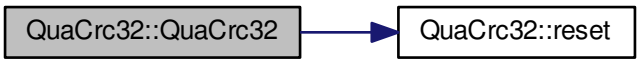
12.67.2.1 QuaCrc32::QuaCrc32 ( )

Definition at line 29 of file quacrc32.cpp.

References reset().

```
30 {
31     reset();
32 }
```

Here is the call graph for this function:



12.67.3 Member Function Documentation

12.67.3.1 quint32 QuaCrc32::calculate (const QByteArray & *data*) [virtual]

data source data

Returns

data checksum

This function has no efect on the value returned by value().

Implements QuaChecksum32.

Definition at line 34 of file quacrc32.cpp.

```
35 {
36     return crc32( crc32(0L, Z_NULL, 0), (const Bytef*)data.data(), data.size() );
37 }
```

12.67.3.2 void QuaCrc32::reset ( ) [virtual]

Implements QuaChecksum32.

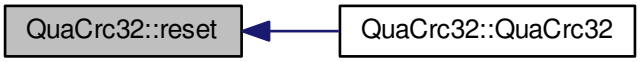
Definition at line 39 of file quacrc32.cpp.

References checksum.

Referenced by QuaCrc32().

```
40 {
41     checksum = crc32(0L, Z_NULL, 0);
42 }
```

Here is the caller graph for this function:



12.67.3.3 void QuaCrc32::update (const QByteArray & *buf*) [virtual]

buf next portion of data from the stream

Implements QuaChecksum32.

Definition at line 44 of file quacrc32.cpp.

References checksum.

```
45 {
46     checksum = crc32( checksum, (const Bytef*)buf.data(), buf.size() );
47 }
```

12.67.3.4 quint32 QuaCrc32::value ( ) [virtual]

Returns

checksum

Implements QuaChecksum32.

Definition at line 49 of file quacrc32.cpp.

References checksum.

```
50 {
51     return checksum;
52 }
```

12.67.4 Field Documentation

12.67.4.1 quint32 QuaCrc32::checksum [private]

Definition at line 47 of file quacrc32.h.

Referenced by reset(), update(), and value().

The documentation for this class was generated from the following files:

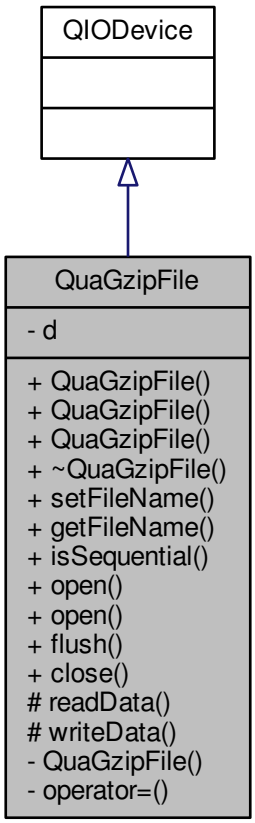
- src/ckb/quazip/quacrc32.h
- src/ckb/quazip/quacrc32.cpp

12.68 QuaGzipFile Class Reference

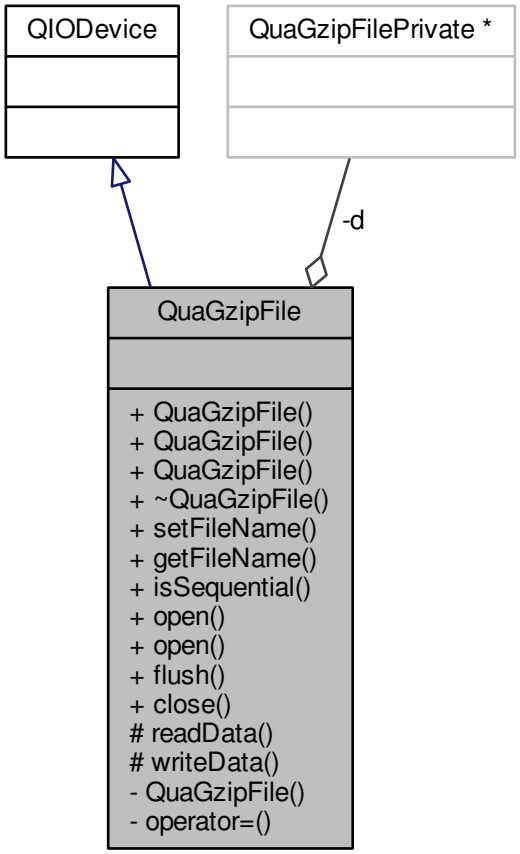
GZIP file.

`#include <src/ckb/quazip/quagzipfile.h>`

Inheritance diagram for QuaGzipFile:



Collaboration diagram for QuaGzipFile:



Public Member Functions

- QuaGzipFile ()

  Empty constructor.
- QuaGzipFile (QObject *parent)

  Empty constructor with a parent.
- QuaGzipFile (const QString &fileName, QObject *parent=NULL)

  Constructor.
- virtual ~QuaGzipFile ()

  Destructor.
- void setFileName (const QString &fileName)

  Sets the name of the GZIP file to be opened.
- QString getFileName () const

  Returns the name of the GZIP file.
- virtual bool isSequential () const

  Returns true.
- virtual bool open (QIODevice::OpenMode mode)

  Opens the file.

- virtual bool open (int fd, QIODevice::OpenMode mode)

  Opens the file.
- virtual bool flush ()

  Flushes data to file.
- virtual void close ()

  Closes the file.

Protected Member Functions

- virtual qint64 readData (char *data, qint64 maxSize)

  Implementation of QIODevice::readData().
- virtual qint64 writeData (const char *data, qint64 maxSize)

  Implementation of QIODevice::writeData().

Private Member Functions

- QuaGzipFile (const QuaGzipFile &that)
- QuaGzipFile & operator= (const QuaGzipFile &that)

Private Attributes

- QuaGzipFilePrivate * d

12.68.1 Detailed Description

This class is a wrapper around GZIP file access functions in zlib. Unlike QuaZip classes, it doesn't allow reading from a GZIP file opened as QIODevice, for example, if your GZIP file is in QBuffer. It only provides QIODevice access to a GZIP file contents, but the GZIP file itself must be identified by its name on disk or by descriptor id.

Definition at line 39 of file quagzipfile.h.

12.68.2 Constructor & Destructor Documentation

12.68.2.1 QuaGzipFile::QuaGzipFile ( )

Must call setFileName() before trying to open.

Definition at line 87 of file quagzipfile.cpp.

```
87                          :
88 d(new QuaGzipFilePrivate())
89 {
90 }
```

12.68.2.2 QuaGzipFile::QuaGzipFile (QObject * *parent*)

Must call setFileName() before trying to open.

Parameters

parent	The parent object, as per QObject logic.

Definition at line 92 of file quagzipfile.cpp.

```
92                                       :
93 QIODevice(parent),
94 d(new QuaGzipFilePrivate())
95 {
96 }
```

12.68.2.3 QuaGzipFile::QuaGzipFile (const QString & *fileName*, QObject * *parent* = NULL)

Parameters

fileName	The name of the GZIP file.
parent	The parent object, as per QObject logic.

Definition at line 98 of file quagzipfile.cpp.

```
98                                                                  :
99   QIODevice(parent),
100 d(new QuaGzipFilePrivate(fileName))
101 {
102 }
```

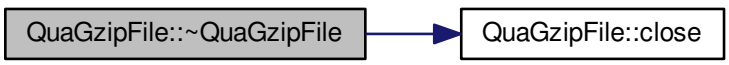
12.68.2.4 QuaGzipFile::~QuaGzipFile ( ) [virtual]

Definition at line 104 of file quagzipfile.cpp.

References close(), and d.

```
105 {
106   if (isOpen()) {
107     close();
108   }
109   delete d;
110 }
```

Here is the call graph for this function:



12.68.2.5 QuaGzipFile::QuaGzipFile (const QuaGzipFile & *that*) [private]

12.68.3 Member Function Documentation

12.68.3.1 void QuaGzipFile::close ( ) [virtual]

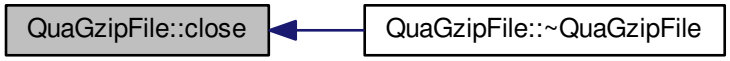
Definition at line 152 of file quagzipfile.cpp.

References d.

Referenced by ~QuaGzipFile().

```
153 {
154   QIODevice::close();
155   gzclose(d->gzd);
156 }
```

Here is the caller graph for this function:



12.68.3.2 bool QuaGzipFile::flush ( ) [virtual]

The data is written using Z_SYNC_FLUSH mode. Doesn't make any sense when reading.

Definition at line 147 of file quagzipfile.cpp.

References d.

```
148 {
149     return gzflush(d->gzd, Z_SYNC_FLUSH) == Z_OK;
150 }
```

12.68.3.3 QString QuaGzipFile::getFileName ( ) const

Definition at line 117 of file quagzipfile.cpp.

References d.

```
118 {
119     return d->fileName;
120 }
```

12.68.3.4 bool QuaGzipFile::isSequential ( ) const [virtual]

Strictly speaking, zlib supports seeking for GZIP files, but it is poorly implemented, because there is no way to implement it properly. For reading, seeking backwards is very slow, and for writing, it is downright impossible. Therefore, QuaGzipFile does not support seeking at all.

Definition at line 122 of file quagzipfile.cpp.

```
123 {
124   return true;
125 }
```

12.68.3.5 bool QuaGzipFile::open (QIODevice::OpenMode *mode*) [virtual]

Parameters

mode	Can be either QIODevice::Write or QIODevice::Read. ReadWrite and Append aren't supported.

Definition at line 127 of file quagzipfile.cpp.

References d.

```
128 {
129     QString error;
130     if (!d->open(d->fileName, mode, error)) {
131         setErrorString(error);
132         return false;
133     }
134     return QIODevice::open(mode);
135 }
```

12.68.3.6 bool QuaGzipFile::open (int *fd*, QIODevice::OpenMode *mode*) [virtual]

This is an overloaded member function, provided for convenience. It differs from the above function only in what argument(s) it accepts.

Parameters

fd	The file descriptor to read/write the GZIP file from/to.
mode	Can be either QIODevice::Write or QIODevice::Read. ReadWrite and Append aren't supported.

Definition at line 137 of file quagzipfile.cpp.

References d.

```
138 {
139     QString error;
140     if (!d->open(fd, mode, error)) {
141         setErrorString(error);
142         return false;
143     }
144     return QIODevice::open(mode);
145 }
```

12.68.3.7 QuaGzipFile& QuaGzipFile::operator= (const QuaGzipFile & *that*) [private]

12.68.3.8 qint64 QuaGzipFile::readData (char * *data*, qint64 *maxSize*) [protected], [virtual]

Definition at line 158 of file quagzipfile.cpp.

References d.

```
159 {
160     return gzread(d->gzd, (voidp)data, (unsigned)maxSize);
161 }
```

12.68.3.9 void QuaGzipFile::setFileName (const QString & *fileName*)

Definition at line 112 of file quagzipfile.cpp.

References d.

```
113 {
114     d->fileName = fileName;
115 }
```

12.68.3.10 qint64 QuaGzipFile::writeData (const char * *data*, qint64 *maxSize*) [protected], [virtual]

Definition at line 163 of file quagzipfile.cpp.

References d.

```
164 {
165     if (maxSize == 0)
166         return 0;
167     int written = gzwrite(d->gzd, (voidp)data, (unsigned)maxSize);
168     if (written == 0)
169         return -1;
170     else
171         return written;
172 }
```

12.68.4 Field Documentation

12.68.4.1 QuaGzipFilePrivate* QuaGzipFile::d [private]

Definition at line 105 of file quagzipfile.h.

Referenced by close(), flush(), getFileName(), open(), readData(), setFileName(), writeData(), and ~QuaGzipFile().

The documentation for this class was generated from the following files:

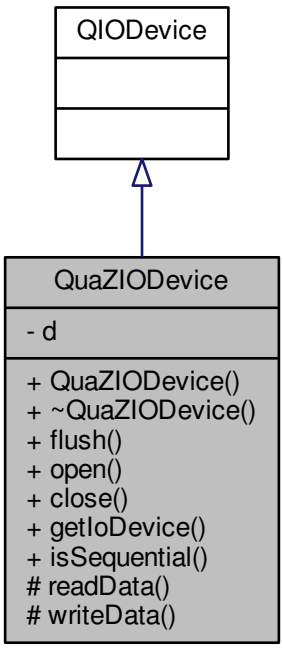
- src/ckb/quazip/quagzipfile.h
- src/ckb/quazip/quagzipfile.cpp

12.69 QuaZIODevice Class Reference

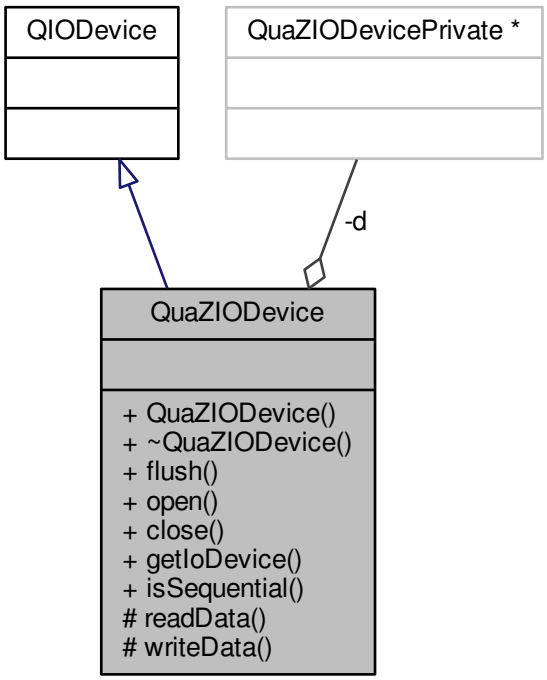
A class to compress/decompress QIODevice.

`#include <src/ckb/quazip/quaziodevice.h>`

Inheritance diagram for QuaZIODevice:



Collaboration diagram for QuaZIODevice:



Public Member Functions

- QuaZIODevice (QIODevice ∗io, QObject ∗parent=NULL)

  Constructor.
- ∼QuaZIODevice ()

  Destructor.
- virtual bool flush ()

  Flushes data waiting to be written.
- virtual bool open (QIODevice::OpenMode mode)

  Opens the device.
- virtual void close ()

  Closes this device, but not the underlying one.
- QIODevice ∗ getIoDevice () const

  Returns the underlying device.
- virtual bool isSequential () const

  Returns true.

Protected Member Functions

- virtual qint64 readData (char ∗data, qint64 maxSize)

  Implementation of QIODevice::readData().
- virtual qint64 writeData (const char ∗data, qint64 maxSize)

  Implementation of QIODevice::writeData().

Private Attributes

- QuaZIODevicePrivate * d

12.69.1 Detailed Description

This class can be used to compress any data written to QIODevice or decompress it back. Compressing data sent over a QTcpSocket is a good example.

Definition at line 41 of file quaziodevice.h.

12.69.2 Constructor & Destructor Documentation

12.69.2.1 QuaZIODevice::QuaZIODevice (QIODevice * *io*, QObject * *parent* = NULL)

Parameters

io	The QIODevice to read/write.
parent	The parent object, as per QObject logic.

Definition at line 123 of file quaziodevice.cpp.

```
123                                                     :
124     QIODevice(parent),
125     d(new QuaZIODevicePrivate(io))
126 {
127   connect(io, SIGNAL(readyRead()), SIGNAL(readyRead()));
128 }
```

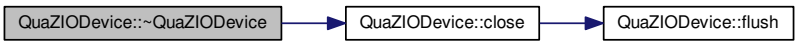
12.69.2.2 QuaZIODevice::~QuaZIODevice ( )

Definition at line 130 of file quaziodevice.cpp.

References close(), and d.

```
131 {
132     if (isOpen())
133         close();
134     delete d;
135 }
```

Here is the call graph for this function:



12.69.3 Member Function Documentation

12.69.3.1 void QuaZIODevice::close ( ) [virtual]

The underlying QIODevice is not closed in case you want to write something else to it.

Definition at line 169 of file quaziodevice.cpp.

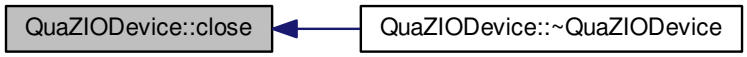
References d, and flush().

Referenced by ~QuaZIODevice().

```
170 {
171     if ((openMode() & QIODevice::ReadOnly) != 0) {
172         if (inflateEnd(&d->zins) != Z_OK) {
173             setErrorString(d->zins.msg);
174         }
175     }
176     if ((openMode() & QIODevice::WriteOnly) != 0) {
177         flush();
178         if (deflateEnd(&d->zouts) != Z_OK) {
179             setErrorString(d->zouts.msg);
180         }
181     }
182     QIODevice::close();
183 }
```

Here is the call graph for this function:

Here is the caller graph for this function:



12.69.3.2 bool QuaZIODevice::flush ( ) [virtual]

Unfortunately, as QIODevice doesn't support flush() by itself, the only thing this method does is write the compressed data into the device using Z_SYNC_FLUSH mode. If you need the compressed data to actually be flushed from the buffer of the underlying QIODevice, you need to call its flush() method as well, providing it supports it (like QTcpSocket does). Example:

```
QuaZIODevice dev(&sock);
dev.open(QIODevice::Write);
dev.write(yourDataGoesHere);
dev.flush();
sock->flush(); // this actually sends data to network
```

This may change in the future versions of QuaZIP by implementing an ugly hack: trying to cast the QIODevice using qobject_cast to known flush()-supporting subclasses, and calling flush if the resulting pointer is not zero.

Definition at line 281 of file quaziodevice.cpp.

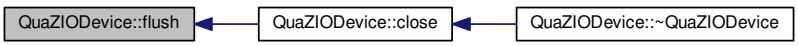
References d, and QUAZIO_OUTBUFSIZE.

Referenced by close().

```
282 {
283     QString error;
284     if (d->doFlush(error) < 0) {
```

```
285         setErrorString(error);
286         return false;
287     }
288     // can't flush buffer, some data is still waiting
289     if (d->outBufPos < d->outBufSize)
290         return true;
291     Bytef c = 0;
292     d->zouts.next_in = &c; // fake input buffer
293     d->zouts.avail_in = 0; // of zero size
294     do {
295         d->zouts.next_out = (Bytef *) d->outBuf;
296         d->zouts.avail_out = QUAZIO_OUTBUFSIZE;
297         switch (deflate(&d->zouts, Z_SYNC_FLUSH)) {
298         case Z_OK:
299           d->outBufSize = (char *) d->zouts.next_out - d->outBuf;
300           if (d->doFlush(error) < 0) {
301               setErrorString(error);
302               return false;
303           }
304           if (d->outBufPos < d->outBufSize)
305               return true;
306           break;
307         case Z_BUF_ERROR: // nothing to write?
308           return true;
309         default:
310           setErrorString(QString::fromLocal8Bit(d->zouts.msg));
311           return false;
312         }
313     } while (d->zouts.avail_out == 0);
314     return true;
315 }
```

Here is the caller graph for this function:



12.69.3.3 QIODevice * QuaZIODevice::getIoDevice ( ) const

Definition at line 137 of file quaziodevice.cpp.

References d.

```
138 {
139     return d->io;
140 }
```

12.69.3.4 bool QuaZIODevice::isSequential ( ) const [virtual]

Definition at line 317 of file quaziodevice.cpp.

```
318 {
319   return true;
320 }
```

12.69.3.5 bool QuaZIODevice::open (QIODevice::OpenMode *mode*) [virtual]

Parameters

mode	Neither QIODevice::ReadWrite nor QIODevice::Append are not supported.

Definition at line 142 of file quaziodevice.cpp.

References d.

```
143 {
144     if ((mode & QIODevice::Append) != 0) {
145         setErrorString(trUtf8("QIODevice::Append is not supported for"
146                     " QuaZIODevice"));
147         return false;
148     }
149     if ((mode & QIODevice::ReadWrite) == QIODevice::ReadWrite) {
150         setErrorString(trUtf8("QIODevice::ReadWrite is not supported for"
151                     " QuaZIODevice"));
152         return false;
153     }
154     if ((mode & QIODevice::ReadOnly) != 0) {
155         if (inflateInit(&d->zins) != Z_OK) {
156             setErrorString(d->zins.msg);
157             return false;
158         }
159     }
160     if ((mode & QIODevice::WriteOnly) != 0) {
161         if (deflateInit(&d->zouts, Z_DEFAULT_COMPRESSION) != Z_OK) {
162             setErrorString(d->zouts.msg);
163             return false;
164         }
165     }
166     return QIODevice::open(mode);
167 }
```

12.69.3.6 qint64 QuaZIODevice::readData (char * *data*, qint64 *maxSize*) [protected], [virtual]

Definition at line 185 of file quaziodevice.cpp.

References d, and QUAZIO_INBUFSIZE.

```
186 {
187   int read = 0;
188   while (read < maxSize) {
189     if (d->inBufPos == d->inBufSize) {
190       d->inBufPos = 0;
191       d->inBufSize = d->io->read(d->inBuf, QUAZIO_INBUFSIZE);
192       if (d->inBufSize == -1) {
193         d->inBufSize = 0;
194         setErrorString(d->io->errorString());
195         return -1;
196       }
197       if (d->inBufSize == 0)
198         break;
199     }
200     while (read < maxSize && d->inBufPos < d->inBufSize) {
201       d->zins.next_in = (Bytef *) (d->inBuf + d->inBufPos);
202       d->zins.avail_in = d->inBufSize - d->inBufPos;
203       d->zins.next_out = (Bytef *) (data + read);
204       d->zins.avail_out = (uInt) (maxSize - read); // hope it's less than 2GB
205       int more = 0;
206       switch (inflate(&d->zins, Z_SYNC_FLUSH)) {
207       case Z_OK:
208         read = (char *) d->zins.next_out - data;
209         d->inBufPos = (char *) d->zins.next_in - d->inBuf;
210         break;
211       case Z_STREAM_END:
212         read = (char *) d->zins.next_out - data;
213         d->inBufPos = (char *) d->zins.next_in - d->inBuf;
214         return read;
215       case Z_BUF_ERROR: // this should never happen, but just in case
216         if (!d->zBufError) {
217           qWarning("Z_BUF_ERROR detected with %d/%d in/out, weird",
218               d->zins.avail_in, d->zins.avail_out);
219           d->zBufError = true;
220         }
221         memmove(d->inBuf, d->inBuf + d->inBufPos, d->inBufSize - d->inBufPos);
222         d->inBufSize -= d->inBufPos;
223         d->inBufPos = 0;
224         more = d->io->read(d->inBuf + d->inBufSize, QUAZIO_INBUFSIZE -
      d->inBufSize);
```

```
225         if (more == -1) {
226           setErrorString(d->io->errorString());
227           return -1;
228         }
229         if (more == 0)
230           return read;
231         d->inBufSize += more;
232         break;
233       default:
234         setErrorString(QString::fromLocal8Bit(d->zins.msg));
235         return -1;
236       }
237     }
238   }
239 #ifdef QUAZIP_ZIODEVICE_DEBUG_INPUT
240   indebug.write(data, read);
241 #endif
242   return read;
243 }
```

12.69.3.7 qint64 QuaZIODevice::writeData (const char * *data*, qint64 *maxSize*) [protected], [virtual]

Definition at line 245 of file quaziodevice.cpp.

References d, and QUAZIO_OUTBUFSIZE.

```
246 {
247   int written = 0;
248   QString error;
249   if (d->doFlush(error) == -1) {
250     setErrorString(error);
251     return -1;
252   }
253   while (written < maxSize) {
254       // there is some data waiting in the output buffer
255     if (d->outBufPos < d->outBufSize)
256       return written;
257     d->zouts.next_in = (Bytef *) (data + written);
258     d->zouts.avail_in = (uInt) (maxSize - written); // hope it's less than 2GB
259     d->zouts.next_out = (Bytef *) d->outBuf;
260     d->zouts.avail_out = QUAZIO_OUTBUFSIZE;
261     switch (deflate(&d->zouts, Z_NO_FLUSH)) {
262     case Z_OK:
263       written = (char *) d->zouts.next_in - data;
264       d->outBufSize = (char *) d->zouts.next_out - d->outBuf;
265       break;
266     default:
267       setErrorString(QString::fromLocal8Bit(d->zouts.msg));
268       return -1;
269     }
270     if (d->doFlush(error) == -1) {
271       setErrorString(error);
272       return -1;
273     }
274   }
275 #ifdef QUAZIP_ZIODEVICE_DEBUG_OUTPUT
276   debug.write(data, written);
277 #endif
278   return written;
279 }
```

12.69.4 Field Documentation

12.69.4.1 QuaZIODevicePrivate* QuaZIODevice::d [private]

Definition at line 96 of file quaziodevice.h.

Referenced by close(), flush(), getIoDevice(), open(), readData(), writeData(), and ~QuaZIODevice().

The documentation for this class was generated from the following files:

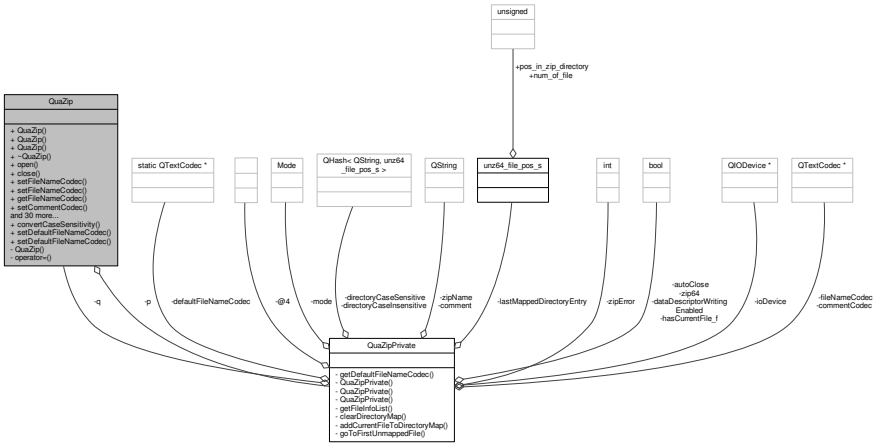
- src/ckb/quazip/quaziodevice.h
- src/ckb/quazip/quaziodevice.cpp

12.70 QuaZip Class Reference

ZIP archive.

`#include <quazip/quazip.h>`

Collaboration diagram for QuaZip:



Public Types

- enum Constants { MAX_FILE_NAME_LENGTH =256 }

  Useful constants.
- enum Mode {
  mdNotOpen, mdUnzip, mdCreate, mdAppend,
  mdAdd }

  Open mode of the ZIP file.
- enum CaseSensitivity { csDefault =0, csSensitive =1, csInsensitive =2 }

  Case sensitivity for the file names.

Public Member Functions

- QuaZip ()

  Constructs QuaZip object.
- QuaZip (const QString &zipName)

  Constructs QuaZip object associated with ZIP file zipName.
- QuaZip (QIODevice *ioDevice)

  Constructs QuaZip object associated with ZIP file represented by ioDevice.
- ~QuaZip ()

  Destroys QuaZip object.
- bool open (Mode mode, zlib_filefunc_def *ioApi=NULL)

  Opens ZIP file.
- void close ()

  Closes ZIP file.
- void setFileNameCodec (QTextCodec *fileNameCodec)

  Sets the codec used to encode/decode file names inside archive.
- void setFileNameCodec (const char *fileNameCodecName)

Sets the codec used to encode/decode file names inside archive.

- QTextCodec * getFileNameCodec () const

  Returns the codec used to encode/decode comments inside archive.
- void setCommentCodec (QTextCodec *commentCodec)

  Sets the codec used to encode/decode comments inside archive.
- void setCommentCodec (const char *commentCodecName)

  Sets the codec used to encode/decode comments inside archive.
- QTextCodec * getCommentCodec () const

  Returns the codec used to encode/decode comments inside archive.
- QString getZipName () const

  Returns the name of the ZIP file.
- void setZipName (const QString &zipName)

  Sets the name of the ZIP file.
- QIODevice * getIoDevice () const

  Returns the device representing this ZIP file.
- void setIoDevice (QIODevice *ioDevice)

  Sets the device representing the ZIP file.
- Mode getMode () const

  Returns the mode in which ZIP file was opened.
- bool isOpen () const

  Returns `true` if ZIP file is open, `false` otherwise.
- int getZipError () const

  Returns the error code of the last operation.
- int getEntriesCount () const

  Returns number of the entries in the ZIP central directory.
- QString getComment () const

  Returns global comment in the ZIP file.
- void setComment (const QString &comment)

  Sets the global comment in the ZIP file.
- bool goToFirstFile ()

  Sets the current file to the first file in the archive.
- bool goToNextFile ()

  Sets the current file to the next file in the archive.
- bool setCurrentFile (const QString &fileName, CaseSensitivity cs=csDefault)

  Sets current file by its name.
- bool hasCurrentFile () const

  Returns `true` if the current file has been set.
- bool getCurrentFileInfo (QuaZipFileInfo *info) const

  Retrieves information about the current file.
- bool getCurrentFileInfo (QuaZipFileInfo64 *info) const

  Retrieves information about the current file.
- QString getCurrentFileName () const

  Returns the current file name.
- unzFile getUnzFile ()

  Returns `unzFile` handle.
- zipFile getZipFile ()

  Returns `zipFile` handle.
- void setDataDescriptorWritingEnabled (bool enabled)

  Changes the data descriptor writing mode.
- bool isDataDescriptorWritingEnabled () const

  Returns the data descriptor default writing mode.

- QStringList getFileNameList () const

  Returns a list of files inside the archive.
- QList< QuaZipFileInfo > getFileInfoList () const

  Returns information list about all files inside the archive.
- QList< QuaZipFileInfo64 > getFileInfoList64 () const

  Returns information list about all files inside the archive.
- void setZip64Enabled (bool zip64)

  Enables the zip64 mode.
- bool isZip64Enabled () const

  Returns whether the zip64 mode is enabled.
- bool isAutoClose () const

  Returns the auto-close flag.
- void setAutoClose (bool autoClose) const

  Sets or unsets the auto-close flag.

Static Public Member Functions

- static Qt::CaseSensitivity convertCaseSensitivity (CaseSensitivity cs)

  Returns the actual case sensitivity for the specified QuaZIP one.
- static void setDefaultFileNameCodec (QTextCodec *codec)

  Sets the default file name codec to use.
- static void setDefaultFileNameCodec (const char *codecName)

  This is an overloaded member function, provided for convenience. It differs from the above function only in what argument(s) it accepts. Equivalent to calling setDefltFileNameCodec(QTextCodec::codecForName(codecName)).

Private Member Functions

- QuaZip (const QuaZip &that)
- QuaZip & operator= (const QuaZip &that)

Private Attributes

- QuaZipPrivate * p

Friends

- class QuaZipPrivate

12.70.1 Detailed Description

This class implements basic interface to the ZIP archive. It can be used to read table contents of the ZIP archive and retreiving information about the files inside it.

You can also use this class to open files inside archive by passing pointer to the instance of this class to the constructor of the QuaZipFile class. But see QuaZipFile::QuaZipFile(QuaZip*, QObject*) for the possible pitfalls.

This class is indended to provide interface to the ZIP subpackage of the ZIP/UNZIP package as well as to the UNZIP subpackage. But currently it supports only UNZIP.

The use of this class is simple - just create instance using constructor, then set ZIP archive file name using setFile() function (if you did not passed the name to the constructor), then open() and then use different functions to work with it! Well, if you are paranoid, you may also wish to call close before destructing the instance, to check for errors on close.

You may also use getUnzFile() and getZipFile() functions to get the ZIP archive handle and use it with ZIP/UNZIP package API directly.

This class supports localized file names inside ZIP archive, but you have to set up proper codec with setCodec() function. By default, locale codec will be used, which is probably ok for UNIX systems, but will almost certainly fail with ZIP archives created in Windows. This is because Windows ZIP programs have strange habit of using DOS encoding for file names in ZIP archives. For example, ZIP archive with cyrillic names created in Windows will have file names in `IBM866` encoding instead of `WINDOWS-1251`. I think that calling one function is not much trouble, but for true platform independency it would be nice to have some mechanism for file name encoding auto detection using locale information. Does anyone know a good way to do it?

Definition at line 84 of file quazip.h.

12.70.2 Member Enumeration Documentation

12.70.2.1 enum QuaZip::CaseSensitivity

This is what you specify when accessing files in the archive. Works perfectly fine with any characters thanks to Qt's great unicode support. This is different from ZIP/UNZIP API, where only US-ASCII characters was supported.

Enumerator

csDefault Default for platform. Case sensitive for UNIX, not for Windows.

csSensitive Case sensitive.

csInsensitive Case insensitive.

Definition at line 114 of file quazip.h.

```
114                         {
115       csDefault=0,
116       csSensitive=1,
117       csInsensitive=2
118     };
```

12.70.2.2 enum QuaZip::Constants

Enumerator

MAX_FILE_NAME_LENGTH Maximum file name length. Taken from `UNZ_MAXFILENAMEINZIP` constant in unzip.c.

Definition at line 88 of file quazip.h.

```
88                  {
89       MAX_FILE_NAME_LENGTH=256
92     };
```

12.70.2.3 enum QuaZip::Mode

Enumerator

mdNotOpen ZIP file is not open. This is the initial mode.

mdUnzip ZIP file is open for reading files inside it.

mdCreate ZIP file was created with open() call.

mdAppend ZIP file was opened in append mode. This refers to `APPEND_STATUS_CREATEAFTER` mode in ZIP/UNZIP package and means that zip is appended to some existing file what is useful when that file contains self-extractor code. This is obviously *not* what you whant to use to add files to the existing ZIP archive.

mdAdd ZIP file was opened for adding files in the archive.

Definition at line 94 of file quazip.h.

```
94              {
95       mdNotOpen,
96       mdUnzip,
97       mdCreate,
98       mdAppend,
106       mdAdd
107     };
```

12.70.3 Constructor & Destructor Documentation

12.70.3.1 QuaZip::QuaZip (const QuaZip & *that*) [private]

12.70.3.2 QuaZip::QuaZip ( )

Call setName() before opening constructed object.

Definition at line 193 of file quazip.cpp.

```
193               :
194   p(new QuaZipPrivate(this))
195 {
196 }
```

12.70.3.3 QuaZip::QuaZip (const QString & *zipName*)

Definition at line 198 of file quazip.cpp.

```
198                                    :
199   p(new QuaZipPrivate(this, zipName))
200 {
201 }
```

12.70.3.4 QuaZip::QuaZip (QIODevice * *ioDevice*)

The IO device must be seekable, otherwise an error will occur when opening.

Definition at line 203 of file quazip.cpp.

```
203                                  :
204   p(new QuaZipPrivate(this, ioDevice))
205 {
206 }
```

12.70.3.5 QuaZip::~QuaZip ( )

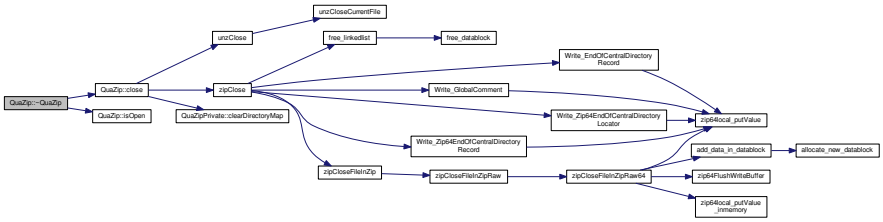
Calls close() if necessary.

Definition at line 208 of file quazip.cpp.

References close(), isOpen(), and p.

```
209 {
210   if(isOpen())
211     close();
212   delete p;
213 }
```

Here is the call graph for this function:



12.70.4 Member Function Documentation

12.70.4.1 void QuaZip::close ( )

Call getZipError() to determine if the close was successful.

If the file was opened by name, then the underlying QIODevice is closed and deleted.

If the underlying QIODevice was set explicitly using setIoDevice() or the appropriate constructor, then it is closed if the auto-close flag is set (which it is by default). Call setAutoClose() to clear the auto-close flag if this behavior is undesirable.

Since Qt 5.1, the QSaveFile was introduced. It breaks the QIODevice API by making close() private and crashing the application if it is called from the base class where it is public. It is an excellent example of poor design that illustrates why you should never ever break an is-a relationship between the base class and a subclass. QuaZIP works around this bug by checking if the QIODevice is an instance of QSaveFile, using qobject_cast<>, and if it is, calls QSaveFile::commit() instead of close(). It is a really ugly hack, but at least it makes your programs work instead of crashing. Note that if the auto-close flag is cleared, then this is a non-issue, and commit() isn't called.

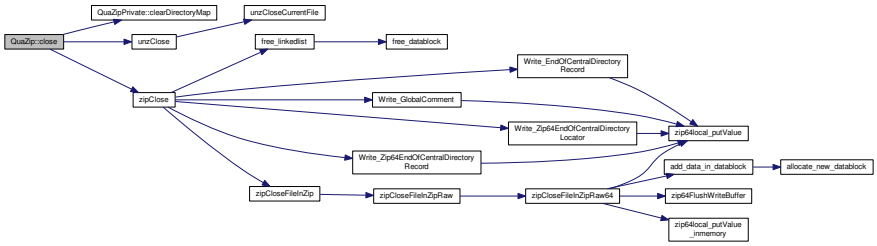
Definition at line 324 of file quazip.cpp.

References QuaZipPrivate::clearDirectoryMap(), QuaZipPrivate::comment, QuaZipPrivate::commentCodec, QuaZipPrivate::ioDevice, mdAdd, mdAppend, mdCreate, mdNotOpen, mdUnzip, QuaZipPrivate::mode, p, UNZ_OK, unzClose(), zipClose(), QuaZipPrivate::zipError, and QuaZipPrivate::zipName.

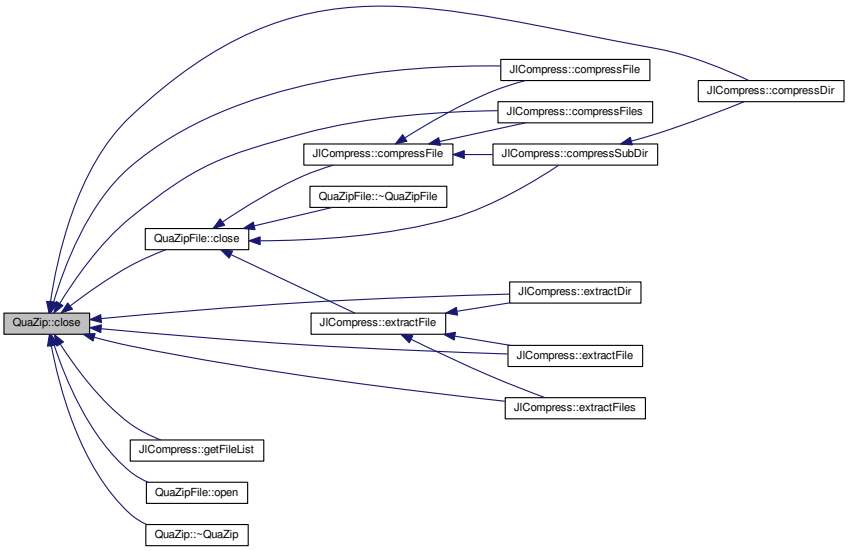
Referenced by QuaZipFile::close(), JlCompress::compressDir(), JlCompress::compressFile(), JlCompress::compressFiles(), JlCompress::extractDir(), JlCompress::extractFile(), JlCompress::extractFiles(), JlCompress::getFileList(), QuaZipFile::open(), and ~QuaZip().

```
325 {
326   p->zipError=UNZ_OK;
327   switch(p->mode) {
328     case mdNotOpen:
329       qWarning("QuaZip::close(): ZIP is not open");
330       return;
331     case mdUnzip:
332       p->zipError=unzClose(p->unzFile_f);
333       break;
334     case mdCreate:
335     case mdAppend:
336     case mdAdd:
337       p->zipError=zipClose(p->zipFile_f,
338           p->comment.isNull() ? NULL
339           : p->commentCodec->fromUnicode(p->comment).constData());
340       break;
341     default:
342       qWarning("QuaZip::close(): unknown mode: %d", (int)p->mode);
343       return;
344   }
345   // opened by name, need to delete the internal IO device
346   if (!p->zipName.isEmpty()) {
347       delete p->ioDevice;
348       p->ioDevice = NULL;
349   }
350   p->clearDirectoryMap();
351   if(p->zipError==UNZ_OK)
352     p->mode=mdNotOpen;
353 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.70.4.2 Qt::CaseSensitivity QuaZip::convertCaseSensitivity (QuaZip::CaseSensitivity *cs*) [static]

Parameters

cs	The value to convert.

Returns

If CaseSensitivity::csDefault, then returns the default file name case sensitivity for the platform. Otherwise, just returns the appropriate value from the Qt::CaseSensitivity enum.

Definition at line 747 of file quazip.cpp.

References csDefault, and csSensitive.

Referenced by QuaZipDir::exists(), and setCurrentFile().

```
748 {
749   if (cs == csDefault) {
750 #ifdef Q_WS_WIN
751       return Qt::CaseInsensitive;
752 #else
```

```
753       return Qt::CaseSensitive;
754 #endif
755   } else {
756       return cs == csSensitive ? Qt::CaseSensitive : Qt::CaseInsensitive;
757   }
758 }
```

Here is the caller graph for this function:

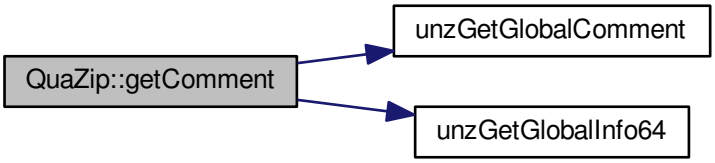
12.70.4.3 QString QuaZip::getComment ( ) const

Definition at line 389 of file quazip.cpp.

References QuaZipPrivate::commentCodec, mdUnzip, QuaZipPrivate::mode, p, unz_global_info64_s::size_comment, UNZ_OK, unzGetGlobalComment(), unzGetGlobalInfo64(), and QuaZipPrivate::zipError.

```
390 {
391   QuaZip *fakeThis=(QuaZip*)this; // non-const
392   fakeThis->p->zipError=UNZ_OK;
393   if(p->mode!=mdUnzip) {
394     qWarning("QuaZip::getComment(): ZIP is not open in mdUnzip mode");
395     return QString();
396   }
397   unz_global_info64 globalInfo;
398   QByteArray comment;
399   if((fakeThis->p->zipError=unzGetGlobalInfo64(p->unzFile_f, &globalInfo))!=
      UNZ_OK)
400     return QString();
401   comment.resize(globalInfo.size_comment);
402   if((fakeThis->p->zipError=unzGetGlobalComment(p->unzFile_f, comment.data(),
      comment.size())) < 0)
403     return QString();
404   fakeThis->p->zipError = UNZ_OK;
405   return p->commentCodec->toUnicode(comment);
406 }
```

Here is the call graph for this function:



12.70.4.4 QTextCodec * QuaZip::getCommentCodec ( ) const

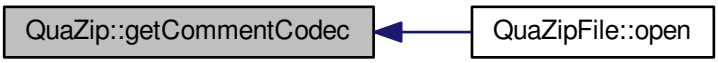
Definition at line 597 of file quazip.cpp.

References QuaZipPrivate::commentCodec, and p.

Referenced by QuaZipFile::open().

```
598 {
599   return p->commentCodec;
600 }
```

Here is the caller graph for this function:



12.70.4.5 bool QuaZip::getCurrentFileInfo (QuaZipFileInfo * *info*) const

Fills the structure pointed by *info*. Returns `true` on success, `false` otherwise. In the latter case structure pointed by *info* remains untouched. If there was an error, getZipError() returns error code.

Should be used only in QuaZip::mdUnzip mode.

Does nothing and returns `false` in any of the following cases.

- ZIP is not open;
- ZIP does not have current file.

In both cases getZipError() returns `UNZ_OK` since there is no ZIP/UNZIP API call.

This overload doesn't support zip64, but will work OK on zip64 archives except that if one of the sizes (compressed or uncompressed) is greater than 0xFFFFFFFFu, it will be set to exactly 0xFFFFFFFFu.

See Also

getCurrentFileInfo(QuaZipFileInfo64* info)const
QuaZipFileInfo64::toQuaZipFileInfo(QuaZipFileInfo&)const

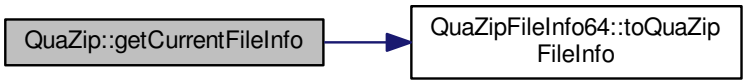
Definition at line 492 of file quazip.cpp.

References QuaZipFileInfo64::toQuaZipFileInfo().

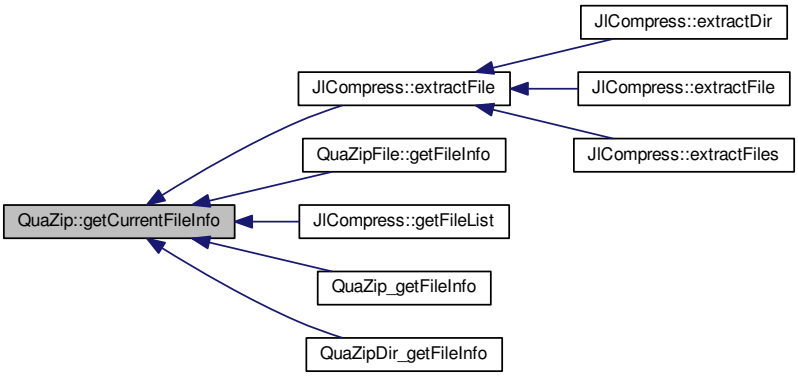
Referenced by JlCompress::extractFile(), QuaZipFile::getFileInfo(), JlCompress::getFileList(), QuaZip_getFileInfo(), and QuaZipDir_getFileInfo().

```
493 {
494     QuaZipFileInfo64 info64;
495     if (info == NULL) { // Very unlikely because of the overloads
496         return false;
497     }
498     if (getCurrentFileInfo(&info64)) {
499         info64.toQuaZipFileInfo(*info);
500         return true;
501     } else {
502         return false;
503     }
504 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.70.4.6 bool QuaZip::getCurrentFileInfo (QuaZipFileInfo64 * *info*) const

This is an overloaded member function, provided for convenience. It differs from the above function only in what argument(s) it accepts.

This function supports zip64. If the archive doesn't use zip64, it is completely equivalent to getCurrentFileInfo(QuaZipFileInfo* info) except for the argument type.

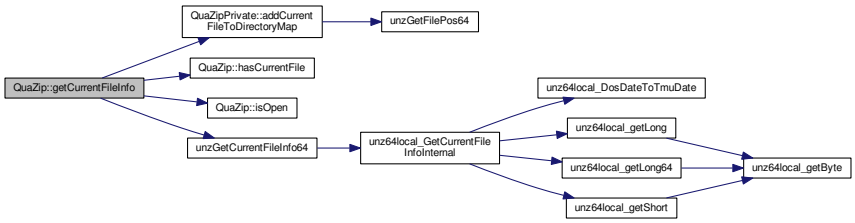
See Also

Definition at line 506 of file quazip.cpp.

References QuaZipPrivate::addCurrentFileToDirectoryMap(), QuaZipFileInfo64::comment, QuaZipPrivate::commentCodec, unz_file_info64_s::compressed_size, QuaZipFileInfo64::compressedSize, unz_file_info64_s::compression_method, QuaZipFileInfo64::crc, unz_file_info64_s::crc, QuaZipFileInfo64::dateTime, unz_file_info64_s::disk_num_start, QuaZipFileInfo64::diskNumberStart, unz_file_info64_s::external_fa, QuaZipFileInfo64::externalAttr, QuaZipFileInfo64::extra, QuaZipPrivate::fileNameCodec, unz_file_info64_s::flag, QuaZipFileInfo64::flags, hasCurrentFile(), unz_file_info64_s::internal_fa, QuaZipFileInfo64::internalAttr, isOpen(), mdUnzip, QuaZipFileInfo64::method, QuaZipPrivate::mode, QuaZipFileInfo64::name, p, unz_file_info64_s::size_file_comment, unz_file_info64_s::size_file_extra, unz_file_info64_s::size_filename, tm_unz_s::tm_hour, tm_unz_s::tm_mday, tm_unz_s::tm_min, tm_unz_s::tm_mon, tm_unz_s::tm_sec, tm_unz_s::tm_year, unz_file_info64_s::tmu_date, unz_file_info64_s::uncompressed_size, QuaZipFileInfo64::uncompressedSize, UNZ_OK, unzGetCurrentFileInfo64(), unz_file_info64_s::version, unz_file_info64_s::version_needed, QuaZipFileInfo64::versionCreated, QuaZipFileInfo64::versionNeeded, and QuaZipPrivate::zipError.

```
507 {
508   QuaZip *fakeThis=(QuaZip*)this; // non-const
509   fakeThis->p->zipError=UNZ_OK;
510   if(p->mode!=mdUnzip) {
511     qWarning("QuaZip::getCurrentFileInfo(): ZIP is not open in mdUnzip mode");
512     return false;
513   }
514   unz_file_info64 info_z;
515   QByteArray fileName;
516   QByteArray extra;
517   QByteArray comment;
518   if(info==NULL) return false;
519   if(!isOpen()||!hasCurrentFile()) return false;
520   if((fakeThis->p->zipError=unzGetCurrentFileInfo64(
      p->unzFile_f, &info_z, NULL, 0, NULL, 0, NULL, 0))!=UNZ_OK)
521     return false;
522   fileName.resize(info_z.size_filename);
523   extra.resize(info_z.size_file_extra);
524   comment.resize(info_z.size_file_comment);
525   if((fakeThis->p->zipError=unzGetCurrentFileInfo64(
      p->unzFile_f, NULL,
526       fileName.data(), fileName.size(),
527       extra.data(), extra.size(),
528       comment.data(), comment.size()))!=UNZ_OK)
529     return false;
530   info->versionCreated=info_z.version;
531   info->versionNeeded=info_z.version_needed;
532   info->flags=info_z.flag;
533   info->method=info_z.compression_method;
534   info->crc=info_z.crc;
535   info->compressedSize=info_z.compressed_size;
536   info->uncompressedSize=info_z.uncompressed_size;
537   info->diskNumberStart=info_z.disk_num_start;
538   info->internalAttr=info_z.internal_fa;
539   info->externalAttr=info_z.external_fa;
540   info->name=p->fileNameCodec->toUnicode(fileName);
541   info->comment=p->commentCodec->toUnicode(comment);
542   info->extra=extra;
543   info->dateTime=QDateTime(
544       QDate(info_z.tmu_date.tm_year, info_z.tmu_date.
      tm_mon+1, info_z.tmu_date.tm_mday),
545       QTime(info_z.tmu_date.tm_hour, info_z.tmu_date.
      tm_min, info_z.tmu_date.tm_sec));
546   // Add to directory map
547   p->addCurrentFileToDirectoryMap(info->name);
548   return true;
549 }
```

Here is the call graph for this function:



12.70.4.7 QString QuaZip::getCurrentFileName ( ) const

Equivalent to calling getCurrentFileInfo() and then getting `name` field of the QuaZipFileInfo structure, but faster and more convenient.

Should be used only in QuaZip::mdUnzip mode.

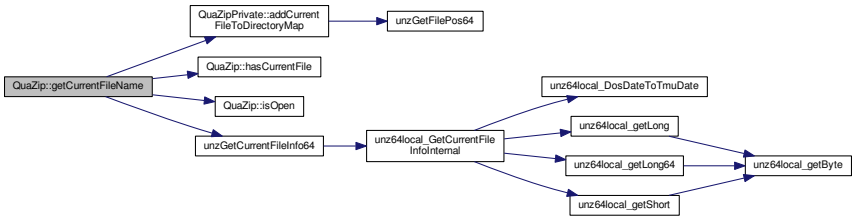
Definition at line 551 of file quazip.cpp.

References QuaZipPrivate::addCurrentFileToDirectoryMap(), QuaZipPrivate::fileNameCodec, hasCurrentFile(), isOpen(), MAX_FILE_NAME_LENGTH, mdUnzip, QuaZipPrivate::mode, p, UNZ_OK, unzGetCurrentFileInfo64(), and QuaZipPrivate::zipError.

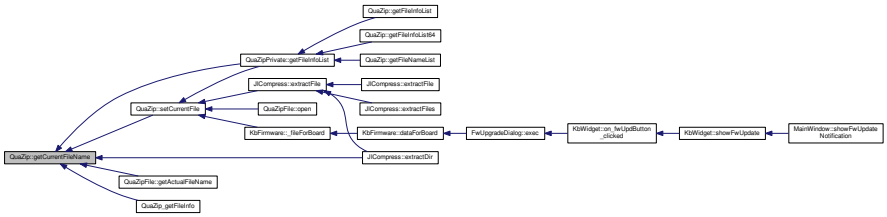
Referenced by JlCompress::extractDir(), QuaZipFile::getActualFileName(), QuaZipPrivate::getFileInfoList(), QuaZip_getFileInfo(), and setCurrentFile().

```
552 {
553   QuaZip *fakeThis=(QuaZip*)this; // non-const
554   fakeThis->p->zipError=UNZ_OK;
555   if(p->mode!=mdUnzip) {
556     qWarning("QuaZip::getCurrentFileName(): ZIP is not open in mdUnzip mode");
557     return QString();
558   }
559   if(!isOpen()||!hasCurrentFile()) return QString();
560   QByteArray fileName(MAX_FILE_NAME_LENGTH, 0);
561   if((fakeThis->p->zipError=unzGetCurrentFileInfo64(
      p->unzFile_f, NULL, fileName.data(), fileName.size(),
562       NULL, 0, NULL, 0))!=UNZ_OK)
563     return QString();
564   QString result = p->fileNameCodec->toUnicode(fileName.constData());
565   if (result.isEmpty())
566       return result;
567   // Add to directory map
568   p->addCurrentFileToDirectoryMap(result);
569   return result;
570 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.70.4.8 int QuaZip::getEntriesCount ( ) const

Returns negative error code in the case of error. The same error code will be returned by subsequent getZipError() call.

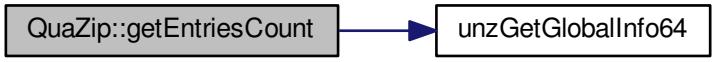
Definition at line 375 of file quazip.cpp.

References mdUnzip, QuaZipPrivate::mode, unz_global_info64_s::number_entry, p, UNZ_OK, unzGetGlobalInfo64(), and QuaZipPrivate::zipError.

```
376 {
377   QuaZip *fakeThis=(QuaZip*)this; // non-const
378   fakeThis->p->zipError=UNZ_OK;
379   if(p->mode!=mdUnzip) {
380     qWarning("QuaZip::getEntriesCount(): ZIP is not open in mdUnzip mode");
381     return -1;
```

```
382   }
383   unz_global_info64 globalInfo;
384   if((fakeThis->p->zipError=unzGetGlobalInfo64(p->unzFile_f, &globalInfo))!=
      UNZ_OK)
385     return p->zipError;
386   return (int)globalInfo.number_entry;
387 }
```

Here is the call graph for this function:



12.70.4.9 QList< QuaZipFileInfo > QuaZip::getFileInfoList ( ) const

Returns

A list of QuaZipFileInfo objects or an empty list if there was an error or if the archive is empty (call getZipError() to figure out which).

This function doesn't support zip64, but will still work with zip64 archives, converting results using QuaZipFileInfo64::toQuaZipFileInfo(). If all file sizes are below 4 GB, it will work just fine.

See Also

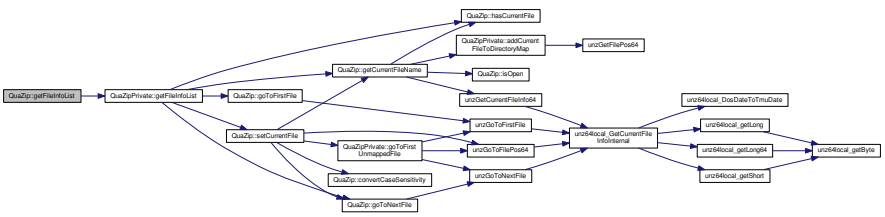
getFileNameList()
getFileInfoList64()

Definition at line 729 of file quazip.cpp.

References QuaZipPrivate::getFileInfoList(), and p.

```
730 {
731     QList<QuaZipFileInfo> list;
732     if (p->getFileInfoList(&list))
733         return list;
734     else
735         return QList<QuaZipFileInfo>();
736 }
```

Here is the call graph for this function:



12.70.4.10 QList< QuaZipFileInfo64 > QuaZip::getFileInfoList64 ( ) const

This is an overloaded member function, provided for convenience. It differs from the above function only in what argument(s) it accepts.

This function supports zip64.

See Also

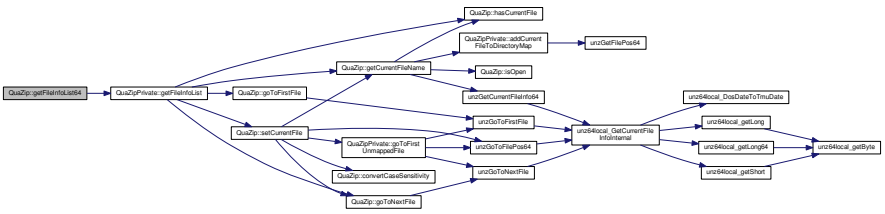
getFileNameList()
getFileInfoList()

Definition at line 738 of file quazip.cpp.

References QuaZipPrivate::getFileInfoList(), and p.

```
739 {
740     QList<QuaZipFileInfo64> list;
741     if (p->getFileInfoList(&list))
742         return list;
743     else
744         return QList<QuaZipFileInfo64>();
745 }
```

Here is the call graph for this function:



12.70.4.11 QTextCodec * QuaZip::getFileNameCodec ( ) const

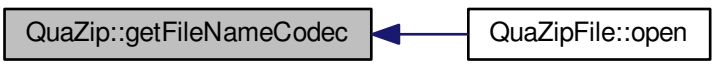
Definition at line 582 of file quazip.cpp.

References QuaZipPrivate::fileNameCodec, and p.

Referenced by QuaZipFile::open().

```
583 {
584   return p->fileNameCodec;
585 }
```

Here is the caller graph for this function:



12.70.4.12 QStringList QuaZip::getFileNameList ( ) const

Returns

A list of file names or an empty list if there was an error or if the archive is empty (call getZipError() to figure out which).

See Also

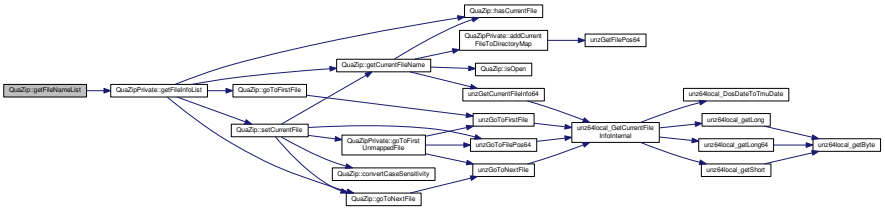
getFileInfoList()

Definition at line 720 of file quazip.cpp.

References QuaZipPrivate::getFileInfoList(), and p.

```
721 {
722     QStringList list;
723     if (p->getFileInfoList(&list))
724         return list;
725     else
726         return QStringList();
727 }
```

Here is the call graph for this function:



12.70.4.13 QIODevice * QuaZip::getIoDevice ( ) const

Returns null string if no device has been set explicitly, for example when opening a ZIP file by name.

See Also

setIoDevice(), getZipName(), setZipName()

Definition at line 607 of file quazip.cpp.

References QuaZipPrivate::ioDevice, p, and QuaZipPrivate::zipName.

```
608 {
609   if (!p->zipName.isEmpty()) // opened by name, using an internal QIODevice
610     return NULL;
611   return p->ioDevice;
612 }
```

12.70.4.14 QuaZip::Mode QuaZip::getMode ( ) const

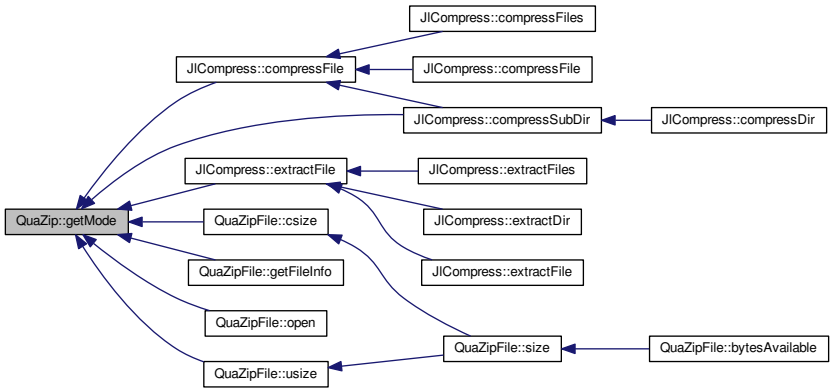
Definition at line 614 of file quazip.cpp.

References QuaZipPrivate::mode, and p.

Referenced by JlCompress::compressFile(), JlCompress::compressSubDir(), QuaZipFile::csize(), JlCompress::extractFile(), QuaZipFile::getFileInfo(), QuaZipFile::open(), and QuaZipFile::usize().

```
615 {
616   return p->mode;
617 }
```

Here is the caller graph for this function:



12.70.4.15 unzFile QuaZip::getUnzFile ( )

You can use this handle to directly call UNZIP part of the ZIP/UNZIP package functions (see unzip.h).

Warning

> When using the handle returned by this function, please keep in mind that QuaZip class is unable to detect any changes you make in the ZIP file state (e. g. changing current file, or closing the handle). So please do not do anything with this handle that is possible to do with the functions of this class. Or at least return the handle in the original state before calling some another function of this class (including implicit destructor calls and calls from the QuaZipFile objects that refer to this QuaZip instance!). So if you have changed the current file in the ZIP archive - then change it back or you may experience some strange behavior or even crashes.

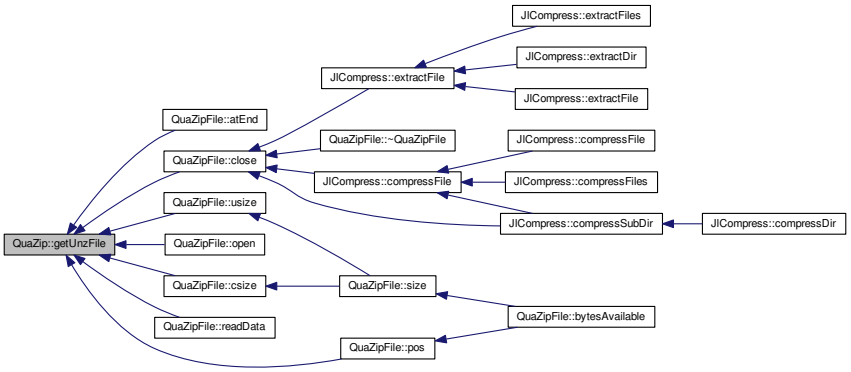
Definition at line 639 of file quazip.cpp.

References p.

Referenced by QuaZipFile::atEnd(), QuaZipFile::close(), QuaZipFile::csize(), QuaZipFile::open(), QuaZipFile::pos(), QuaZipFile::readData(), and QuaZipFile::usize().

```
640 {
641   return p->unzFile_f;
642 }
```

Here is the caller graph for this function:



12.70.4.16 int QuaZip::getZipError ( ) const

Returns `UNZ_OK` if the last operation was successful.

Error code resets to `UNZ_OK` every time you call any function that accesses something inside ZIP archive, even if it is `const` (like getEntriesCount()). open() and close() calls reset error code too. See documentation for the specific functions for details on error detection.

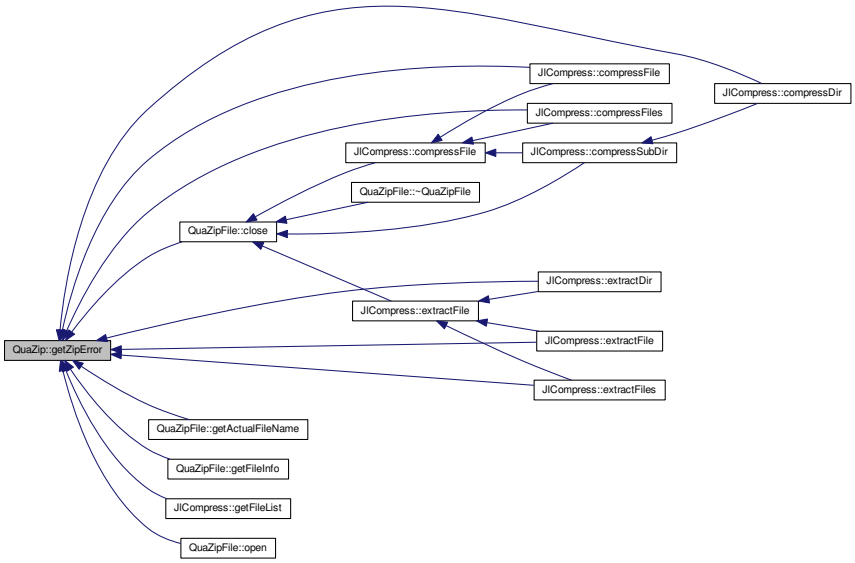
Definition at line 624 of file quazip.cpp.

References p, and QuaZipPrivate::zipError.

Referenced by QuaZipFile::close(), JlCompress::compressDir(), JlCompress::compressFile(), JlCompress::compressFiles(), JlCompress::extractDir(), JlCompress::extractFile(), JlCompress::extractFiles(), QuaZipFile::getActualFileName(), QuaZipFile::getFileInfo(), JlCompress::getFileList(), and QuaZipFile::open().

```
625 {
626   return p->zipError;
627 }
```

Here is the caller graph for this function:



12.70.4.17 zipFile QuaZip::getZipFile ( )

You can use this handle to directly call ZIP part of the ZIP/UNZIP package functions (see zip.h). Warnings about the getUnzFile() function also apply to this function.

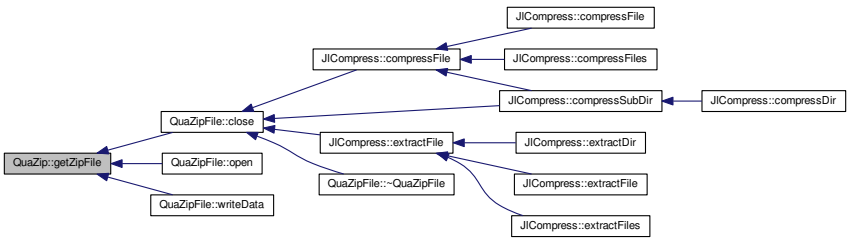
Definition at line 644 of file quazip.cpp.

References p.

Referenced by QuaZipFile::close(), QuaZipFile::open(), and QuaZipFile::writeData().

```
645 {
646   return p->zipFile_f;
647 }
```

Here is the caller graph for this function:



12.70.4.18 QString QuaZip::getZipName ( ) const

Returns null string if no ZIP file name has been set, for example when the QuaZip instance is set up to use a QIODevice instead.

See Also

setZipName(), setIoDevice(), getIoDevice()

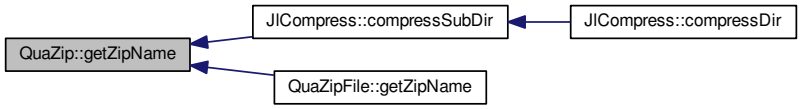
Definition at line 602 of file quazip.cpp.

References p, and QuaZipPrivate::zipName.

Referenced by JlCompress::compressSubDir(), and QuaZipFile::getZipName().

```
603 {
604   return p->zipName;
605 }
```

Here is the caller graph for this function:



12.70.4.19 bool QuaZip::goToFirstFile ( )

Returns `true` on success, `false` otherwise. Call getZipError() to get the error code.

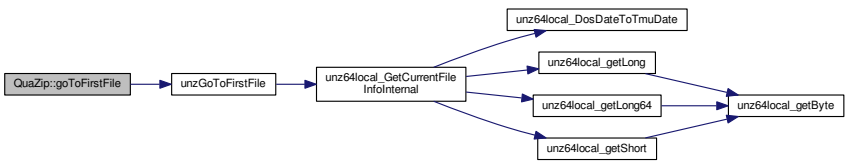
Definition at line 466 of file quazip.cpp.

References QuaZipPrivate::hasCurrentFile_f, mdUnzip, QuaZipPrivate::mode, p, UNZ_OK, unzGoToFirstFile(), and QuaZipPrivate::zipError.

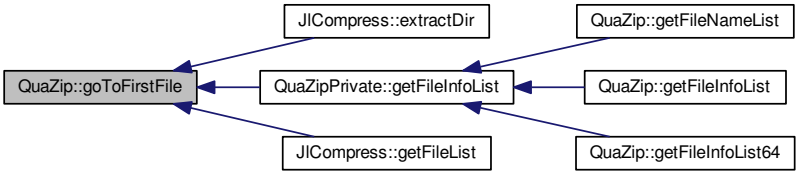
Referenced by JlCompress::extractDir(), QuaZipPrivate::getFileInfoList(), and JlCompress::getFileList().

```
467 {
468   p->zipError=UNZ_OK;
469   if(p->mode!=mdUnzip) {
470     qWarning("QuaZip::goToFirstFile(): ZIP is not open in mdUnzip mode");
471     return false;
472   }
473   p->zipError=unzGoToFirstFile(p->unzFile_f);
474   p->hasCurrentFile_f=p->zipError==UNZ_OK;
475   return p->hasCurrentFile_f;
476 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.70.4.20 bool QuaZip::goToNextFile ( )

Returns `true` on success, `false` otherwise. Call getZipError() to determine if there was an error.

Should be used only in QuaZip::mdUnzip mode.

Note

If the end of file was reached, getZipError() will return `UNZ_OK` instead of `UNZ_END_OF_LIST_OF_FILE`. This is to make things like this easier:

```
for(bool more=zip.goToFirstFile(); more; more=zip.goToNextFile()) {
  // do something
}
if(zip.getZipError()==UNZ_OK) {
  // ok, there was no error
}
```

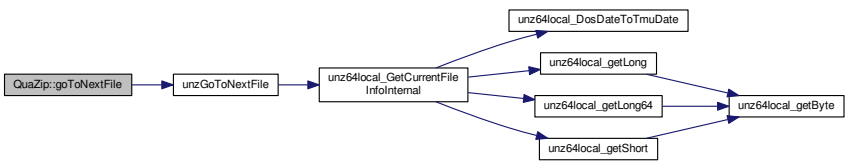
Definition at line 478 of file quazip.cpp.

References QuaZipPrivate::hasCurrentFile_f, mdUnzip, QuaZipPrivate::mode, p, UNZ_END_OF_LIST_OF_FILE, UNZ_OK, unzGoToNextFile(), and QuaZipPrivate::zipError.

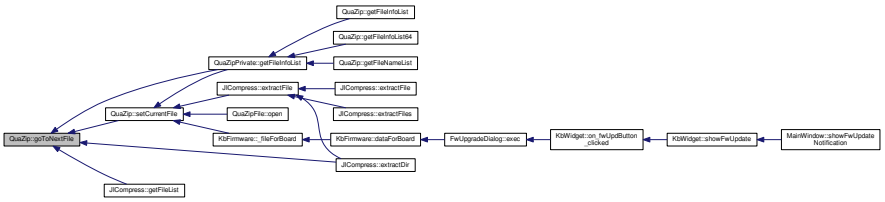
Referenced by JlCompress::extractDir(), QuaZipPrivate::getFileInfoList(), JlCompress::getFileList(), and setCurrentFile().

```
479 {
480   p->zipError=UNZ_OK;
481   if(p->mode!=mdUnzip) {
482     qWarning("QuaZip::goToFirstFile(): ZIP is not open in mdUnzip mode");
483     return false;
484   }
485   p->zipError=unzGoToNextFile(p->unzFile_f);
486   p->hasCurrentFile_f=p->zipError==UNZ_OK;
487   if(p->zipError==UNZ_END_OF_LIST_OF_FILE)
488     p->zipError=UNZ_OK;
489   return p->hasCurrentFile_f;
490 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.70.4.21 bool QuaZip::hasCurrentFile ( ) const

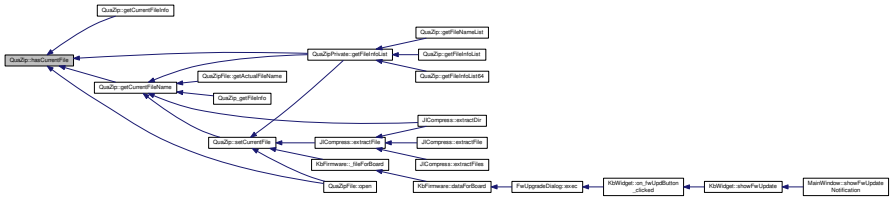
Definition at line 634 of file quazip.cpp.

References QuaZipPrivate::hasCurrentFile_f, and p.

Referenced by getCurrentFileInfo(), getCurrentFileName(), QuaZipPrivate::getFileInfoList(), and QuaZipFile::open().

```
635 {
636   return p->hasCurrentFile_f;
637 }
```

Here is the caller graph for this function:



12.70.4.22 bool QuaZip::isAutoClose ( ) const

See Also

setAutoClose()

Definition at line 780 of file quazip.cpp.

References QuaZipPrivate::autoClose, and p.

```
781 {
782     return p->autoClose;
783 }
```

12.70.4.23 bool QuaZip::isDataDescriptorWritingEnabled ( ) const

See Also

setDataDescriptorWritingEnabled()

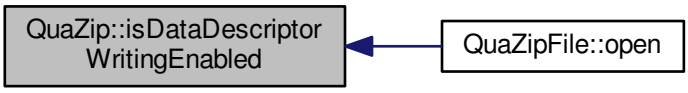
Definition at line 654 of file quazip.cpp.

References QuaZipPrivate::dataDescriptorWritingEnabled, and p.

Referenced by QuaZipFile::open().

```
655 {
656     return p->dataDescriptorWritingEnabled;
657 }
```

Here is the caller graph for this function:



12.70.4.24 bool QuaZip::isOpen ( ) const

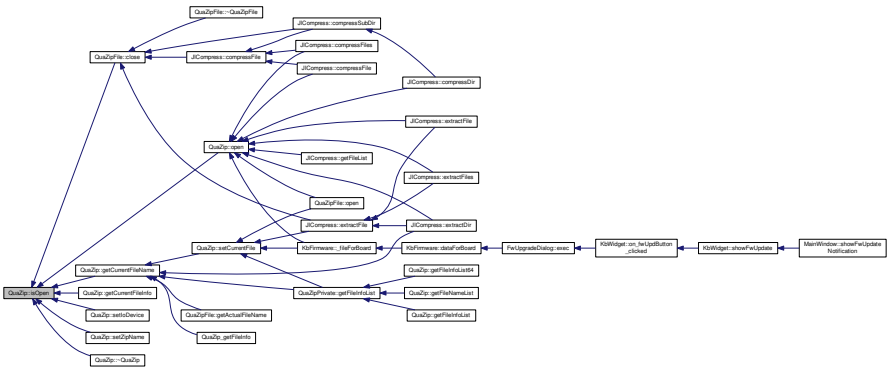
Definition at line 619 of file quazip.cpp.

References mdNotOpen, QuaZipPrivate::mode, and p.

Referenced by QuaZipFile::close(), getCurrentFileInfo(), getCurrentFileName(), open(), setIoDevice(), setZipName(), and ~QuaZip().

```
620 {
621   return p->mode!=mdNotOpen;
622 }
```

Here is the caller graph for this function:



12.70.4.25 bool QuaZip::isZip64Enabled ( ) const

Returns

`true` if and only if the zip64 mode is enabled.

See Also

setZip64Enabled()

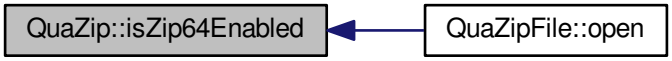
Definition at line 775 of file quazip.cpp.

References p, and QuaZipPrivate::zip64.

Referenced by QuaZipFile::open().

```
776 {
777     return p->zip64;
778 }
```

Here is the caller graph for this function:



12.70.4.26 bool QuaZip::open (Mode *mode*, zlib_filefunc_def * *ioApi* = NULL)

Argument *mode* specifies open mode of the ZIP archive. See Mode for details. Note that there is zipOpen2() function in the ZIP/UNZIP API which accepts *globalcomment* argument, but it does not use it anywhere, so this open() function does not have this argument. See setComment() if you need to set global comment.

If the ZIP file is accessed via explicitly set QIODevice, then this device is opened in the necessary mode. If the device was already opened by some other means, then QuaZIP checks if the open mode is compatible to the mode needed for the requested operation. If necessary, seeking is performed to position the device properly.

Returns

`true` if successful, `false` otherwise.

Note

ZIP/UNZIP API open calls do not return error code - they just return `NULL` indicating an error. But to make things easier, quazip.h header defines additional error code `UNZ_ERROROPEN` and getZipError() will return it if the open call of the ZIP/UNZIP API returns `NULL`.

Argument *ioApi* specifies IO function set for ZIP/UNZIP package to use. See unzip.h, zip.h and ioapi.h for details. Note that IO API for QuaZip is different from the original package. The file path argument was changed to be of type `voidpf`, and QuaZip passes a QIODevice pointer there. This QIODevice is either set explicitly via setIoDevice() or the QuaZip(QIODevice*) constructor, or it is created internally when opening the archive by its file name. The default API (qioapi.cpp) just delegates everything to the QIODevice API. Not only this allows to use a QIODevice instead of file name, but also has a nice side effect of raising the file size limit from 2G to 4G (in non-zip64 archives).

Note

If the zip64 support is needed, the ioApi argument *must* be NULL because due to the backwards compatibility issues it can be used to provide a 32-bit API only.
If the no-auto-close feature is used, then the *ioApi* argument *should* be NULL because the old API doesn't support the 'fake close' operation, causing slight memory leaks and other possible troubles (like closing the output device in case when an error occurs during opening).

In short: just forget about the *ioApi* argument and you'll be fine.

Definition at line 215 of file quazip.cpp.

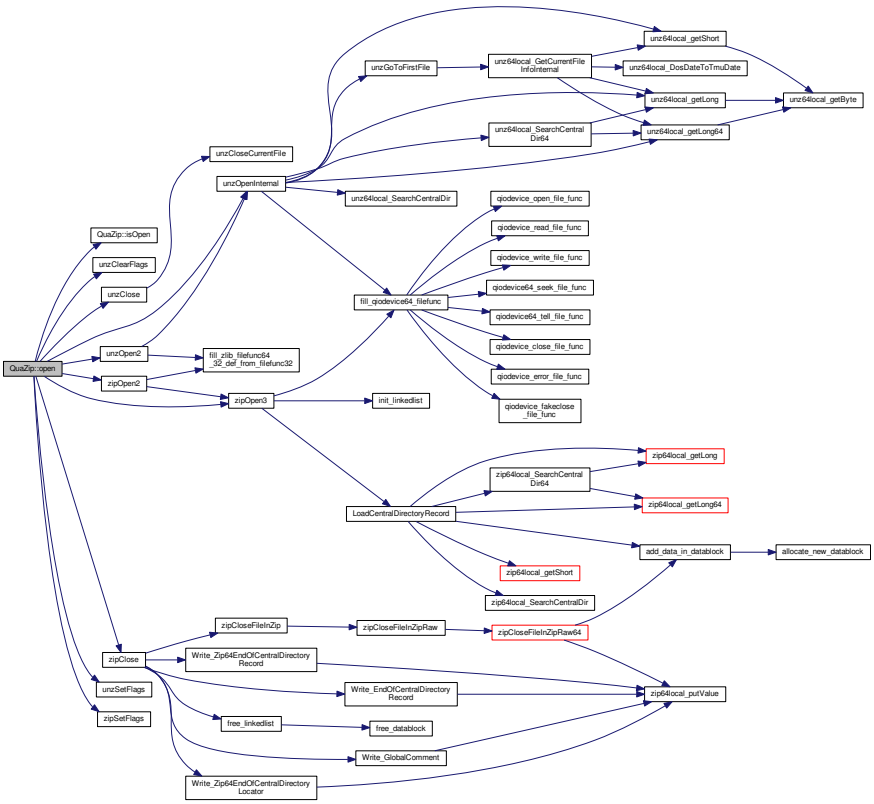
References APPEND_STATUS_ADDINZIP, APPEND_STATUS_CREATE, APPEND_STATUS_CREATEAFTER, QuaZipPrivate::autoClose, QuaZipPrivate::dataDescriptorWritingEnabled, QuaZipPrivate::ioDevice, isOpen(), mdAdd, mdAppend, mdCreate, mdUnzip, QuaZipPrivate::mode, p, UNZ_AUTO_CLOSE, UNZ_OK, UNZ_OPENERROR, unzClearFlags(), unzClose(), unzOpen2(), unzOpenInternal(), unzSetFlags(), ZIP_AUTO_CLOSE, ZIP_SEQUENTIAL, ZIP_WRITE_DATA_DESCRIPTOR, zipClose(), QuaZipPrivate::zipError, QuaZipPrivate::zipName, zipOpen2(), zipOpen3(), and zipSetFlags().

Referenced by KbFirmware::_fileForBoard(), JlCompress::compressDir(), JlCompress::compressFile(), JlCompress::compressFiles(), JlCompress::extractDir(), JlCompress::extractFile(), JlCompress::extractFiles(), JlCompress::getFileList(), and QuaZipFile::open().

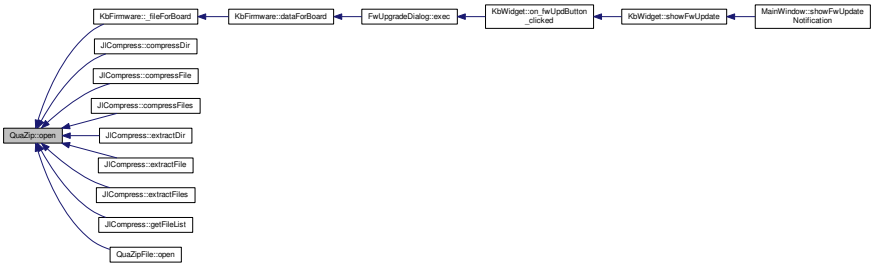
```
216 {
217   p->zipError=UNZ_OK;
218   if(isOpen()) {
219     qWarning("QuaZip::open(): ZIP already opened");
220     return false;
221   }
222   QIODevice *ioDevice = p->ioDevice;
223   if (ioDevice == NULL) {
224     if (p->zipName.isEmpty()) {
225       qWarning("QuaZip::open(): set either ZIP file name or IO device first");
226       return false;
227     } else {
228       ioDevice = new QFile(p->zipName);
229     }
230   }
231   unsigned flags = 0;
232   switch(mode) {
233     case mdUnzip:
234       if (ioApi == NULL) {
235           if (p->autoClose)
236               flags |= UNZ_AUTO_CLOSE;
237           p->unzFile_f=unzOpenInternal(ioDevice, NULL, 1, flags);
238       } else {
239           // QuaZIP pre-zip64 compatibility mode
240           p->unzFile_f=unzOpen2(ioDevice, ioApi);
241           if (p->unzFile_f != NULL) {
242               if (p->autoClose) {
243                   unzSetFlags(p->unzFile_f, UNZ_AUTO_CLOSE);
244               } else {
245                   unzClearFlags(p->unzFile_f, UNZ_AUTO_CLOSE);
246               }
247           }
248       }
249       if(p->unzFile_f!=NULL) {
250         if (ioDevice->isSequential()) {
251             unzClose(p->unzFile_f);
252             if (!p->zipName.isEmpty())
253                 delete ioDevice;
254             qWarning("QuaZip::open(): "
255                      "only mdCreate can be used with "
256                      "sequential devices");
257             return false;
258         }
259         p->mode=mode;
260         p->ioDevice = ioDevice;
261         return true;
262       } else {
263         p->zipError=UNZ_OPENERROR;
264         if (!p->zipName.isEmpty())
265           delete ioDevice;
266         return false;
267       }
268     case mdCreate:
269     case mdAppend:
270     case mdAdd:
271       if (ioApi == NULL) {
```

```
272             if (p->autoClose)
273                 flags |= ZIP_AUTO_CLOSE;
274             if (p->dataDescriptorWritingEnabled)
275                 flags |= ZIP_WRITE_DATA_DESCRIPTOR;
276             p->zipFile_f=zipOpen3(ioDevice,
277                 mode==mdCreate?APPEND_STATUS_CREATE:
278                 mode==mdAppend?APPEND_STATUS_CREATEAFTER:
279                 APPEND_STATUS_ADDINZIP,
280                 NULL, NULL, flags);
281         } else {
282             // QuaZIP pre-zip64 compatibility mode
283             p->zipFile_f=zipOpen2(ioDevice,
284                 mode==mdCreate?APPEND_STATUS_CREATE:
285                 mode==mdAppend?APPEND_STATUS_CREATEAFTER:
286                 APPEND_STATUS_ADDINZIP,
287                 NULL,
288                 ioApi);
289             if (p->zipFile_f != NULL) {
290                 zipSetFlags(p->zipFile_f, flags);
291             }
292         }
293         if(p->zipFile_f!=NULL) {
294           if (ioDevice->isSequential()) {
295               if (mode != mdCreate) {
296                   zipClose(p->zipFile_f, NULL);
297                   qWarning("QuaZip::open(): "
298                           "only mdCreate can be used with "
299                            "sequential devices");
300                   if (!p->zipName.isEmpty())
301                       delete ioDevice;
302                   return false;
303               }
304               zipSetFlags(p->zipFile_f, ZIP_SEQUENTIAL);
305           }
306           p->mode=mode;
307           p->ioDevice = ioDevice;
308           return true;
309         } else {
310           p->zipError=UNZ_OPENERROR;
311           if (!p->zipName.isEmpty())
312             delete ioDevice;
313           return false;
314         }
315       default:
316         qWarning("QuaZip::open(): unknown mode: %d", (int)mode);
317         if (!p->zipName.isEmpty())
318           delete ioDevice;
319         return false;
320         break;
321     }
322 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.70.4.27 QuaZip& QuaZip::operator= (const QuaZip & *that*) [private]

12.70.4.28 void QuaZip::setAutoClose (bool *autoClose*) const

By default, QuaZIP opens the underlying QIODevice when open() is called, and closes it when close() is called. In some cases, when the device is set explicitly using setIoDevice(), it may be desirable to leave the device open. If the auto-close flag is unset using this method, then the device isn't closed automatically if it was set explicitly.

If it is needed to clear this flag, it is recommended to do so before opening the archive because otherwise QuaZIP may close the device during the open() call if an error is encountered after the device is opened.

If the device was not set explicitly, but rather the setZipName() or the appropriate constructor was used to set the

ZIP file name instead, then the auto-close flag has no effect, and the internal device is closed nevertheless because there is no other way to close it.

See Also

isAutoClose()
setIoDevice()

Definition at line 785 of file quazip.cpp.

References QuaZipPrivate::autoClose, and p.

```
786 {
787     p->autoClose = autoClose;
788 }
```

12.70.4.29 void QuaZip::setComment (const QString & *comment*)

The comment will be written to the archive on close operation. QuaZip makes a distinction between a null QByteArray() comment and an empty "" comment in the QuaZip::mdAdd mode. A null comment is the default and it means "don't change the comment". An empty comment removes the original comment.

See Also

open()

Definition at line 629 of file quazip.cpp.

References QuaZipPrivate::comment, and p.

```
630 {
631   p->comment=comment;
632 }
```

12.70.4.30 void QuaZip::setCommentCodec (QTextCodec * *commentCodec*)

This codec defaults to locale codec, which is probably ok.

Definition at line 587 of file quazip.cpp.

References QuaZipPrivate::commentCodec, and p.

```
588 {
589   p->commentCodec=commentCodec;
590 }
```

12.70.4.31 void QuaZip::setCommentCodec (const char * *commentCodecName*)

This is an overloaded member function, provided for convenience. It differs from the above function only in what argument(s) it accepts. Equivalent to calling setCommentCodec(QTextCodec::codecForName(codecName));

Definition at line 592 of file quazip.cpp.

References QuaZipPrivate::commentCodec, and p.

```
593 {
594   p->commentCodec=QTextCodec::codecForName(commentCodecName);
595 }
```

12.70.4.32 bool QuaZip::setCurrentFile (const QString & *fileName*, CaseSensitivity *cs* = csDefault)

Returns `true` if successful, `false` otherwise. Argument *cs* specifies case sensitivity of the file name. Call getZipError() in the case of a failure to get error code.

This is not a wrapper to unzLocateFile() function. That is because I had to implement locale-specific case-insensitive comparison.

Here are the differences from the original implementation:

- If the file was not found, error code is `UNZ_OK`, not `UNZ_END_OF_LIST_OF_FILE` (see also goToNextFile()).
- If this function fails, it unsets the current file rather than resetting it back to what it was before the call.

If *fileName* is null string then this function unsets the current file and return `true`. Note that you should close the file first if it is open! See QuaZipFile::QuaZipFile(QuaZip*,QObject*) for the details.

Should be used only in QuaZip::mdUnzip mode.

See Also

setFileNameCodec(), CaseSensitivity

Definition at line 408 of file quazip.cpp.

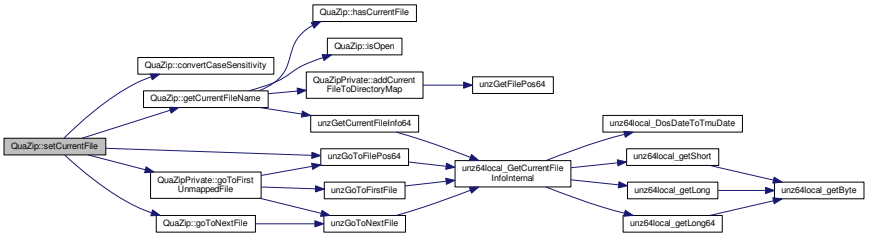
References convertCaseSensitivity(), QuaZipPrivate::directoryCaseInsensitive, QuaZipPrivate::directoryCaseSensitive, getCurrentFileName(), QuaZipPrivate::goToFirstUnmappedFile(), goToNextFile(), QuaZipPrivate::hasCurrentFile_f, MAX_FILE_NAME_LENGTH, mdUnzip, QuaZipPrivate::mode, p, unz64_file_pos_s::pos_in_zip_directory, UNZ_OK, UNZ_PARAMERROR, unzGoToFilePos64(), and QuaZipPrivate::zipError.

Referenced by KbFirmware::_fileForBoard(), JlCompress::extractFile(), QuaZipPrivate::getFileInfoList(), and QuaZipFile::open().

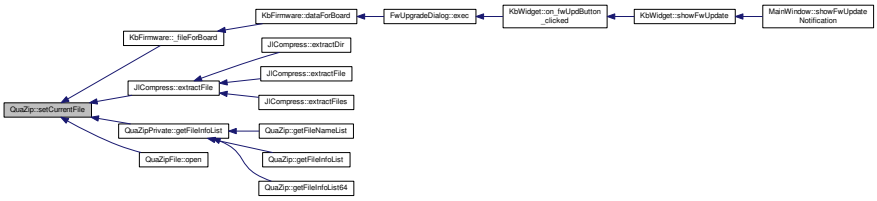
```
409 {
410   p->zipError=UNZ_OK;
411   if(p->mode!=mdUnzip) {
412     qWarning("QuaZip::setCurrentFile(): ZIP is not open in mdUnzip mode");
413     return false;
414   }
415   if(fileName.isEmpty()) {
416     p->hasCurrentFile_f=false;
417     return true;
418   }
419   // Unicode-aware reimplementation of the unzLocateFile function
420   if(p->unzFile_f==NULL) {
421     p->zipError=UNZ_PARAMERROR;
422     return false;
423   }
424   if(fileName.length()>MAX_FILE_NAME_LENGTH) {
425     p->zipError=UNZ_PARAMERROR;
426     return false;
427   }
428   // Find the file by name
429   bool sens = convertCaseSensitivity(cs) == Qt::CaseSensitive;
430   QString lower, current;
431   if(!sens) lower=fileName.toLower();
432   p->hasCurrentFile_f=false;
433
434   // Check the appropriate Map
435   unz64_file_pos fileDirPos;
436   fileDirPos.pos_in_zip_directory = 0;
437   if (sens) {
438       if (p->directoryCaseSensitive.contains(fileName))
439           fileDirPos = p->directoryCaseSensitive.value(fileName);
440   } else {
441       if (p->directoryCaseInsensitive.contains(lower))
442           fileDirPos = p->directoryCaseInsensitive.value(lower);
443   }
444
445   if (fileDirPos.pos_in_zip_directory != 0) {
446       p->zipError = unzGoToFilePos64(p->unzFile_f, &fileDirPos);
447       p->hasCurrentFile_f = p->zipError == UNZ_OK;
448   }
```

```
449
450   if (p->hasCurrentFile_f)
451       return p->hasCurrentFile_f;
452
453   // Not mapped yet, start from where we have got to so far
454   for(bool more=p->goToFirstUnmappedFile(); more; more=
      goToNextFile()) {
455     current=getCurrentFileName();
456     if(current.isEmpty()) return false;
457     if(sens) {
458       if(current==fileName) break;
459     } else {
460       if(current.toLower()==lower) break;
461     }
462   }
463   return p->hasCurrentFile_f;
464 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.70.4.33 void QuaZip::setDataDescriptorWritingEnabled (bool *enabled*)

According to the ZIP format specification, a file inside archive may have a data descriptor immediately following the file data. This is reflected by a special flag in the local file header and in the central directory. By default, QuaZIP sets this flag and writes the data descriptor unless both method and level were set to 0, in which case it operates in 1.0-compatible mode and never writes data descriptors.

By setting this flag to false, it is possible to disable data descriptor writing, thus increasing compatibility with archive readers that don't understand this feature of the ZIP file format.

Setting this flag affects all the QuaZipFile instances that are opened after this flag is set.

The data descriptor writing mode is enabled by default.

Note that if the ZIP archive is written into a QIODevice for which QIODevice::isSequential() returns `true`, then the data descriptor is mandatory and will be written even if this flag is set to false.

Parameters

enabled	If `true`, enable local descriptor writing, disable it otherwise.

See Also

QuaZipFile::isDataDescriptorWritingEnabled()

Definition at line 649 of file quazip.cpp.

References QuaZipPrivate::dataDescriptorWritingEnabled, and p.

```
650 {
651     p->dataDescriptorWritingEnabled = enabled;
652 }
```

12.70.4.34 void QuaZip::setDefaultFileNameCodec (QTextCodec * *codec*) [static]

The default codec is used by the constructors, so calling this function won't affect the QuaZip instances already created at that moment.

The codec specified here can be overriden by calling setFileNameCodec(). If neither function is called, QTextCodec-::codecForLocale() will be used to decode or encode file names. Use this function with caution if the application uses other libraries that depend on QuaZIP. Those libraries can either call this function by themselves, thus overriding your setting or can rely on the default encoding, thus failing mysteriously if you change it. For these reasons, it isn't recommended to use this function if you are developing a library, not an application. Instead, ask your library users to call it in case they need specific encoding.

In most cases, using setFileNameCodec() instead is the right choice. However, if you depend on third-party code that uses QuaZIP, then the reasons stated above can actually become a reason to use this function in case the third-party code in question fails because it doesn't understand the encoding you need and doesn't provide a way to specify it. This applies to the JlCompress class as well, as it was contributed and doesn't support explicit encoding parameters.

In short: use setFileNameCodec() when you can, resort to setDefaultFileNameCodec() when you don't have access to the QuaZip instance.

Parameters

codec	The codec to use by default. If NULL, resets to default.

Definition at line 760 of file quazip.cpp.

References QuaZipPrivate::defaultFileNameCodec.

Referenced by setDefaultFileNameCodec().

```
761 {
762     QuaZipPrivate::defaultFileNameCodec = codec;
763 }
```

Here is the caller graph for this function:

12.70.4.35 void QuaZip::setDefaultFileNameCodec (const char * *codecName*) [static]

Definition at line 765 of file quazip.cpp.

References setDefaultFileNameCodec().

```
766 {
767     setDefaultFileNameCodec(QTextCodec::codecForName(codecName));
768 }
```

Here is the call graph for this function:

12.70.4.36 void QuaZip::setFileNameCodec (QTextCodec * *fileNameCodec*)

This is necessary to access files in the ZIP archive created under Windows with non-latin characters in file names. For example, file names with cyrillic letters will be in `IBM866` encoding.

Definition at line 572 of file quazip.cpp.

References QuaZipPrivate::fileNameCodec, and p.

```
573 {
574   p->fileNameCodec=fileNameCodec;
575 }
```

12.70.4.37 void QuaZip::setFileNameCodec (const char * *fileNameCodecName*)

This is an overloaded member function, provided for convenience. It differs from the above function only in what argument(s) it accepts. Equivalent to calling setFileNameCodec(QTextCodec::codecForName(codecName));

Definition at line 577 of file quazip.cpp.

References QuaZipPrivate::fileNameCodec, and p.

```
578 {
579   p->fileNameCodec=QTextCodec::codecForName(fileNameCodecName);
580 }
```

12.70.4.38 void QuaZip::setIoDevice (QIODevice * *ioDevice*)

Does nothing if the ZIP file is open.

Does not reset error code returned by getZipError().

See Also

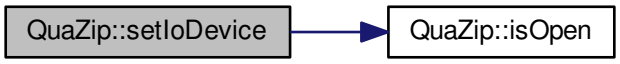
getIoDevice(), getZipName(), setZipName()

Definition at line 365 of file quazip.cpp.

References QuaZipPrivate::ioDevice, isOpen(), p, and QuaZipPrivate::zipName.

```
366 {
367   if(isOpen()) {
368     qWarning("QuaZip::setIoDevice(): ZIP is already open!");
369     return;
370   }
371   p->ioDevice = ioDevice;
372   p->zipName = QString();
373 }
```

Here is the call graph for this function:



12.70.4.39 void QuaZip::setZip64Enabled (bool *zip64*)

Parameters

zip64	If `true`, the zip64 mode is enabled, disabled otherwise.

Once this is enabled, all new files (until the mode is disabled again) will be created in the zip64 mode, thus enabling the ability to write files larger than 4 GB. By default, the zip64 mode is off due to compatibility reasons.

Note that this does not affect the ability to read zip64 archives in any way.

See Also

isZip64Enabled()

Definition at line 770 of file quazip.cpp.

References p, and QuaZipPrivate::zip64.

```
771 {
772     p->zip64 = zip64;
773 }
```

12.70.4.40 void QuaZip::setZipName (const QString & *zipName*)

Does nothing if the ZIP file is open.

Does not reset error code returned by getZipError().

See Also

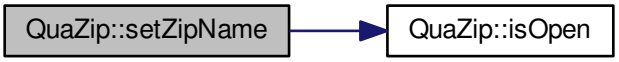
setIoDevice(), getIoDevice(), getZipName()

Definition at line 355 of file quazip.cpp.

References QuaZipPrivate::ioDevice, isOpen(), p, and QuaZipPrivate::zipName.

```
356 {
357   if(isOpen()) {
358     qWarning("QuaZip::setZipName(): ZIP is already open!");
359     return;
360   }
361   p->zipName=zipName;
362   p->ioDevice = NULL;
363 }
```

Here is the call graph for this function:



12.70.5 Friends And Related Function Documentation

12.70.5.1 friend class QuaZipPrivate [friend]

Definition at line 85 of file quazip.h.

12.70.6 Field Documentation

12.70.6.1 QuaZipPrivate* QuaZip::p [private]

Definition at line 129 of file quazip.h.

Referenced by close(), getComment(), getCommentCodec(), getCurrentFileInfo(), getCurrentFileName(), getEntriesCount(), getFileInfoList(), getFileInfoList64(), getFileNameCodec(), getFileNameList(), getIoDevice(), getMode(), getUnzFile(), getZipError(), getZipFile(), getZipName(), goToFirstFile(), goToNextFile(), hasCurrentFile(), isAutoClose(), isDataDescriptorWritingEnabled(), isOpen(), isZip64Enabled(), open(), setAutoClose(), setComment(), setCommentCodec(), setCurrentFile(), setDataDescriptorWritingEnabled(), setFileNameCodec(), setIoDevice(), setZip64Enabled(), setZipName(), and ~QuaZip().

The documentation for this class was generated from the following files:

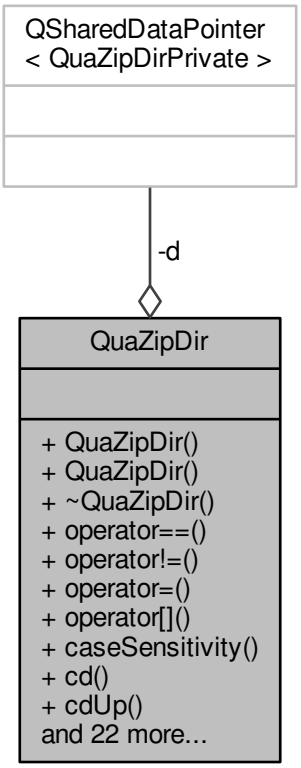
- src/ckb/quazip/quazip.h
- src/ckb/quazip/quazip.cpp

12.71 QuaZipDir Class Reference

Provides ZIP archive navigation.

`#include <src/ckb/quazip/quazipdir.h>`

Collaboration diagram for QuaZipDir:



Public Member Functions

- QuaZipDir (const QuaZipDir &that)

  The copy constructor.
- QuaZipDir (QuaZip ∗zip, const QString &dir=QString())

  Constructs a QuaZipDir instance pointing to the specified directory.
- ∼QuaZipDir ()

  Destructor.
- bool operator== (const QuaZipDir &that)

  The assignment operator.
- bool operator!= (const QuaZipDir &that)

  operator!=
- QuaZipDir & operator= (const QuaZipDir &that)

  operator==
- QString operator[] (int pos) const

  Returns the name of the entry at the specified position.
- QuaZip::CaseSensitivity caseSensitivity () const

  Returns the current case sensitivity mode.
- bool cd (const QString &dirName)

  Changes the 'current' directory.

- bool cdUp ()

  Goes up.
- uint count () const

  Returns the number of entries in the directory.
- QString dirName () const

  Returns the current directory name.
- QList< QuaZipFileInfo > entryInfoList (const QStringList &nameFilters, QDir::Filters filters=QDir::NoFilter, QDir::SortFlags sort=QDir::NoSort) const

  Returns the list of the entries in the directory.
- QList< QuaZipFileInfo > entryInfoList (QDir::Filters filters=QDir::NoFilter, QDir::SortFlags sort=QDir::NoSort) const

  Returns the list of the entries in the directory.
- QList< QuaZipFileInfo64 > entryInfoList64 (const QStringList &nameFilters, QDir::Filters filters=QDir::NoFilter, QDir::SortFlags sort=QDir::NoSort) const

  Returns the list of the entries in the directory with zip64 support.
- QList< QuaZipFileInfo64 > entryInfoList64 (QDir::Filters filters=QDir::NoFilter, QDir::SortFlags sort=QDir::NoSort) const

  Returns the list of the entries in the directory with zip64 support.
- QStringList entryList (const QStringList &nameFilters, QDir::Filters filters=QDir::NoFilter, QDir::SortFlags sort=QDir::NoSort) const

  Returns the list of the entry names in the directory.
- QStringList entryList (QDir::Filters filters=QDir::NoFilter, QDir::SortFlags sort=QDir::NoSort) const

  Returns the list of the entry names in the directory.
- bool exists (const QString &fileName) const

  Returns `true` if the entry with the specified name exists.
- bool exists () const

  Return `true` if the directory pointed by this QuaZipDir exists.
- QString filePath (const QString &fileName) const

  Returns the full path to the specified file.
- QDir::Filters filter ()

  Returns the default filter.
- bool isRoot () const

  Returns if the QuaZipDir points to the root of the archive.
- QStringList nameFilters () const

  Return the default name filter.
- QString path () const

  Returns the path to the current dir.
- QString relativeFilePath (const QString &fileName) const

  Returns the path to the specified file relative to the current dir.
- void setCaseSensitivity (QuaZip::CaseSensitivity caseSensitivity)

  Sets the default case sensitivity mode.
- void setFilter (QDir::Filters filters)

  Sets the default filter.
- void setNameFilters (const QStringList &nameFilters)

  Sets the default name filter.
- void setPath (const QString &path)

  Goes to the specified path.
- void setSorting (QDir::SortFlags sort)

  Sets the default sorting mode.
- QDir::SortFlags sorting () const

  Returns the default sorting mode.

Private Attributes

- QSharedDataPointer
  < QuaZipDirPrivate > d

12.71.1 Detailed Description

This class is modelled after QDir, and is designed to provide similar features for ZIP archives.

The only significant difference from QDir is that the root path is not '/', but an empty string since that's how the file paths are stored in the archive. However, QuaZipDir understands the paths starting with '/'. It is important in a few places:

- In the cd() function.
- In the constructor.
- In the exists() function.
- In the relativePath() function.

Note that since ZIP uses '/' on all platforms, the '\' separator is not supported.

Definition at line 54 of file quazipdir.h.

12.71.2 Constructor & Destructor Documentation

12.71.2.1 QuaZipDir::QuaZipDir (const QuaZipDir & *that*)

Definition at line 50 of file quazipdir.cpp.

Referenced by exists().

```
50                                          :
51     d(that.d)
52 {
53 }
```

Here is the caller graph for this function:

12.71.2.2 QuaZipDir::QuaZipDir (QuaZip * *zip*, const QString & *dir* = QString())

If *dir* is not specified, points to the root of the archive. The same happens if the *dir* is "/".

Definition at line 55 of file quazipdir.cpp.

References d.

```
55                                                  :
56     d(new QuaZipDirPrivate(zip, dir))
57 {
58     if (d->dir.startsWith('/'))
59         d->dir = d->dir.mid(1);
60 }
```

12.71.2.3 QuaZipDir::~QuaZipDir ( )

Definition at line 62 of file quazipdir.cpp.

```
63 {
64 }
```

12.71.3 Member Function Documentation

12.71.3.1 QuaZip::CaseSensitivity QuaZipDir::caseSensitivity ( ) const

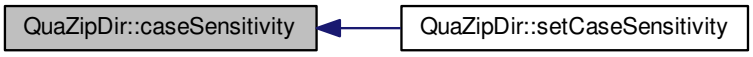
Definition at line 82 of file quazipdir.cpp.

References d.

Referenced by setCaseSensitivity().

```
83 {
84     return d->caseSensitivity;
85 }
```

Here is the caller graph for this function:



12.71.3.2 bool QuaZipDir::cd (const QString & *dirName*)

If the path starts with '/', it is interpreted as an absolute path from the root of the archive. Otherwise, it is interpreted as a path relative to the current directory as was set by the previous cd() or the constructor.

Note that the subsequent path() call will not return a path starting with '/' in all cases.

Definition at line 87 of file quazipdir.cpp.

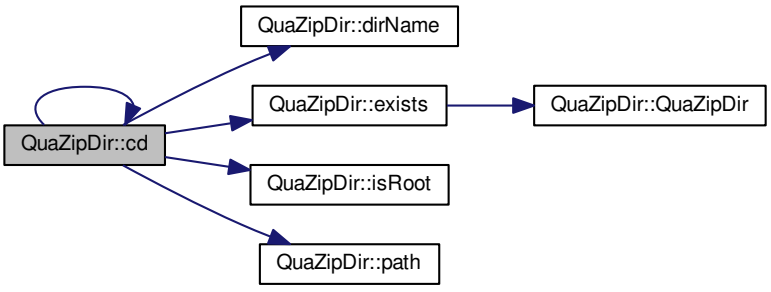
References cd(), d, dirName(), exists(), isRoot(), and path().

Referenced by cd(), and cdUp().

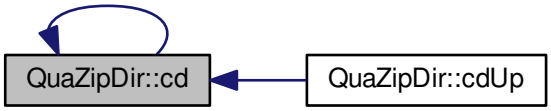
```
88 {
89     if (directoryName == "/") {
90         d->dir = "";
91         return true;
92     }
93     QString dirName = directoryName;
94     if (dirName.endsWith('/'))
95         dirName.chop(1);
96     if (dirName.contains('/')) {
97         QuaZipDir dir(*this);
98         if (dirName.startsWith('/')) {
99 #ifdef QUAZIP_QUAZIPDIR_DEBUG
100             qDebug("QuaZipDir::cd(%s): going to /",
101                    dirName.toUtf8().constData());
102 #endif
103             if (!dir.cd("/"))
104                 return false;
105         }
106         QStringList path = dirName.split('/', QString::SkipEmptyParts);
107         for (QStringList::const_iterator i = path.constBegin();
108                 i != path.end();
```

```
109                 ++i) {
110             const QString &step = *i;
111 #ifdef QUAZIP_QUAZIPDIR_DEBUG
112             qDebug("QuaZipDir::cd(%s): going to %s",
113                    dirName.toUtf8().constData(),
114                    step.toUtf8().constData());
115 #endif
116             if (!dir.cd(step))
117                 return false;
118         }
119         d->dir = dir.path();
120         return true;
121     } else { // no '/'
122         if (dirName == ".") {
123             return true;
124         } else if (dirName == "..") {
125             if (isRoot()) {
126                 return false;
127             } else {
128                 int slashPos = d->dir.lastIndexOf('/');
129                 if (slashPos == -1) {
130                     d->dir = "";
131                 } else {
132                     d->dir = d->dir.left(slashPos);
133                 }
134                 return true;
135             }
136         } else { // a simple subdirectory
137             if (exists(dirName)) {
138                 if (isRoot())
139                     d->dir = dirName;
140                 else
141                     d->dir += "/" + dirName;
142                 return true;
143             } else {
144                 return false;
145             }
146         }
147     }
148 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



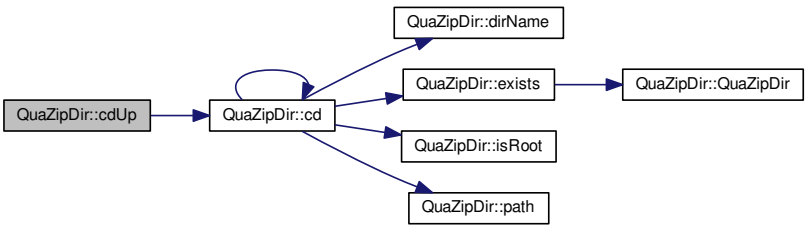
12.71.3.3 bool QuaZipDir::cdUp ( )

Definition at line 150 of file quazipdir.cpp.

References cd().

```
151 {
152     return cd("..");
153 }
```

Here is the call graph for this function:



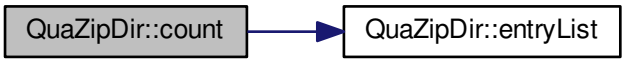
12.71.3.4 uint QuaZipDir::count ( ) const

Definition at line 155 of file quazipdir.cpp.

References entryList().

```
156 {
157     return entryList().count();
158 }
```

Here is the call graph for this function:



12.71.3.5 QString QuaZipDir::dirName ( ) const

The name doesn't include the path.

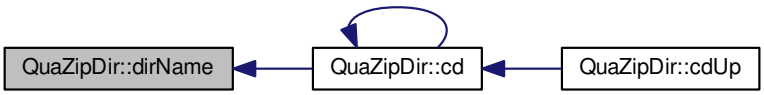
Definition at line 160 of file quazipdir.cpp.

References d.

Referenced by cd().

```
161 {
162     return QDir(d->dir).dirName();
163 }
```

Here is the caller graph for this function:



12.71.3.6 QList< QuaZipFileInfo > QuaZipDir::entryInfoList (const QStringList & *nameFilters*, QDir::Filters *filters* = QDir::NoFilter, QDir::SortFlags *sort* = QDir::NoSort) const

Parameters

nameFilters	The list of file patterns to list, uses the same syntax as QDir.
filters	The entry type filters, only Files and Dirs are accepted.
sort	Sorting mode.

Definition at line 401 of file quazipdir.cpp.

References d.

Referenced by entryInfoList().

```
403 {
404     QList<QuaZipFileInfo> result;
405     if (d->entryInfoList(nameFilters, filters, sort, result))
406         return result;
407     else
408         return QList<QuaZipFileInfo>();
409 }
```

Here is the caller graph for this function:

12.71.3.7 QList< QuaZipFileInfo > QuaZipDir::entryInfoList (QDir::Filters *filters* = QDir::NoFilter, QDir::SortFlags *sort* = QDir::NoSort) const

This is an overloaded member function, provided for convenience. It differs from the above function only in what argument(s) it accepts.

The same as entryInfoList(QStringList(), filters, sort).

Definition at line 411 of file quazipdir.cpp.

References entryInfoList().

```
413 {
414     return entryInfoList(QStringList(), filters, sort);
415 }
```

Here is the call graph for this function:

12.71.3.8 QList< QuaZipFileInfo64 > QuaZipDir::entryInfoList64 (const QStringList & *nameFilters*, QDir::Filters *filters* = QDir::NoFilter, QDir::SortFlags *sort* = QDir::NoSort) const

Parameters

nameFilters	The list of file patterns to list, uses the same syntax as QDir.
filters	The entry type filters, only Files and Dirs are accepted.
sort	Sorting mode.

Definition at line 417 of file quazipdir.cpp.

References d.

Referenced by entryInfoList64().

```
419 {
420     QList<QuaZipFileInfo64> result;
421     if (d->entryInfoList(nameFilters, filters, sort, result))
422         return result;
423     else
424         return QList<QuaZipFileInfo64>();
425 }
```

Here is the caller graph for this function:

12.71.3.9 QList< QuaZipFileInfo64 > QuaZipDir::entryInfoList64 (QDir::Filters *filters* = QDir::NoFilter, QDir::SortFlags *sort* = QDir::NoSort) const

This is an overloaded member function, provided for convenience. It differs from the above function only in what argument(s) it accepts.

The same as entryInfoList64(QStringList(), filters, sort).

Definition at line 427 of file quazipdir.cpp.

References entryInfoList64().

```
429 {
430     return entryInfoList64(QStringList(), filters, sort);
431 }
```

Here is the call graph for this function:

12.71.3.10 QStringList QuaZipDir::entryList (const QStringList & *nameFilters*, QDir::Filters *filters* = QDir::NoFilter, QDir::SortFlags *sort* = QDir::NoSort) const

The same as entryInfoList(nameFilters, filters, sort), but only returns entry names.

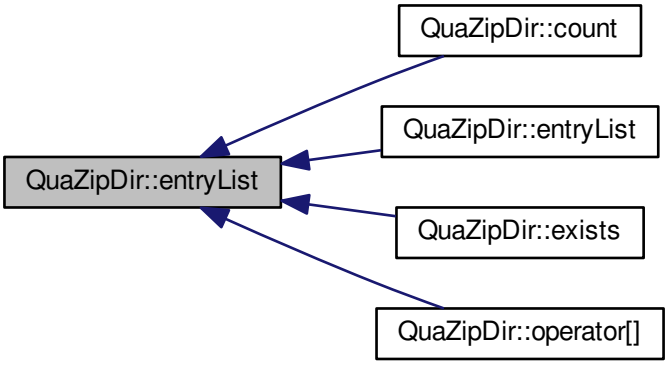
Definition at line 433 of file quazipdir.cpp.

References d.

Referenced by count(), entryList(), exists(), and operator[]().

```
435 {
436     QStringList result;
437     if (d->entryInfoList(nameFilters, filters, sort, result))
438         return result;
439     else
440         return QStringList();
441 }
```

Here is the caller graph for this function:



12.71.3.11 QStringList QuaZipDir::entryList (QDir::Filters *filters* = `QDir::NoFilter`, QDir::SortFlags *sort* = `QDir::NoSort`) const

This is an overloaded member function, provided for convenience. It differs from the above function only in what argument(s) it accepts.

The same as entryList(QStringList(), filters, sort).

Definition at line 443 of file quazipdir.cpp.

References entryList().

```
445 {
446     return entryList(QStringList(), filters, sort);
447 }
```

Here is the call graph for this function:

12.71.3.12 bool QuaZipDir::exists (const QString & *fileName*) const

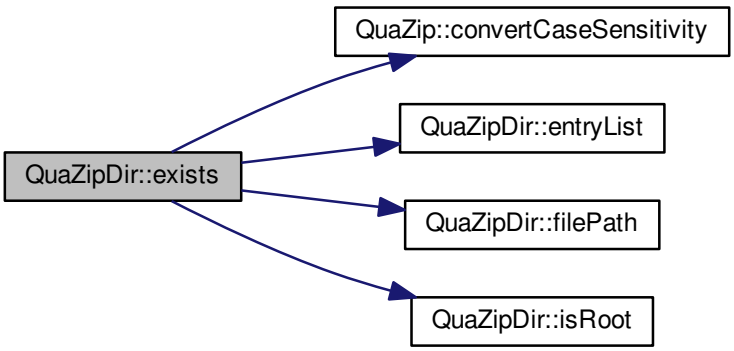
The ".." is considered to exist if the current directory is not root. The "." and "/" are considered to always exist. Paths starting with "/" are relative to the archive root, other paths are relative to the current dir.

Definition at line 449 of file quazipdir.cpp.

References QuaZip::convertCaseSensitivity(), d, entryList(), filePath(), and isRoot().

```
450 {
451     if (filePath == "/" || filePath.isEmpty())
452         return true;
453     QString fileName = filePath;
454     if (fileName.endsWith('/'))
455         fileName.chop(1);
456     if (fileName.contains('/')) {
457         QFileInfo fileInfo(fileName);
458 #ifdef QUAZIP_QUAZIPDIR_DEBUG
459         qDebug("QuaZipDir::exists(): fileName=%s, fileInfo.fileName()=%s, "
460                 "fileInfo.path()=%s", fileName.toUtf8().constData(),
461                 fileInfo.fileName().toUtf8().constData(),
462                 fileInfo.path().toUtf8().constData());
463 #endif
464         QuaZipDir dir(*this);
465         return dir.cd(fileInfo.path()) && dir.exists(fileInfo.fileName());
466     } else {
467         if (fileName == "..") {
468             return !isRoot();
469         } else if (fileName == ".") {
470             return true;
471         } else {
472             QStringList entries = entryList(QDir::AllEntries, QDir::NoSort);
473 #ifdef QUAZIP_QUAZIPDIR_DEBUG
474             qDebug("QuaZipDir::exists(): looking for %s",
475                     fileName.toUtf8().constData());
476             for (QStringList::const_iterator i = entries.constBegin();
477                     i != entries.constEnd();
478                     ++i) {
479                 qDebug("QuaZipDir::exists(): entry: %s",
480                         i->toUtf8().constData());
481             }
482 #endif
483             Qt::CaseSensitivity cs = QuaZip::convertCaseSensitivity(
484                     d->caseSensitivity);
485             if (filePath.endsWith('/')) {
486                 return entries.contains(filePath, cs);
487             } else {
488                 return entries.contains(fileName, cs)
489                     || entries.contains(fileName + "/", cs);
490             }
491         }
492     }
493 }
```

Here is the call graph for this function:



12.71.3.13 bool QuaZipDir::exists ( ) const

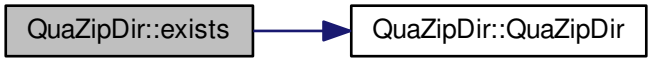
Definition at line 495 of file quazipdir.cpp.

References d, and QuaZipDir().

Referenced by cd().

```
496 {
497     return QuaZipDir(d->zip).exists(d->dir);
498 }
```

Here is the call graph for this function:



Here is the caller graph for this function:

12.71.3.14 QString QuaZipDir::filePath (const QString & *fileName*) const

Doesn't check if the file actually exists.

Definition at line 500 of file quazipdir.cpp.

References d.

Referenced by exists().

```
501 {
502     return QDir(d->dir).filePath(fileName);
503 }
```

Here is the caller graph for this function:

12.71.3.15 QDir::Filters QuaZipDir::filter ( )

Definition at line 505 of file quazipdir.cpp.

References d.

```
506 {
507     return d->filter;
508 }
```

12.71.3.16 bool QuaZipDir::isRoot ( ) const

Not that the root path is the empty string, not '/'.

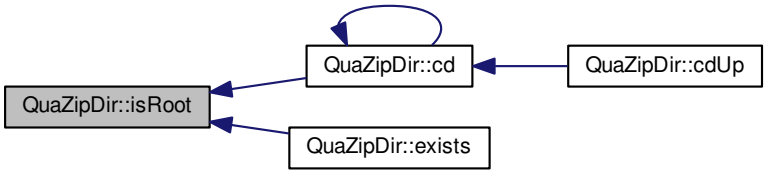
Definition at line 510 of file quazipdir.cpp.

References d.

Referenced by cd(), and exists().

```
511 {
512     return d->simplePath().isEmpty();
513 }
```

Here is the caller graph for this function:



12.71.3.17 QStringList QuaZipDir::nameFilters ( ) const

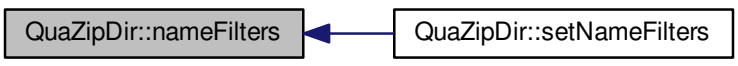
Definition at line 515 of file quazipdir.cpp.

References d.

Referenced by setNameFilters().

```
516 {
517     return d->nameFilters;
518 }
```

Here is the caller graph for this function:



12.71.3.18 bool QuaZipDir::operator!= (const QuaZipDir & *that*) [inline]

Returns

`true` if either this and *that* use different QuaZip instances or if they point to different directories.

Definition at line 75 of file quazipdir.h.

```
75 {return !operator==(that);}
```

12.71.3.19 QuaZipDir & QuaZipDir::operator= (const QuaZipDir & *that*)

Returns

`true` if both this and *that* use the same QuaZip instance and point to the same directory.

Definition at line 71 of file quazipdir.cpp.

References d.

```
72 {
73     this->d = that.d;
74     return *this;
75 }
```

12.71.3.20 bool QuaZipDir::operator== (const QuaZipDir & *that*)

Definition at line 66 of file quazipdir.cpp.

References d.

```
67 {
68     return d->zip == that.d->zip && d->dir == that.d->dir;
69 }
```

12.71.3.21 QString QuaZipDir::operator[] (int *pos*) const

Definition at line 77 of file quazipdir.cpp.

References entryList().

```
78 {
79     return entryList().at(pos);
80 }
```

Here is the call graph for this function:

12.71.3.22 QString QuaZipDir::path ( ) const

The path never starts with '/', and the root path is an empty string.

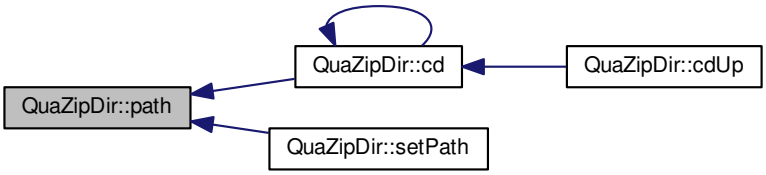
Definition at line 520 of file quazipdir.cpp.

References d.

Referenced by cd(), and setPath().

```
521 {
522     return d->dir;
523 }
```

Here is the caller graph for this function:



12.71.3.23 QString QuaZipDir::relativeFilePath (const QString & *fileName*) const

This function is mostly useless, provided only for the sake of completeness.

Parameters

fileName	The path to the file, should start with "/" if relative to the archive root.

Returns

Path relative to the current dir.

Definition at line 525 of file quazipdir.cpp.

References d.

```
526 {
527     return QDir("/" + d->dir).relativeFilePath(fileName);
528 }
```

12.71.3.24 void QuaZipDir::setCaseSensitivity (QuaZip::CaseSensitivity *caseSensitivity*)

Definition at line 530 of file quazipdir.cpp.

References caseSensitivity(), and d.

```
531 {
532     d->caseSensitivity = caseSensitivity;
533 }
```

Here is the call graph for this function:

12.71.3.25 void QuaZipDir::setFilter (QDir::Filters *filters*)

Definition at line 535 of file quazipdir.cpp.

References d.

```
536 {
537     d->filter = filters;
538 }
```

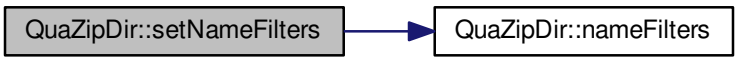
12.71.3.26 void QuaZipDir::setNameFilters (const QStringList & *nameFilters*)

Definition at line 540 of file quazipdir.cpp.

References d, and nameFilters().

```
541 {
542     d->nameFilters = nameFilters;
543 }
```

Here is the call graph for this function:



12.71.3.27 void QuaZipDir::setPath (const QString & *path*)

The difference from cd() is that this function never checks if the path actually exists and doesn't use relative paths, so it's possible to go to the root directory with setPath("").

Note that this function still chops the trailing and/or leading '/' and treats a single '/' as the root path (path() will still return an empty string).

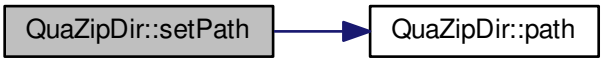
Definition at line 545 of file quazipdir.cpp.

References d, and path().

```
546 {
547     QString newDir = path;
```

```
548     if (newDir == "/") {
549         d->dir = "";
550     } else {
551         if (newDir.endsWith('/'))
552             newDir.chop(1);
553         if (newDir.startsWith('/'))
554             newDir = newDir.mid(1);
555         d->dir = newDir;
556     }
557 }
```

Here is the call graph for this function:



12.71.3.28 void QuaZipDir::setSorting (QDir::SortFlags *sort*)

Definition at line 559 of file quazipdir.cpp.

References d.

```
560 {
561     d->sorting = sort;
562 }
```

12.71.3.29 QDir::SortFlags QuaZipDir::sorting ( ) const

Definition at line 564 of file quazipdir.cpp.

References d.

```
565 {
566     return d->sorting;
567 }
```

12.71.4 Field Documentation

12.71.4.1 QSharedDataPointer<QuaZipDirPrivate> QuaZipDir::d [private]

Definition at line 56 of file quazipdir.h.

Referenced by caseSensitivity(), cd(), dirName(), entryInfoList(), entryInfoList64(), entryList(), exists(), filePath(), filter(), isRoot(), nameFilters(), operator=(), operator==(), path(), QuaZipDir(), relativeFilePath(), setCaseSensitivity(), setFilter(), setNameFilters(), setPath(), setSorting(), and sorting().

The documentation for this class was generated from the following files:

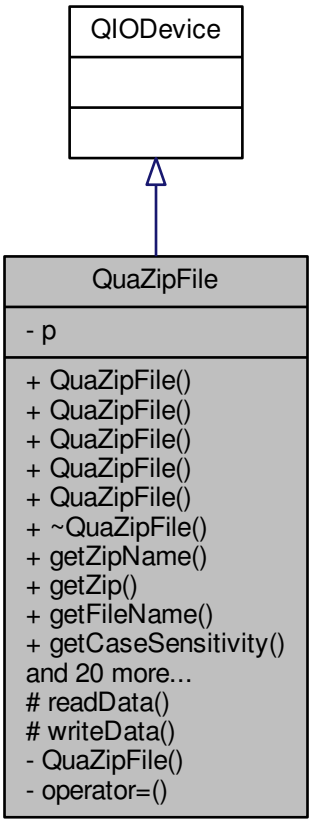
- src/ckb/quazip/quazipdir.h
- src/ckb/quazip/quazipdir.cpp

12.72 QuaZipFile Class Reference

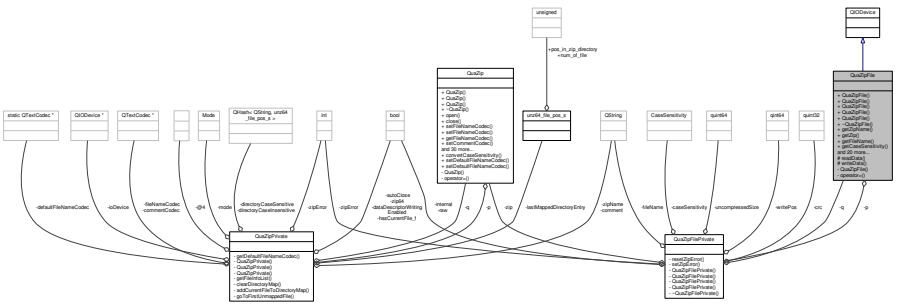
A file inside ZIP archive.

`#include <quazip/quazipfile.h>`

Inheritance diagram for QuaZipFile:



Collaboration diagram for QuaZipFile:



Public Member Functions

- QuaZipFile ()

  Constructs a QuaZipFile instance.
- QuaZipFile (QObject *parent)

  Constructs a QuaZipFile instance.
- QuaZipFile (const QString &zipName, QObject *parent=NULL)

  Constructs a QuaZipFile instance.
- QuaZipFile (const QString &zipName, const QString &fileName, QuaZip::CaseSensitivity cs=QuaZip::csDefault, QObject *parent=NULL)

  Constructs a QuaZipFile instance.
- QuaZipFile (QuaZip *zip, QObject *parent=NULL)

  Constructs a QuaZipFile instance.
- virtual ~QuaZipFile ()

  Destroys a QuaZipFile instance.
- QString getZipName () const

  Returns the ZIP archive file name.
- QuaZip * getZip () const

  Returns a pointer to the associated QuaZip object.
- QString getFileName () const

  Returns file name.
- QuaZip::CaseSensitivity getCaseSensitivity () const

  Returns case sensitivity of the file name.
- QString getActualFileName () const

  Returns the actual file name in the archive.
- void setZipName (const QString &zipName)

  Sets the ZIP archive file name.
- bool isRaw () const

  Returns `true` if the file was opened in raw mode.
- void setZip (QuaZip *zip)

  Binds to the existing QuaZip instance.
- void setFileName (const QString &fileName, QuaZip::CaseSensitivity cs=QuaZip::csDefault)

  Sets the file name.
- virtual bool open (OpenMode mode)

  Opens a file for reading.
- bool open (OpenMode mode, const char *password)

  Opens a file for reading.
- bool open (OpenMode mode, int *method, int *level, bool raw, const char *password=NULL)

  Opens a file for reading.
- bool open (OpenMode mode, const QuaZipNewInfo &info, const char *password=NULL, quint32 crc=0, int method=Z_DEFLATED, int level=Z_DEFAULT_COMPRESSION, bool raw=false, int windowBits=-MAX_WBITS, int memLevel=MAX_MEM_LEVEL, int strategy=Z_DEFAULT_STRATEGY)

  Opens a file for writing.
- virtual bool isSequential () const

  Returns `true`, but beware!
- virtual qint64 pos () const

  Returns current position in the file.
- virtual bool atEnd () const

  Returns `true` if the end of file was reached.
- virtual qint64 size () const

  Returns file size.

- qint64 csize () const

  Returns compressed file size.
- qint64 usize () const

  Returns uncompressed file size.
- bool getFileInfo (QuaZipFileInfo *info)

  Gets information about current file.
- bool getFileInfo (QuaZipFileInfo64 *info)

  Gets information about current file with zip64 support.
- virtual void close ()

  Closes the file.
- int getZipError () const

  Returns the error code returned by the last ZIP/UNZIP API call.
- virtual qint64 bytesAvailable () const

  Returns the number of bytes available for reading.

Protected Member Functions

- qint64 readData (char *data, qint64 maxSize)

  Implementation of the QIODevice::readData().
- qint64 writeData (const char *data, qint64 maxSize)

  Implementation of the QIODevice::writeData().

Private Member Functions

- QuaZipFile (const QuaZipFile &that)
- QuaZipFile & operator= (const QuaZipFile &that)

Private Attributes

- QuaZipFilePrivate * p

Friends

- class QuaZipFilePrivate

12.72.1 Detailed Description

This is the most interesting class. Not only it provides C++ interface to the ZIP/UNZIP package, but also integrates it with Qt by subclassing QIODevice. This makes possible to access files inside ZIP archive using QTextStream or QDataStream, for example. Actually, this is the main purpose of the whole QuaZIP library.

You can either use existing QuaZip instance to create instance of this class or pass ZIP archive file name to this class, in which case it will create internal QuaZip object. See constructors' descriptions for details. Writing is only possible with the existing instance.

Note that due to the underlying library's limitation it is not possible to use multiple QuaZipFile instances to open several files in the same archive at the same time. If you need to write to multiple files in parallel, then you should write to temporary files first, then pack them all at once when you have finished writing. If you need to read multiple files inside the same archive in parallel, you should extract them all into a temporary directory first.

12.72.2 Sequential or random-access?

At the first thought, QuaZipFile has fixed size, the start and the end and should be therefore considered random-access device. But there is one major obstacle to making it random-access: ZIP/UNZIP API does not support seek() operation and the only way to implement it is through reopening the file and re-reading to the required position, but this is prohibitively slow.

Therefore, QuaZipFile is considered to be a sequential device. This has advantage of availability of the ungetChar() operation (QIODevice does not implement it properly for non-sequential devices unless they support seek()). Disadvantage is a somewhat strange behaviour of the size() and pos() functions. This should be kept in mind while using this class.

Definition at line 74 of file quazipfile.h.

12.72.3 Constructor & Destructor Documentation

12.72.3.1 QuaZipFile::QuaZipFile (const QuaZipFile & *that*) [private]

12.72.3.2 QuaZipFile::QuaZipFile ( )

You should use setZipName() and setFileName() or setZip() before trying to call open() on the constructed object.

Definition at line 108 of file quazipfile.cpp.

```
108                       :
109   p(new QuaZipFilePrivate(this))
110 {
111 }
```

12.72.3.3 QuaZipFile::QuaZipFile (QObject * *parent*)

parent argument specifies this object's parent object.

You should use setZipName() and setFileName() or setZip() before trying to call open() on the constructed object.

Definition at line 113 of file quazipfile.cpp.

```
113                                       :
114   QIODevice(parent),
115   p(new QuaZipFilePrivate(this))
116 {
117 }
```

12.72.3.4 QuaZipFile::QuaZipFile (const QString & *zipName*, QObject * *parent* = NULL)

parent argument specifies this object's parent object and *zipName* specifies ZIP archive file name.

You should use setFileName() before trying to call open() on the constructed object.

QuaZipFile constructed by this constructor can be used for read only access. Use QuaZipFile(QuaZip*,QObject*) for writing.

Definition at line 119 of file quazipfile.cpp.

```
119                                                           :
120   QIODevice(parent),
121   p(new QuaZipFilePrivate(this, zipName))
122 {
123 }
```

12.72.3.5 QuaZipFile::QuaZipFile (const QString & *zipName*, const QString & *fileName*, QuaZip::CaseSensitivity *cs* = QuaZip::csDefault, QObject * *parent* = NULL)

parent argument specifies this object's parent object, *zipName* specifies ZIP archive file name and *fileName* and *cs* specify a name of the file to open inside archive.

QuaZipFile constructed by this constructor can be used for read only access. Use QuaZipFile(QuaZip*,QObject*) for writing.

See Also

QuaZip::setCurrentFile()

Definition at line 125 of file quazipfile.cpp.

```
126                                                :
127     QIODevice(parent),
128     p(new QuaZipFilePrivate(this, zipName, fileName, cs))
129 {
130 }
```

12.72.3.6 QuaZipFile::QuaZipFile (QuaZip * *zip*, QObject * *parent* = NULL)

parent argument specifies this object's parent object.

zip is the pointer to the existing QuaZip object. This QuaZipFile object then can be used to read current file in the *zip* or to write to the file inside it.

Warning

Using this constructor for reading current file can be tricky. Let's take the following example:

```
QuaZip zip("archive.zip");
zip.open(QuaZip::mdUnzip);
zip.setCurrentFile("file-in-archive");
QuaZipFile file(&zip);
file.open(QIODevice::ReadOnly);
// ok, now we can read from the file
file.read(somewhere, some);
zip.setCurrentFile("another-file-in-archive"); // oops...
QuaZipFile anotherFile(&zip);
anotherFile.open(QIODevice::ReadOnly);
anotherFile.read(somewhere, some); // this is still ok...
file.read(somewhere, some); // and this is NOT
```

So, what exactly happens here? When we change current file in the `zip` archive, `file` that references it becomes invalid (actually, as far as I understand ZIP/UNZIP sources, it becomes closed, but QuaZipFile has no means to detect it).

Summary: do not close `zip` object or change its current file as long as QuaZipFile is open. Even better - use another constructors which create internal QuaZip instances, one per object, and therefore do not cause unnecessary trouble. This constructor may be useful, though, if you already have a QuaZip instance and do not want to access several files at once. Good example:

```
QuaZip zip("archive.zip");
zip.open(QuaZip::mdUnzip);
// first, we need some information about archive itself
QByteArray comment=zip.getComment();
// and now we are going to access files inside it
QuaZipFile file(&zip);
for(bool more=zip.goToFirstFile(); more; more=zip.goToNextFile()) {
  file.open(QIODevice::ReadOnly);
  // do something cool with file here
  file.close(); // do not forget to close!
}
zip.close();
```

Definition at line 132 of file quazipfile.cpp.

```
132                                                 :
133   QIODevice(parent),
134   p(new QuaZipFilePrivate(this, zip))
135 {
136 }
```

12.72.3.7 QuaZipFile::~QuaZipFile ( ) [virtual]

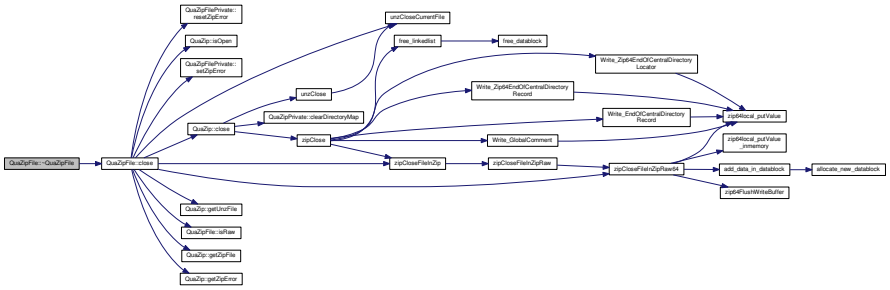
Closes file if open, destructs internal QuaZip object (if it exists and *is* internal, of course).

Definition at line 138 of file quazipfile.cpp.

References close(), and p.

```
139 {
140   if (isOpen())
141     close();
142   delete p;
143 }
```

Here is the call graph for this function:



12.72.4 Member Function Documentation

12.72.4.1 bool QuaZipFile::atEnd ( ) const [virtual]

This function returns `false` in the case of error. This means that you called this function on either not open file, or a file in the not open archive or even on a QuaZipFile instance that does not even have QuaZip instance associated. Do not do that because there is no means to determine whether `false` is returned because of error or because end of file was reached. Well, on the other side you may interpret `false` return value as "there is no file open to check for end of file and there is no end of file therefore".

When writing, this function always returns `true` (because you are always writing to the end of file).

Error code returned by getZipError() is not affected by this function call.

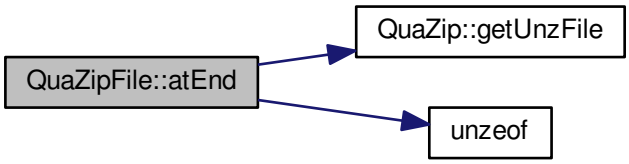
Definition at line 361 of file quazipfile.cpp.

References QuaZip::getUnzFile(), p, unzeof(), and QuaZipFilePrivate::zip.

```
362 {
363   if(p->zip==NULL) {
364     qWarning("QuaZipFile::atEnd(): call setZipName() or setZip() first");
365     return false;
366   }
367   if(!isOpen()) {
368     qWarning("QuaZipFile::atEnd(): file is not open");
369     return false;
370   }
371   if(openMode()&ReadOnly)
372       // the same problem as with pos()
373     return QIODevice::bytesAvailable() == 0
374         && unzeof(p->zip->getUnzFile())==1;
```

```
375   else
376     return true;
377 }
```

Here is the call graph for this function:



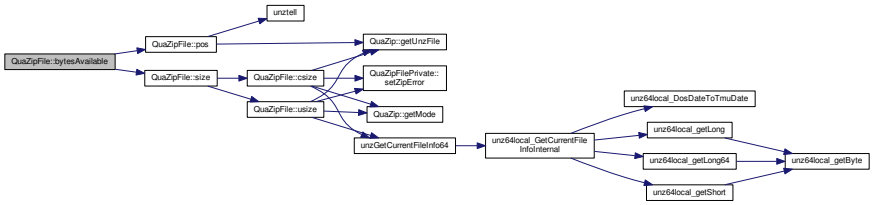
12.72.4.2 qint64 QuaZipFile::bytesAvailable ( ) const [virtual]

Definition at line 499 of file quazipfile.cpp.

References pos(), and size().

```
500 {
501     return size() - pos();
502 }
```

Here is the call graph for this function:



12.72.4.3 void QuaZipFile::close ( ) [virtual]

Call getZipError() to determine if the close was successful.

Definition at line 432 of file quazipfile.cpp.

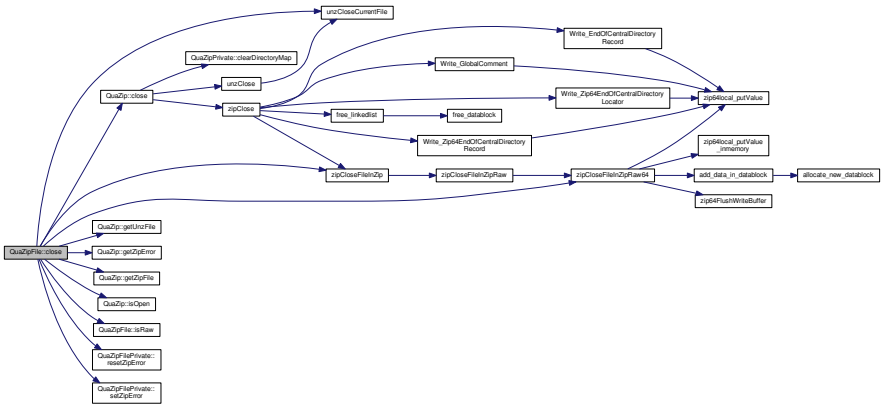
References QuaZip::close(), QuaZipFilePrivate::crc, QuaZip::getUnzFile(), QuaZip::getZipError(), QuaZip::getZipFile(), QuaZipFilePrivate::internal, QuaZip::isOpen(), isRaw(), p, QuaZipFilePrivate::resetZipError(), QuaZipFilePrivate::setZipError(), QuaZipFilePrivate::uncompressedSize, UNZ_OK, unzCloseCurrentFile(), QuaZipFilePrivate::zip, zipCloseFileInZip(), zipCloseFileInZipRaw64(), and QuaZipFilePrivate::zipError.

Referenced by JlCompress::compressFile(), JlCompress::compressSubDir(), JlCompress::extractFile(), and ~QuaZipFile().

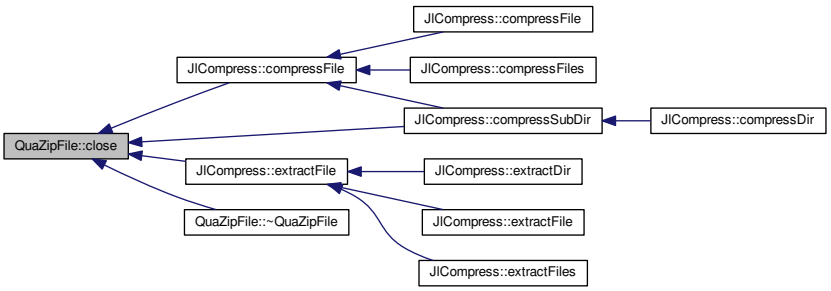
```
433 {
434   p->resetZipError();
435   if(p->zip==NULL||!p->zip->isOpen()) return;
```

```
436   if(!isOpen()) {
437     qWarning("QuaZipFile::close(): file isn't open");
438     return;
439   }
440   if(openMode()&ReadOnly)
441     p->setZipError(unzCloseCurrentFile(p->zip->
      getUnzFile()));
442   else if(openMode()&WriteOnly)
443     if(isRaw()) p->setZipError(zipCloseFileInZipRaw64(
      p->zip->getZipFile(), p->uncompressedSize, p->
      crc));
444     else p->setZipError(zipCloseFileInZip(p->zip->
      getZipFile()));
445   else {
446     qWarning("Wrong open mode: %d", (int)openMode());
447     return;
448   }
449   if(p->zipError==UNZ_OK) setOpenMode(QIODevice::NotOpen);
450   else return;
451   if(p->internal) {
452     p->zip->close();
453     p->setZipError(p->zip->getZipError());
454   }
455 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.72.4.4 qint64 QuaZipFile::csize ( ) const

Equivalent to calling getFileInfo() and then getting compressedSize field, but more convenient and faster.

File must be open for reading before calling this function.

Returns -1 on error, call getZipError() to get error code.

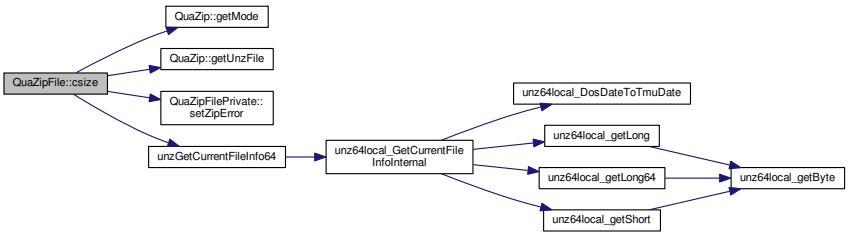
Definition at line 391 of file quazipfile.cpp.

References unz_file_info64_s::compressed_size, QuaZip::getMode(), QuaZip::getUnzFile(), QuaZip::mdUnzip, p, QuaZipFilePrivate::setZipError(), UNZ_OK, unzGetCurrentFileInfo64(), QuaZipFilePrivate::zip, and QuaZipFilePrivate::zipError.

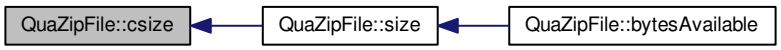
Referenced by size().

```
392 {
393   unz_file_info64 info_z;
394   p->setZipError(UNZ_OK);
395   if(p->zip==NULL||p->zip->getMode()!=QuaZip::mdUnzip) return -1;
396   p->setZipError(unzGetCurrentFileInfo64(p->
      zip->getUnzFile(), &info_z, NULL, 0, NULL, 0, NULL, 0));
397   if(p->zipError!=UNZ_OK)
398     return -1;
399   return info_z.compressed_size;
400 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.72.4.5 QString QuaZipFile::getActualFileName ( ) const

This is *not* a ZIP archive file name, but a name of file inside archive. It is not necessary the same name that you have passed to the QuaZipFile(const QString&,const QString&,QuaZip::CaseSensitivity,QObject*), setFileName() or QuaZip::setCurrentFile() - this is the real file name inside archive, so it may differ in case if the file name search was case-insensitive.

Equivalent to calling getCurrentFileName() on the associated QuaZip object. Returns null string if there is no associated QuaZip object or if it does not have a current file yet. And this is the case if you called setFileName() but did not open the file yet. So this is perfectly fine:

```
QuaZipFile file("somezip.zip");
file.setFileName("somefile");
QString name=file.getName(); // name=="somefile"
QString actual=file.getActualFileName(); // actual is null string
file.open(QIODevice::ReadOnly);
QString actual=file.getActualFileName(); // actual can be "SoMeFiLe" on Windows
```

See Also

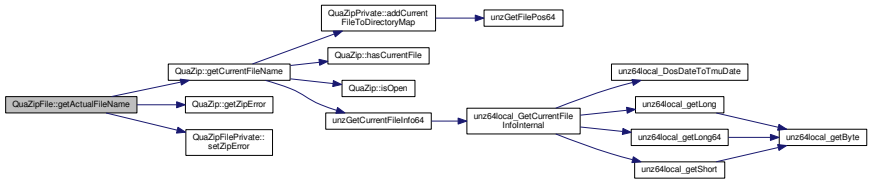
getZipName(), getFileName(), QuaZip::CaseSensitivity

Definition at line 155 of file quazipfile.cpp.

References QuaZip::getCurrentFileName(), QuaZip::getZipError(), p, QuaZipFilePrivate::setZipError(), UNZ_OK, and QuaZipFilePrivate::zip.

```
156 {
157   p->setZipError(UNZ_OK);
158   if (p->zip == NULL || (openMode() & WriteOnly))
159     return QString();
160   QString name=p->zip->getCurrentFileName();
161   if(name.isNull())
162     p->setZipError(p->zip->getZipError());
163   return name;
164 }
```

Here is the call graph for this function:



12.72.4.6 QuaZip::CaseSensitivity QuaZipFile::getCaseSensitivity ( ) const

This function returns case sensitivity argument you passed to this object either by using QuaZipFile(const QString&,const QString&,QuaZip::CaseSensitivity,QObject*) or by calling setFileName().

Returns unpredictable value if getFileName() returns null string (this is the case when you did not used setFileName() or constructor above).

See Also

getFileName

Definition at line 484 of file quazipfile.cpp.

References QuaZipFilePrivate::caseSensitivity, and p.

```
485 {
486   return p->caseSensitivity;
487 }
```

12.72.4.7 bool QuaZipFile::getFileInfo (QuaZipFileInfo * *info*)

This function does the same thing as calling QuaZip::getCurrentFileInfo() on the associated QuaZip object, but you can not call getCurrentFileInfo() if the associated QuaZip is internal (because you do not have access to it), while you still can call this function in that case.

File must be open for reading before calling this function.

Returns

`false` in the case of an error.

This function doesn't support zip64, but will still work fine on zip64 archives if file sizes are below 4 GB, otherwise the values will be set as if converted using QuaZipFileInfo64::toQuaZipFileInfo().

See Also

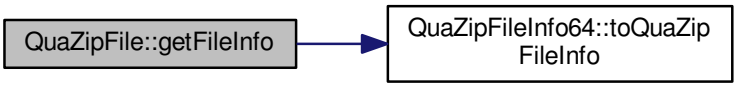
getFileInfo(QuaZipFileInfo64*)

Definition at line 413 of file quazipfile.cpp.

References QuaZipFileInfo64::toQuaZipFileInfo().

```
414 {
415     QuaZipFileInfo64 info64;
416     if (getFileInfo(&info64)) {
417         info64.toQuaZipFileInfo(*info);
418         return true;
419     } else {
420         return false;
421     }
422 }
```

Here is the call graph for this function:



12.72.4.8 bool QuaZipFile::getFileInfo (QuaZipFileInfo64 * *info*)

This is an overloaded member function, provided for convenience. It differs from the above function only in what argument(s) it accepts.

See Also

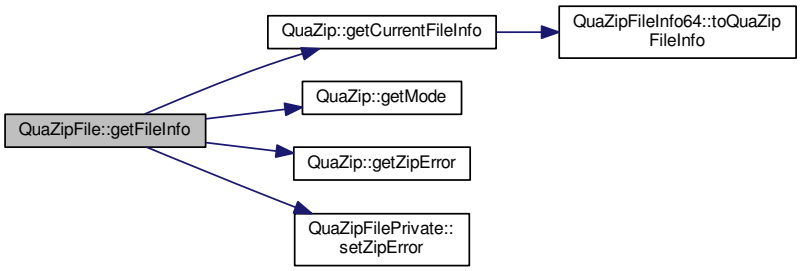
getFileInfo(QuaZipFileInfo*)

Definition at line 424 of file quazipfile.cpp.

References QuaZip::getCurrentFileInfo(), QuaZip::getMode(), QuaZip::getZipError(), QuaZip::mdUnzip, p, QuaZipFilePrivate::setZipError(), UNZ_OK, QuaZipFilePrivate::zip, and QuaZipFilePrivate::zipError.

```
425 {
426     if(p->zip==NULL||p->zip->getMode()!=QuaZip::mdUnzip) return false;
427     p->zip->getCurrentFileInfo(info);
428     p->setZipError(p->zip->getZipError());
429     return p->zipError==UNZ_OK;
430 }
```

Here is the call graph for this function:



12.72.4.9 QString QuaZipFile::getFileName ( ) const

This function returns file name you passed to this object either by using QuaZipFile(const QString&,const QString&,-QuaZip::CaseSensitivity,QObject*) or by calling setFileName(). Real name of the file may differ in case if you used case-insensitivity.

Returns null string if there is no file name set yet. This is the case when this QuaZipFile operates on the existing QuaZip object (constructor QuaZipFile(QuaZip*,QObject*) or setZip() was used).

See Also

getActualFileName

Definition at line 479 of file quazipfile.cpp.

References QuaZipFilePrivate::fileName, and p.

```
480 {
481   return p->fileName;
482 }
```

12.72.4.10 QuaZip * QuaZipFile::getZip ( ) const

Returns NULL if there is no associated QuaZip or it is internal (so you will not mess with it).

Definition at line 150 of file quazipfile.cpp.

References QuaZipFilePrivate::internal, p, and QuaZipFilePrivate::zip.

```
151 {
152     return p->internal ? NULL : p->zip;
153 }
```

12.72.4.11 int QuaZipFile::getZipError ( ) const

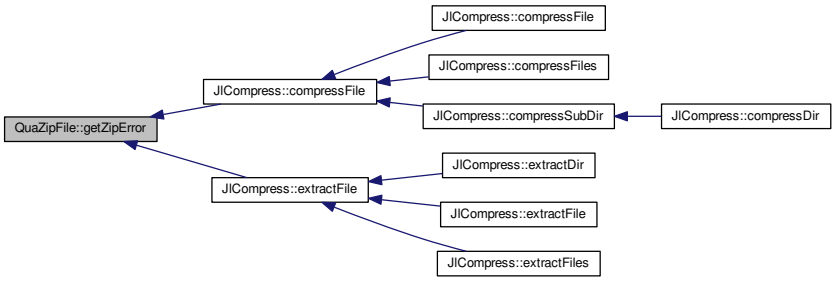
Definition at line 494 of file quazipfile.cpp.

References p, and QuaZipFilePrivate::zipError.

Referenced by JlCompress::compressFile(), and JlCompress::extractFile().

```
495 {
496   return p->zipError;
497 }
```

Here is the caller graph for this function:



12.72.4.12 QString QuaZipFile::getZipName ( ) const

If this object was created by passing QuaZip pointer to the constructor, this function will return that QuaZip's file name (or null string if that object does not have file name yet).

Otherwise, returns associated ZIP archive file name or null string if there are no name set yet.

See Also

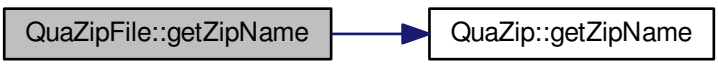
setZipName() getFileName()

Definition at line 145 of file quazipfile.cpp.

References QuaZip::getZipName(), p, and QuaZipFilePrivate::zip.

```
146 {
147   return p->zip==NULL ? QString() : p->zip->getZipName();
148 }
```

Here is the call graph for this function:



12.72.4.13 bool QuaZipFile::isRaw ( ) const

If the file is not open, the returned value is undefined.

See Also

open(OpenMode,int*,int*,bool,const char*)

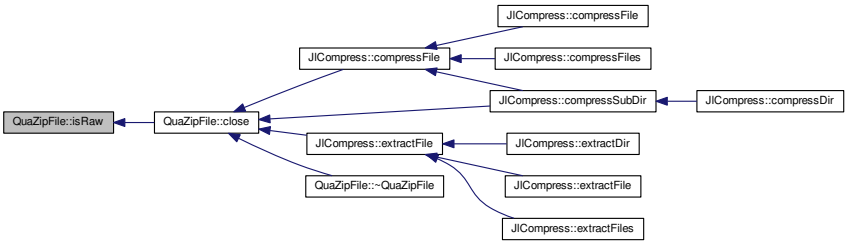
Definition at line 489 of file quazipfile.cpp.

References p, and QuaZipFilePrivate::raw.

Referenced by close().

```
490 {
491   return p->raw;
492 }
```

Here is the caller graph for this function:



12.72.4.14 bool QuaZipFile::isSequential ( ) const [virtual]

Definition at line 337 of file quazipfile.cpp.

```
338 {
339   return true;
340 }
```

12.72.4.15 bool QuaZipFile::open (OpenMode *mode*) [virtual]

Returns `true` on success, `false` otherwise. Call getZipError() to get error code.

Note

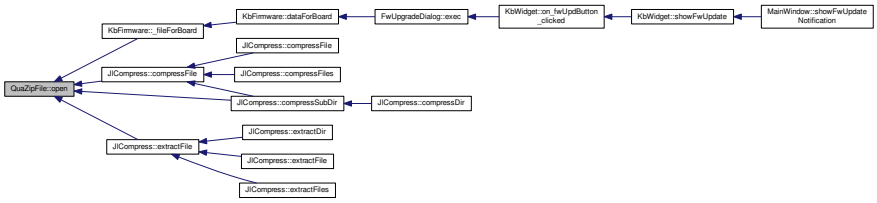
Since ZIP/UNZIP API provides buffered reading only, QuaZipFile does not support unbuffered reading. So do not pass QIODevice::Unbuffered flag in *mode*, or open will fail.

Definition at line 221 of file quazipfile.cpp.

Referenced by KbFirmware::_fileForBoard(), JICompress::compressFile(), JICompress::compressSubDir(), and JICompress::extractFile().

```
222 {
223   return open(mode, NULL);
224 }
```

Here is the caller graph for this function:



12.72.4.16 bool QuaZipFile::open (OpenMode *mode*, const char * *password*) [inline]

This is an overloaded member function, provided for convenience. It differs from the above function only in what argument(s) it accepts. Argument *password* specifies a password to decrypt the file. If it is NULL then this function behaves just like open(OpenMode).

Definition at line 298 of file quazipfile.h.

```
299     {return open(mode, NULL, NULL, false, password);}
```

12.72.4.17 bool QuaZipFile::open (OpenMode *mode*, int * *method*, int * *level*, bool *raw*, const char * *password* = NULL)

This is an overloaded member function, provided for convenience. It differs from the above function only in what argument(s) it accepts. Argument *password* specifies a password to decrypt the file.

An integers pointed by *method* and *level* will receive codes of the compression method and level used. See unzip.h.

If raw is `true` then no decompression is performed.

method should not be `NULL`. *level* can be `NULL` if you don't want to know the compression level.

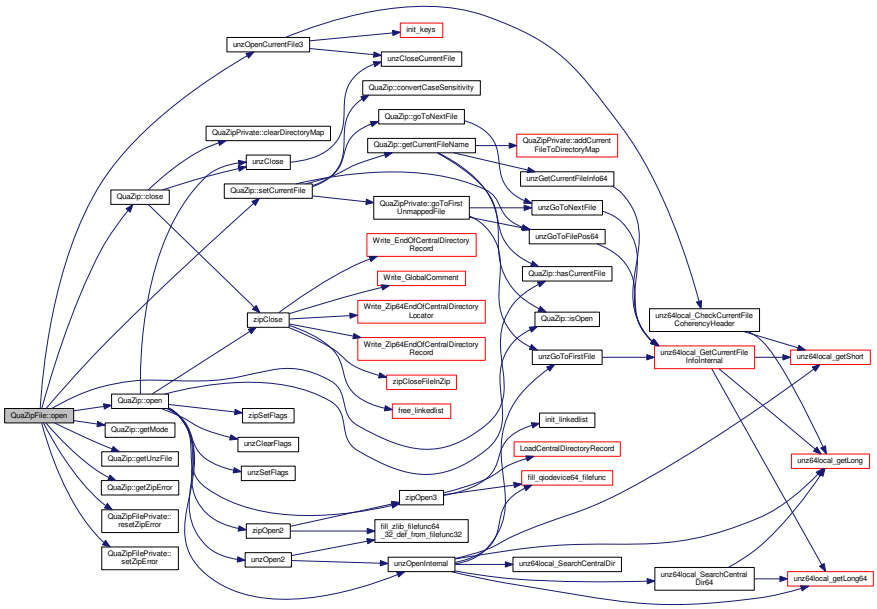
Definition at line 226 of file quazipfile.cpp.

References QuaZipFilePrivate::caseSensitivity, QuaZip::close(), QuaZipFilePrivate::fileName, QuaZip::getMode(), QuaZip::getUnzFile(), QuaZip::getZipError(), QuaZip::hasCurrentFile(), QuaZipFilePrivate::internal, QuaZip::mdUnzip, QuaZip::open(), p, QuaZipFilePrivate::raw, QuaZipFilePrivate::resetZipError(), QuaZip::setCurrentFile(), QuaZipFilePrivate::setZipError(), UNZ_OK, unzOpenCurrentFile3(), QuaZipFilePrivate::zip, and QuaZipFilePrivate::zipError.

```
227 {
228   p->resetZipError();
229   if(isOpen()) {
230     qWarning("QuaZipFile::open(): already opened");
231     return false;
232   }
233   if(mode&Unbuffered) {
234     qWarning("QuaZipFile::open(): Unbuffered mode is not supported");
235     return false;
236   }
237   if((mode&ReadOnly)&&!(mode&WriteOnly)) {
238     if(p->internal) {
239       if(!p->zip->open(QuaZip::mdUnzip)) {
240         p->setZipError(p->zip->getZipError());
241         return false;
242       }
243       if(!p->zip->setCurrentFile(p->fileName, p->
      caseSensitivity)) {
244         p->setZipError(p->zip->getZipError());
245         p->zip->close();
246         return false;
247       }
248     } else {
249       if(p->zip==NULL) {
250         qWarning("QuaZipFile::open(): zip is NULL");
251         return false;
```

```
252       }
253       if(p->zip->getMode()!=QuaZip::mdUnzip) {
254         qWarning("QuaZipFile::open(): file open mode %d incompatible with ZIP open mode %d",
255             (int)mode, (int)p->zip->getMode());
256         return false;
257       }
258       if(!p->zip->hasCurrentFile()) {
259         qWarning("QuaZipFile::open(): zip does not have current file");
260         return false;
261       }
262     }
263     p->setZipError(unzOpenCurrentFile3(p->zip->
    getUnzFile(), method, level, (int)raw, password));
264     if(p->zipError==UNZ_OK) {
265       setOpenMode(mode);
266       p->raw=raw;
267       return true;
268     } else
269       return false;
270   }
271   qWarning("QuaZipFile::open(): open mode %d not supported by this function", (int)mode);
272   return false;
273 }
```

Here is the call graph for this function:



12.72.4.18 bool QuaZipFile::open (OpenMode *mode*, const QuaZipNewInfo & *info*, const char * *password* = NULL, quint32 *crc* = 0, int *method* = Z_DEFLATED, int *level* = Z_DEFAULT_COMPRESSION, bool *raw* = false, int *windowBits* = -MAX_WBITS, int *memLevel* = MAX_MEM_LEVEL, int *strategy* = Z_DEFAULT_STRATEGY)

info argument specifies information about file. It should at least specify a correct file name. Also, it is a good idea to specify correct timestamp (by default, current time will be used). See QuaZipNewInfo.

The *password* argument specifies the password for crypting. Pass NULL if you don't need any crypting. The *crc* argument was supposed to be used for crypting too, but then it turned out that it's false information, so you need to set it to 0 unless you want to use the raw mode (see below).

Arguments *method* and *level* specify compression method and level. The only method supported is Z_DEFLATED, but you may also specify 0 for no compression. If all of the files in the archive use both method 0 and either level 0 is explicitly specified or data descriptor writing is disabled with QuaZip::setDataDescriptorWritingEnabled(), then the resulting archive is supposed to be compatible with the 1.0 ZIP format version, should you need that. Except for this, *level* has no other effects with method 0.

If *raw* is `true`, no compression is performed. In this case, *crc* and uncompressedSize field of the *info* are required.

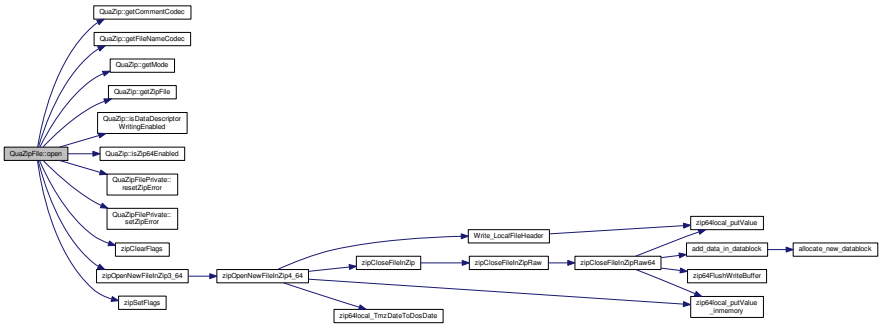
Arguments *windowBits*, *memLevel*, *strategy* provide zlib algorithms tuning. See deflateInit2() in zlib.

Definition at line 275 of file quazipfile.cpp.

References QuaZipNewInfo::comment, QuaZipFilePrivate::crc, QuaZipNewInfo::dateTime, zip_fileinfo::dosDate, zip_fileinfo::external_fa, QuaZipNewInfo::externalAttr, QuaZipNewInfo::extraGlobal, QuaZipNewInfo::extraLocal, QuaZip::getCommentCodec(), QuaZip::getFileNameCodec(), QuaZip::getMode(), QuaZip::getZipFile(), QuaZipFilePrivate::internal, zip_fileinfo::internal_fa, QuaZipNewInfo::internalAttr, QuaZip::isDataDescriptorWritingEnabled(), QuaZip::isZip64Enabled(), QuaZip::mdAdd, QuaZip::mdAppend, QuaZip::mdCreate, QuaZipNewInfo::name, p, QuaZipFilePrivate::raw, QuaZipFilePrivate::resetZipError(), QuaZipFilePrivate::setZipError(), tm_zip_s::tm_hour, tm_zip_s::tm_mday, tm_zip_s::tm_min, tm_zip_s::tm_mon, tm_zip_s::tm_sec, tm_zip_s::tm_year, zip_fileinfo::tmz_date, QuaZipFilePrivate::uncompressedSize, QuaZipNewInfo::uncompressedSize, UNZ_OK, QuaZipFilePrivate::writePos, QuaZipFilePrivate::zip, ZIP_WRITE_DATA_DESCRIPTOR, zipClearFlags(), QuaZipFilePrivate::zipError, zipOpenNewFileInZip3_64(), and zipSetFlags().

```
279 {
280   zip_fileinfo info_z;
281   p->resetZipError();
282   if(isOpen()) {
283     qWarning("QuaZipFile::open(): already opened");
284     return false;
285   }
286   if((mode&WriteOnly)&&!(mode&ReadOnly)) {
287     if(p->internal) {
288       qWarning("QuaZipFile::open(): write mode is incompatible with internal QuaZip approach");
289       return false;
290     }
291     if(p->zip==NULL) {
292       qWarning("QuaZipFile::open(): zip is NULL");
293       return false;
294     }
295     if(p->zip->getMode()!=QuaZip::mdCreate&&p->zip->
    getMode()!=QuaZip::mdAppend&&p->zip->getMode()!=
    QuaZip::mdAdd) {
296       qWarning("QuaZipFile::open(): file open mode %d incompatible with ZIP open mode %d",
297           (int)mode, (int)p->zip->getMode());
298       return false;
299     }
300     info_z.tmz_date.tm_year=info.dateTime.date().year();
301     info_z.tmz_date.tm_mon=info.dateTime.date().month() - 1;
302     info_z.tmz_date.tm_mday=info.dateTime.date().day();
303     info_z.tmz_date.tm_hour=info.dateTime.time().hour();
304     info_z.tmz_date.tm_min=info.dateTime.time().minute();
305     info_z.tmz_date.tm_sec=info.dateTime.time().second();
306     info_z.dosDate = 0;
307     info_z.internal_fa=(uLong)info.internalAttr;
308     info_z.external_fa=(uLong)info.externalAttr;
309     if (p->zip->isDataDescriptorWritingEnabled())
310         zipSetFlags(p->zip->getZipFile(),
    ZIP_WRITE_DATA_DESCRIPTOR);
311     else
312         zipClearFlags(p->zip->getZipFile(),
    ZIP_WRITE_DATA_DESCRIPTOR);
313     p->setZipError(zipOpenNewFileInZip3_64(p->
    zip->getZipFile(),
314           p->zip->getFileNameCodec()->fromUnicode(info.name).constData(), &info_z,
315           info.extraLocal.constData(), info.extraLocal.length(),
316           info.extraGlobal.constData(), info.extraGlobal.length(),
317           p->zip->getCommentCodec()->fromUnicode(info.comment).constData(),
318           method, level, (int)raw,
319           windowBits, memLevel, strategy,
320           password, (uLong)crc, p->zip->isZip64Enabled()));
321     if(p->zipError==UNZ_OK) {
322       p->writePos=0;
323       setOpenMode(mode);
324       p->raw=raw;
325       if(raw) {
326         p->crc=crc;
327         p->uncompressedSize=info.uncompressedSize;
328       }
329       return true;
330     } else
331       return false;
332   }
333   qWarning("QuaZipFile::open(): open mode %d not supported by this function", (int)mode);
334   return false;
335 }
```

Here is the call graph for this function:



12.72.4.19 QuaZipFile& QuaZipFile::operator= (const QuaZipFile & *that*) [private]

12.72.4.20 qint64 QuaZipFile::pos ( ) const [virtual]

Implementation of the QIODevice::pos(). When reading, this function is a wrapper to the ZIP/UNZIP unztell(), therefore it is unable to keep track of the ungetChar() calls (which is non-virtual and therefore is dangerous to reimplement). So if you are using ungetChar() feature of the QIODevice, this function reports incorrect value until you get back characters which you ungot.

When writing, pos() returns number of bytes already written (uncompressed unless you use raw mode).

Note

Although QuaZipFile is a sequential device and therefore pos() should always return zero, it does not, because it would be misguiding. Keep this in mind.

This function returns -1 if the file or archive is not open.

Error code returned by getZipError() is not affected by this function call.

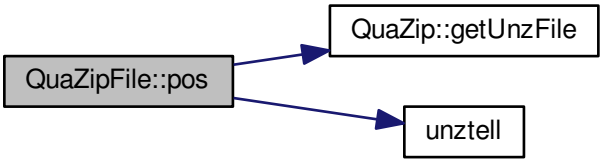
Definition at line 342 of file quazipfile.cpp.

References QuaZip::getUnzFile(), p, unztell(), QuaZipFilePrivate::writePos, and QuaZipFilePrivate::zip.

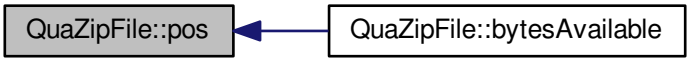
Referenced by bytesAvailable().

```
343 {
344   if(p->zip==NULL) {
345     qWarning("QuaZipFile::pos(): call setZipName() or setZip() first");
346     return -1;
347   }
348   if(!isOpen()) {
349     qWarning("QuaZipFile::pos(): file is not open");
350     return -1;
351   }
352   if(openMode()&ReadOnly)
353       // QIODevice::pos() is broken for sequential devices,
354       // but thankfully bytesAvailable() returns the number of
355       // bytes buffered, so we know how far ahead we are.
356     return unztell(p->zip->getUnzFile()) - QIODevice::bytesAvailable();
357   else
358     return p->writePos;
359 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



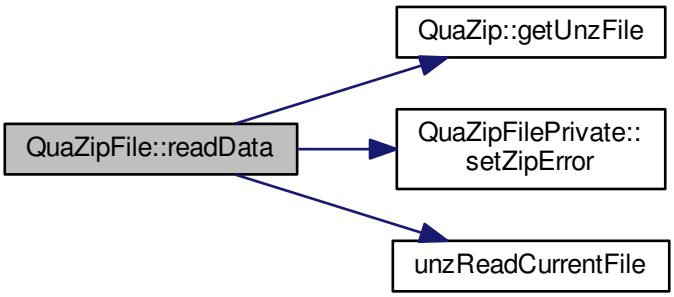
12.72.4.21 qint64 QuaZipFile::readData (char * *data*, qint64 *maxSize*) [protected]

Definition at line 457 of file quazipfile.cpp.

References QuaZip::getUnzFile(), p, QuaZipFilePrivate::setZipError(), UNZ_OK, unzReadCurrentFile(), and QuaZipFilePrivate::zip.

```
458 {
459   p->setZipError(UNZ_OK);
460   qint64 bytesRead=unzReadCurrentFile(p->zip->getUnzFile(), data, (unsigned
      )maxSize);
461   if (bytesRead < 0) {
462     p->setZipError((int) bytesRead);
463     return -1;
464   }
465   return bytesRead;
466 }
```

Here is the call graph for this function:



12.72.4.22 void QuaZipFile::setFileName (const QString & *fileName,* QuaZip::CaseSensitivity *cs* = QuaZip::csDefault)

Will do nothing if at least one of the following conditions is met:

- ZIP name has not been set yet (getZipName() returns null string).
- This QuaZipFile is associated with external QuaZip. In this case you should call that QuaZip's setCurrentFile() function instead!
- File is already open so setting the name is meaningless.

See Also

QuaZip::setCurrentFile

Definition at line 191 of file quazipfile.cpp.

References QuaZipFilePrivate::caseSensitivity, QuaZipFilePrivate::fileName, QuaZipFilePrivate::internal, p, and QuaZipFilePrivate::zip.

```
192 {
193   if(p->zip==NULL) {
194     qWarning("QuaZipFile::setFileName(): call setZipName() first");
195     return;
196   }
197   if(!p->internal) {
198     qWarning("QuaZipFile::setFileName(): should not be used when not using internal QuaZip");
199     return;
200   }
201   if(isOpen()) {
202     qWarning("QuaZipFile::setFileName(): can not set file name for already opened file");
203     return;
204   }
205   p->fileName=fileName;
206   if (p->fileName.startsWith('/'))
207       p->fileName = p->fileName.mid(1);
208   p->caseSensitivity=cs;
209 }
```

12.72.4.23 void QuaZipFile::setZip (QuaZip * *zip*)

This function destroys internal QuaZip object, if any, and makes this QuaZipFile to use current file in the *zip* object for any further operations. See QuaZipFile(QuaZip*,QObject*) for the possible pitfalls.

Will do nothing if the file is currently open. You must close() it first.

Definition at line 178 of file quazipfile.cpp.

References QuaZipFilePrivate::fileName, QuaZipFilePrivate::internal, p, and QuaZipFilePrivate::zip.

```
179 {
180   if(isOpen()) {
181     qWarning("QuaZipFile::setZip(): file is already open - can not set ZIP");
182     return;
183   }
184   if(p->zip!=NULL && p->internal)
185     delete p->zip;
186   p->zip=zip;
187   p->fileName=QString();
188   p->internal=false;
189 }
```

12.72.4.24 void QuaZipFile::setZipName (const QString & *zipName*)

Automatically creates internal QuaZip object and destroys previously created internal QuaZip object, if any.

Will do nothing if this file is already open. You must close() it first.

Definition at line 166 of file quazipfile.cpp.

References QuaZipFilePrivate::internal, p, and QuaZipFilePrivate::zip.

```
167 {
168   if(isOpen()) {
169     qWarning("QuaZipFile::setZipName(): file is already open - can not set ZIP name");
170     return;
171   }
172   if(p->zip!=NULL && p->internal)
173     delete p->zip;
174   p->zip=new QuaZip(zipName);
175   p->internal=true;
176 }
```

12.72.4.25 qint64 QuaZipFile::size ( ) const [virtual]

This function returns csize() if the file is open for reading in raw mode, usize() if it is open for reading in normal mode and pos() if it is open for writing.

Returns -1 on error, call getZipError() to get error code.

Note

> This function returns file size despite that QuaZipFile is considered to be sequential device, for which size() should return bytesAvailable() instead. But its name would be very misguiding otherwise, so just keep in mind this inconsistence.

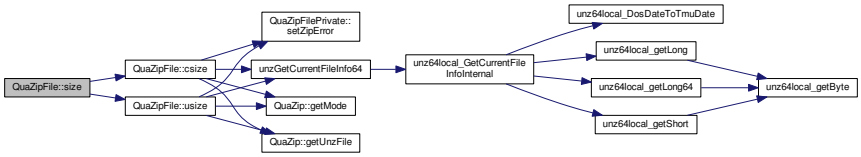
Definition at line 379 of file quazipfile.cpp.

References csize(), p, QuaZipFilePrivate::raw, usize(), and QuaZipFilePrivate::writePos.

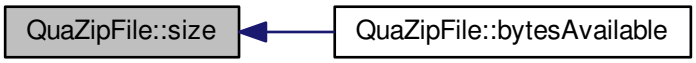
Referenced by bytesAvailable().

```
380 {
381   if(!isOpen()) {
382     qWarning("QuaZipFile::atEnd(): file is not open");
383     return -1;
384   }
385   if(openMode()&ReadOnly)
386     return p->raw?csize():usize();
387   else
388     return p->writePos;
389 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.72.4.26 qint64 QuaZipFile::usize ( ) const

Equivalent to calling getFileInfo() and then getting uncompressedSize field, but more convenient and faster. See getFileInfo() for a warning.

File must be open for reading before calling this function.

Returns -1 on error, call getZipError() to get error code.

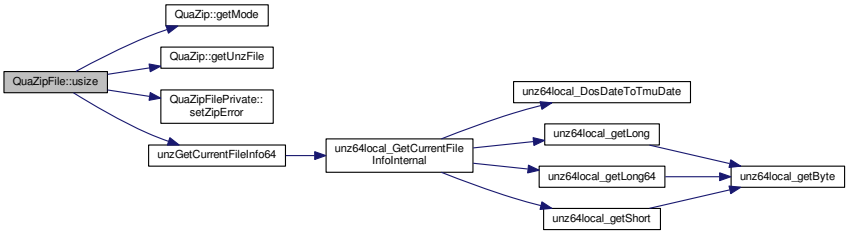
Definition at line 402 of file quazipfile.cpp.

References QuaZip::getMode(), QuaZip::getUnzFile(), QuaZip::mdUnzip, p, QuaZipFilePrivate::setZipError(), unz_file_info64_s::uncompressed_size, UNZ_OK, unzGetCurrentFileInfo64(), QuaZipFilePrivate::zip, and QuaZipFilePrivate::zipError.

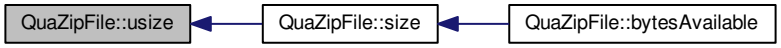
Referenced by size().

```
403 {
404   unz_file_info64 info_z;
405   p->setZipError(UNZ_OK);
406   if(p->zip==NULL||p->zip->getMode()!=QuaZip::mdUnzip) return -1;
407   p->setZipError(unzGetCurrentFileInfo64(p->
      zip->getUnzFile(), &info_z, NULL, 0, NULL, 0, NULL, 0));
408   if(p->zipError!=UNZ_OK)
409     return -1;
410   return info_z.uncompressed_size;
411 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



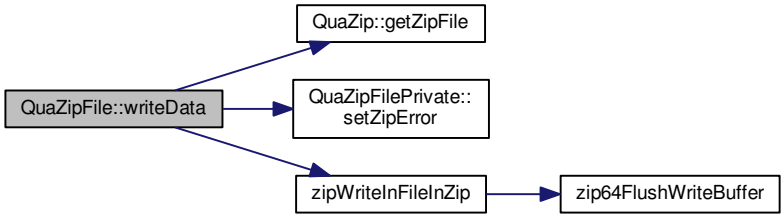
12.72.4.27 qint64 QuaZipFile::writeData (const char * *data*, qint64 *maxSize*) [protected]

Definition at line 468 of file quazipfile.cpp.

References QuaZip::getZipFile(), p, QuaZipFilePrivate::setZipError(), QuaZipFilePrivate::writePos, QuaZipFilePrivate::zip, ZIP_OK, QuaZipFilePrivate::zipError, and zipWriteInFileInZip().

```
469 {
470   p->setZipError(ZIP_OK);
471   p->setZipError(zipWriteInFileInZip(p->zip->
      getZipFile(), data, (uint)maxSize));
472   if(p->zipError!=ZIP_OK) return -1;
473   else {
474     p->writePos+=maxSize;
475     return maxSize;
476   }
477 }
```

Here is the call graph for this function:



12.72.5 Friends And Related Function Documentation

12.72.5.1 friend class QuaZipFilePrivate [friend]

Definition at line 75 of file quazipfile.h.

12.72.6 Field Documentation

12.72.6.1 QuaZipFilePrivate* QuaZipFile::p [private]

Definition at line 78 of file quazipfile.h.

Referenced by atEnd(), close(), csize(), getActualFileName(), getCaseSensitivity(), getFileInfo(), getFileName(), getZip(), getZipError(), getZipName(), isRaw(), open(), pos(), readData(), setFileName(), setZip(), setZipName(), size(), usize(), writeData(), and ~QuaZipFile().

The documentation for this class was generated from the following files:

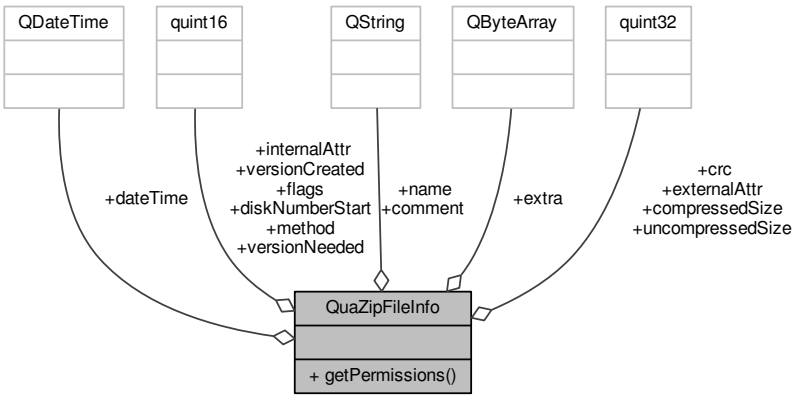
- src/ckb/quazip/quazipfile.h
- src/ckb/quazip/quazipfile.cpp

12.73 QuaZipFileInfo Struct Reference

Information about a file inside archive.

`#include <src/ckb/quazip/quazipfileinfo.h>`

Collaboration diagram for QuaZipFileInfo:



Public Member Functions

- QFile::Permissions getPermissions () const

  Get the file permissions.

Data Fields

- QString name

  File name.
- quint16 versionCreated

Version created by.

- quint16 versionNeeded

  Version needed to extract.

- quint16 flags

  General purpose flags.

- quint16 method

  Compression method.

- QDateTime dateTime

  Last modification date and time.

- quint32 crc

  CRC.

- quint32 compressedSize

  Compressed file size.

- quint32 uncompressedSize

  Uncompressed file size.

- quint16 diskNumberStart

  Disk number start.

- quint16 internalAttr

  Internal file attributes.

- quint32 externalAttr

  External file attributes.

- QString comment

  Comment.

- QByteArray extra

  Extra field.

12.73.1 Detailed Description

Deprecated Use QuaZipFileInfo64 instead. Not only it supports large files, but also more convenience methods as well.

Call QuaZip::getCurrentFileInfo() or QuaZipFile::getFileInfo() to fill this structure.

Definition at line 41 of file quazipfileinfo.h.

12.73.2 Member Function Documentation

12.73.2.1 QFile::Permissions QuaZipFileInfo::getPermissions ( ) const

Returns the high 16 bits of external attributes converted to QFile::Permissions.

Definition at line 52 of file quazipfileinfo.cpp.

References externalAttr, and permissionsFromExternalAttr().

```
53 {
54     return permissionsFromExternalAttr(externalAttr);
55 }
```

Here is the call graph for this function:

12.73.3 Field Documentation

12.73.3.1 QString QuaZipFileInfo::comment

Definition at line 67 of file quazipfileinfo.h.

Referenced by QuaZipFileInfo64::toQuaZipFileInfo().

12.73.3.2 quint32 QuaZipFileInfo::compressedSize

Definition at line 57 of file quazipfileinfo.h.

Referenced by QuaZipFileInfo64::toQuaZipFileInfo().

12.73.3.3 quint32 QuaZipFileInfo::crc

Definition at line 55 of file quazipfileinfo.h.

Referenced by QuaZipFileInfo64::toQuaZipFileInfo().

12.73.3.4 QDateTime QuaZipFileInfo::dateTime

Definition at line 53 of file quazipfileinfo.h.

Referenced by QuaZipFileInfo64::toQuaZipFileInfo().

12.73.3.5 quint16 QuaZipFileInfo::diskNumberStart

Definition at line 61 of file quazipfileinfo.h.

Referenced by QuaZipFileInfo64::toQuaZipFileInfo().

12.73.3.6 quint32 QuaZipFileInfo::externalAttr

Definition at line 65 of file quazipfileinfo.h.

Referenced by getPermissions(), and QuaZipFileInfo64::toQuaZipFileInfo().

12.73.3.7 QByteArray QuaZipFileInfo::extra

Definition at line 69 of file quazipfileinfo.h.

Referenced by QuaZipFileInfo64::toQuaZipFileInfo().

12.73.3.8 quint16 QuaZipFileInfo::flags

Definition at line 49 of file quazipfileinfo.h.

Referenced by QuaZipFileInfo64::toQuaZipFileInfo().

12.73.3.9 quint16 QuaZipFileInfo::internalAttr

Definition at line 63 of file quazipfileinfo.h.

Referenced by QuaZipFileInfo64::toQuaZipFileInfo().

12.73.3.10 quint16 QuaZipFileInfo::method

Definition at line 51 of file quazipfileinfo.h.

Referenced by QuaZipFileInfo64::toQuaZipFileInfo().

12.73.3.11 QString QuaZipFileInfo::name

Definition at line 43 of file quazipfileinfo.h.

Referenced by QuaZipFileInfo64::toQuaZipFileInfo().

12.73.3.12 quint32 QuaZipFileInfo::uncompressedSize

Definition at line 59 of file quazipfileinfo.h.

Referenced by QuaZipFileInfo64::toQuaZipFileInfo().

12.73.3.13 quint16 QuaZipFileInfo::versionCreated

Definition at line 45 of file quazipfileinfo.h.

Referenced by QuaZipFileInfo64::toQuaZipFileInfo().

12.73.3.14 quint16 QuaZipFileInfo::versionNeeded

Definition at line 47 of file quazipfileinfo.h.

Referenced by QuaZipFileInfo64::toQuaZipFileInfo().

The documentation for this struct was generated from the following files:

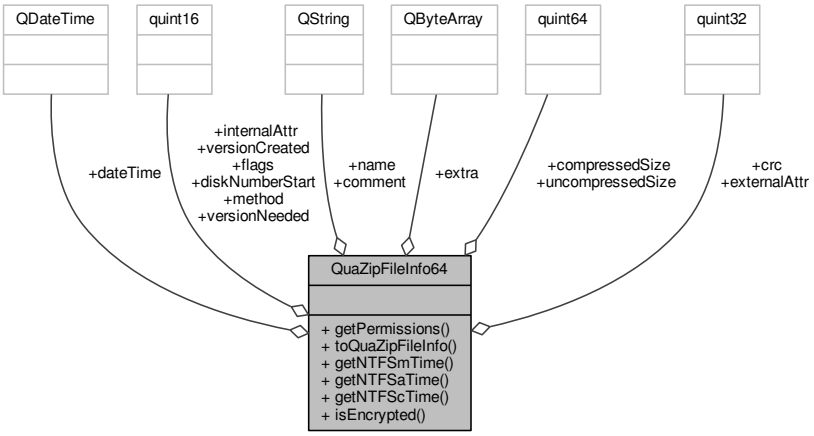
- src/ckb/quazip/quazipfileinfo.h
- src/ckb/quazip/quazipfileinfo.cpp

12.74 QuaZipFileInfo64 Struct Reference

Information about a file inside archive (with zip64 support).

`#include <src/ckb/quazip/quazipfileinfo.h>`

Collaboration diagram for QuaZipFileInfo64:



Public Member Functions

- QFile::Permissions getPermissions () const

  Get the file permissions.
- bool toQuaZipFileInfo (QuaZipFileInfo &info) const

  Converts to QuaZipFileInfo.
- QDateTime getNTFSmTime (int *fineTicks=NULL) const

  Returns the NTFS modification time.
- QDateTime getNTFSaTime (int *fineTicks=NULL) const

  Returns the NTFS access time.
- QDateTime getNTFScTime (int *fineTicks=NULL) const

  Returns the NTFS creation time.
- bool isEncrypted () const

  Checks whether the file is encrypted.

Data Fields

- QString name

  File name.
- quint16 versionCreated

  Version created by.
- quint16 versionNeeded

  Version needed to extract.
- quint16 flags

  General purpose flags.
- quint16 method

  Compression method.
- QDateTime dateTime

  Last modification date and time.
- quint32 crc

  CRC.

- quint64 compressedSize

  Compressed file size.
- quint64 uncompressedSize

  Uncompressed file size.
- quint16 diskNumberStart

  Disk number start.
- quint16 internalAttr

  Internal file attributes.
- quint32 externalAttr

  External file attributes.
- QString comment

  Comment.
- QByteArray extra

  Extra field.

12.74.1 Detailed Description

Call QuaZip::getCurrentFileInfo() or QuaZipFile::getFileInfo() to fill this structure.

Definition at line 81 of file quazipfileinfo.h.

12.74.2 Member Function Documentation

12.74.2.1 QDateTime QuaZipFileInfo64::getNTFSaTime (int * *fineTicks* = NULL) const

The getNTFS*Time() functions only work if there is an NTFS extra field present. Otherwise, they all return invalid null timestamps.

Parameters

fineTicks	If not NULL, the fractional part of milliseconds returned there, measured in 100-nanosecond ticks. Will be set to zero if there is no NTFS extra field.

See Also

dateTime
getNTFSmTime()
getNTFScTime()

Returns

The NTFS access time, UTC

Definition at line 168 of file quazipfileinfo.cpp.

References extra, and getNTFSTime().

```
169 {
170     return getNTFSTime(extra, 8, fineTicks);
171 }
```

Here is the call graph for this function:

12.74.2.2 QDateTime QuaZipFileInfo64::getNTFScTime (int * *fineTicks* = NULL) const

The getNTFS*Time() functions only work if there is an NTFS extra field present. Otherwise, they all return invalid null timestamps.

Parameters

fineTicks	If not NULL, the fractional part of milliseconds returned there, measured in 100-nanosecond ticks. Will be set to zero if there is no NTFS extra field.

See Also

dateTime
getNTFSmTime()
getNTFSaTime()

Returns

The NTFS creation time, UTC

Definition at line 173 of file quazipfileinfo.cpp.

References extra, and getNTFSTime().

```
174 {
175     return getNTFSTime(extra, 16, fineTicks);
176 }
```

Here is the call graph for this function:

12.74.2.3 QDateTime QuaZipFileInfo64::getNTFSmTime (int * *fineTicks* = NULL) const

The getNTFS*Time() functions only work if there is an NTFS extra field present. Otherwise, they all return invalid null timestamps.

Parameters

fineTicks	If not NULL, the fractional part of milliseconds returned there, measured in 100-nanosecond ticks. Will be set to zero if there is no NTFS extra field.

See Also

dateTime
getNTFSaTime()
getNTFScTime()

Returns

The NTFS modification time, UTC

Definition at line 163 of file quazipfileinfo.cpp.

References extra, and getNTFSTime().

```
164 {
165     return getNTFSTime(extra, 0, fineTicks);
166 }
```

Here is the call graph for this function:

12.74.2.4 QFile::Permissions QuaZipFileInfo64::getPermissions ( ) const

Returns the high 16 bits of external attributes converted to QFile::Permissions.

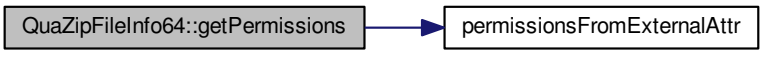
Definition at line 57 of file quazipfileinfo.cpp.

References externalAttr, and permissionsFromExternalAttr().

Referenced by JlCompress::extractFile().

```
58 {
59     return permissionsFromExternalAttr(externalAttr);
60 }
```

Here is the call graph for this function:



Here is the caller graph for this function:

12.74.2.5 bool QuaZipFileInfo64::isEncrypted ( ) const [inline]

Definition at line 175 of file quazipfileinfo.h.

```
175 {return (flags & 1) != 0;}
```

12.74.2.6 bool QuaZipFileInfo64::toQuaZipFileInfo (QuaZipFileInfo & *info*) const

If any of the fields are greater than 0xFFFFFFFFu, they are set to 0xFFFFFFFFu exactly, not just truncated. This function should be mainly used for compatibility with the old code expecting QuaZipFileInfo, in the cases when it's impossible or otherwise unadvisable (due to ABI compatibility reasons, for example) to modify that old code to use QuaZipFileInfo64.

Returns

`true` if all fields converted correctly, `false` if an overflow occured.

Definition at line 62 of file quazipfileinfo.cpp.

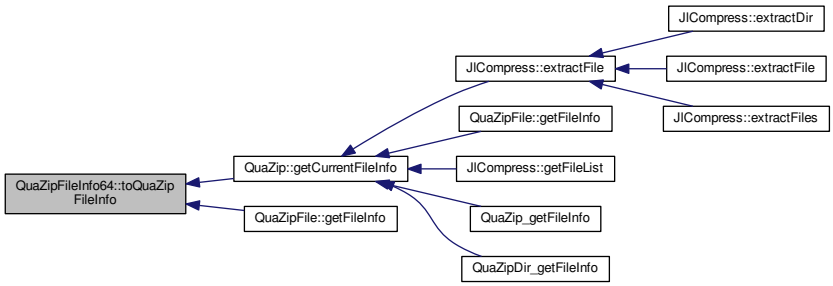
References QuaZipFileInfo::comment, comment, QuaZipFileInfo::compressedSize, compressedSize, QuaZipFileInfo::crc, crc, QuaZipFileInfo::dateTime, dateTime, QuaZipFileInfo::diskNumberStart, diskNumberStart, QuaZipFileInfo::externalAttr, externalAttr, QuaZipFileInfo::extra, extra, QuaZipFileInfo::flags, flags, QuaZipFileInfo::internalAttr, internalAttr, QuaZipFileInfo::method, method, QuaZipFileInfo::name, name, QuaZipFileInfo::uncompressedSize, uncompressedSize, QuaZipFileInfo::versionCreated, versionCreated, QuaZipFileInfo::versionNeeded, and versionNeeded.

Referenced by QuaZip::getCurrentFileInfo(), and QuaZipFile::getFileInfo().

```
63 {
64     bool noOverflow = true;
65     info.name = name;
66     info.versionCreated = versionCreated;
67     info.versionNeeded = versionNeeded;
68     info.flags = flags;
69     info.method = method;
70     info.dateTime = dateTime;
71     info.crc = crc;
72     if (compressedSize > 0xFFFFFFFFu) {
73         info.compressedSize = 0xFFFFFFFFu;
74         noOverflow = false;
75     } else {
76         info.compressedSize = compressedSize;
77     }
78     if (uncompressedSize > 0xFFFFFFFFu) {
79         info.uncompressedSize = 0xFFFFFFFFu;
80         noOverflow = false;
81     } else {
82         info.uncompressedSize = uncompressedSize;
83     }
84     info.diskNumberStart = diskNumberStart;
85     info.internalAttr = internalAttr;
```

```
86      info.externalAttr = externalAttr;
87      info.comment = comment;
88      info.extra = extra;
89      return noOverflow;
90  }
```

Here is the caller graph for this function:



12.74.3 Field Documentation

12.74.3.1 QString QuaZipFileInfo64::comment

Definition at line 114 of file quazipfileinfo.h.

Referenced by QuaZip::getCurrentFileInfo(), and toQuaZipFileInfo().

12.74.3.2 quint64 QuaZipFileInfo64::compressedSize

Definition at line 104 of file quazipfileinfo.h.

Referenced by QuaZip::getCurrentFileInfo(), QuaZipDir_getFileInfo(), and toQuaZipFileInfo().

12.74.3.3 quint32 QuaZipFileInfo64::crc

Definition at line 102 of file quazipfileinfo.h.

Referenced by QuaZip::getCurrentFileInfo(), QuaZipDir_getFileInfo(), and toQuaZipFileInfo().

12.74.3.4 QDateTime QuaZipFileInfo64::dateTime

This is the time stored in the standard ZIP header. This format only allows to store time with 2-second precision, so the seconds will always be even and the milliseconds will always be zero. If you need more precise date and time, you can try to call the getNTFSmTime() function or its siblings, provided that the archive itself contains these NTFS times.

Definition at line 100 of file quazipfileinfo.h.

Referenced by QuaZip::getCurrentFileInfo(), and toQuaZipFileInfo().

12.74.3.5 quint16 QuaZipFileInfo64::diskNumberStart

Definition at line 108 of file quazipfileinfo.h.

Referenced by QuaZip::getCurrentFileInfo(), QuaZipDir_getFileInfo(), and toQuaZipFileInfo().

12.74.3.6 quint32 QuaZipFileInfo64::externalAttr

Definition at line 112 of file quazipfileinfo.h.

Referenced by QuaZip::getCurrentFileInfo(), getPermissions(), QuaZipDir_getFileInfo(), and toQuaZipFileInfo().

12.74.3.7 QByteArray QuaZipFileInfo64::extra

Definition at line 116 of file quazipfileinfo.h.

Referenced by QuaZip::getCurrentFileInfo(), getNTFSaTime(), getNTFScTime(), getNTFSmTime(), and toQuaZipFileInfo().

12.74.3.8 quint16 QuaZipFileInfo64::flags

Definition at line 89 of file quazipfileinfo.h.

Referenced by QuaZip::getCurrentFileInfo(), QuaZipDir_getFileInfo(), and toQuaZipFileInfo().

12.74.3.9 quint16 QuaZipFileInfo64::internalAttr

Definition at line 110 of file quazipfileinfo.h.

Referenced by QuaZip::getCurrentFileInfo(), QuaZipDir_getFileInfo(), and toQuaZipFileInfo().

12.74.3.10 quint16 QuaZipFileInfo64::method

Definition at line 91 of file quazipfileinfo.h.

Referenced by QuaZip::getCurrentFileInfo(), QuaZipDir_getFileInfo(), and toQuaZipFileInfo().

12.74.3.11 QString QuaZipFileInfo64::name

Definition at line 83 of file quazipfileinfo.h.

Referenced by QuaZip::getCurrentFileInfo(), JlCompress::getFileList(), QuaZipDir_getFileInfo(), and toQuaZipFileInfo().

12.74.3.12 quint64 QuaZipFileInfo64::uncompressedSize

Definition at line 106 of file quazipfileinfo.h.

Referenced by QuaZip::getCurrentFileInfo(), QuaZipDir_getFileInfo(), and toQuaZipFileInfo().

12.74.3.13 quint16 QuaZipFileInfo64::versionCreated

Definition at line 85 of file quazipfileinfo.h.

Referenced by QuaZip::getCurrentFileInfo(), QuaZipDir_getFileInfo(), and toQuaZipFileInfo().

12.74.3.14 quint16 QuaZipFileInfo64::versionNeeded

Definition at line 87 of file quazipfileinfo.h.

Referenced by QuaZip::getCurrentFileInfo(), QuaZipDir_getFileInfo(), and toQuaZipFileInfo().

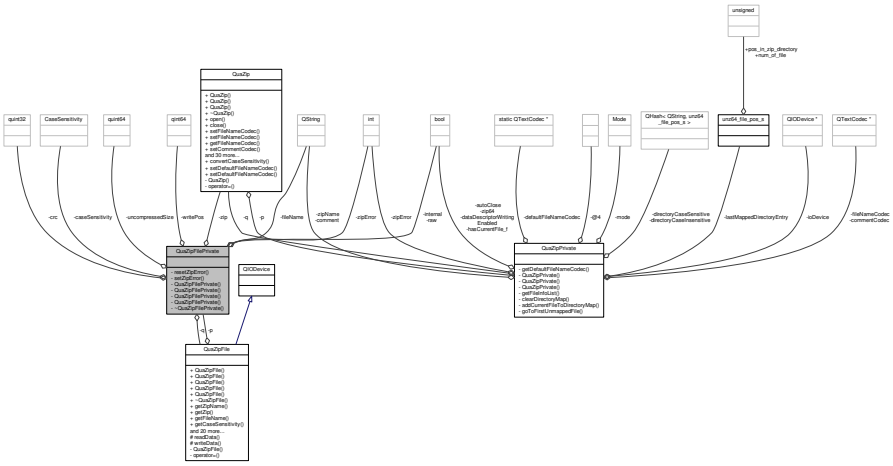
The documentation for this struct was generated from the following files:

- src/ckb/quazip/quazipfileinfo.h
- src/ckb/quazip/quazipfileinfo.cpp

12.75 QuaZipFilePrivate Class Reference

The implementation class for QuaZip.

Collaboration diagram for QuaZipFilePrivate:



Private Member Functions

- void resetZipError () const

  Resets zipError.
- void setZipError (int zipError) const

  Sets the zip error.
- QuaZipFilePrivate (QuaZipFile *q)

  The constructor for the corresponding QuaZipFile constructor.
- QuaZipFilePrivate (QuaZipFile *q, const QString &zipName)

  The constructor for the corresponding QuaZipFile constructor.
- QuaZipFilePrivate (QuaZipFile *q, const QString &zipName, const QString &fileName, QuaZip::CaseSensitivity cs)

  The constructor for the corresponding QuaZipFile constructor.
- QuaZipFilePrivate (QuaZipFile *q, QuaZip *zip)

  The constructor for the QuaZipFile constructor accepting a file name.
- ~QuaZipFilePrivate ()

  The destructor.

Private Attributes

- QuaZipFile * q

  The pointer to the associated QuaZipFile instance.
- QuaZip * zip

  The QuaZip object to work with.
- QString fileName

  The file name.

- QuaZip::CaseSensitivity caseSensitivity

  Case sensitivity mode.
- bool raw

  Whether this file is opened in the raw mode.
- qint64 writePos

  Write position to keep track of.
- quint64 uncompressedSize

  Uncompressed size to write along with a raw file.
- quint32 crc

  CRC to write along with a raw file.
- bool internal

  Whether zip points to an internal QuaZip instance.
- int zipError

  The last error.

Friends

- class QuaZipFile

12.75.1 Detailed Description

This class contains all the private stuff for the QuaZipFile class, thus allowing to preserve binary compatibility between releases, the technique known as the Pimpl (private implementation) idiom.

Definition at line 37 of file quazipfile.cpp.

12.75.2 Constructor & Destructor Documentation

12.75.2.1 QuaZipFilePrivate::QuaZipFilePrivate (QuaZipFile * *q*) [inline], [private]

Definition at line 78 of file quazipfile.cpp.

```
78                                          :
79       q(q), zip(NULL), internal(true), zipError(UNZ_OK) {}
```

12.75.2.2 QuaZipFilePrivate::QuaZipFilePrivate (QuaZipFile * *q*, const QString & *zipName*) [inline], [private]

Definition at line 81 of file quazipfile.cpp.

```
81                                                                  :
82       q(q), internal(true), zipError(UNZ_OK)
83       {
84         zip=new QuaZip(zipName);
85       }
```

12.75.2.3 QuaZipFilePrivate::QuaZipFilePrivate (QuaZipFile * *q*, const QString & *zipName*, const QString & *fileName*, QuaZip::CaseSensitivity *cs*) [inline], [private]

Definition at line 87 of file quazipfile.cpp.

```
88                                  :
89       q(q), internal(true), zipError(UNZ_OK)
90       {
91         zip=new QuaZip(zipName);
92         this->fileName=fileName;
93         if (this->fileName.startsWith('/'))
94             this->fileName = this->fileName.mid(1);
95         this->caseSensitivity=cs;
96       }
```

12.75.2.4 QuaZipFilePrivate::QuaZipFilePrivate (QuaZipFile * *q*, QuaZip * *zip*) [inline], [private]

Definition at line 98 of file quazipfile.cpp.

```
98                                                          :
99       q(q), zip(zip), internal(false), zipError(UNZ_OK) {}
```

12.75.2.5 QuaZipFilePrivate::~QuaZipFilePrivate ( ) [inline], [private]

Definition at line 101 of file quazipfile.cpp.

```
102     {
103       if (internal)
104         delete zip;
105     }
```

12.75.3 Member Function Documentation

12.75.3.1 void QuaZipFilePrivate::resetZipError ( ) const [inline], [private]

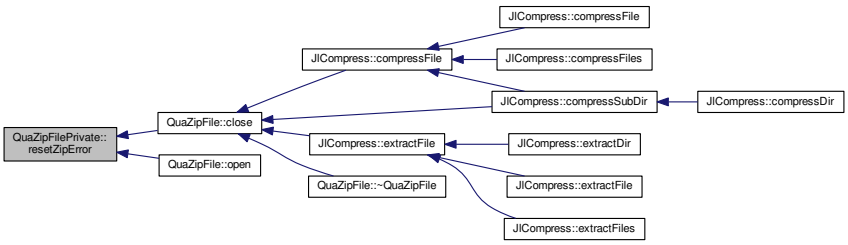
Definition at line 69 of file quazipfile.cpp.

References UNZ_OK.

Referenced by QuaZipFile::close(), and QuaZipFile::open().

```
69 {setZipError(UNZ_OK);}
```

Here is the caller graph for this function:



12.75.3.2 void QuaZipFilePrivate::setZipError (int *zipError*) const [private]

This function is marked as const although it changes one field. This allows to call it from const functions that don't change anything by themselves.

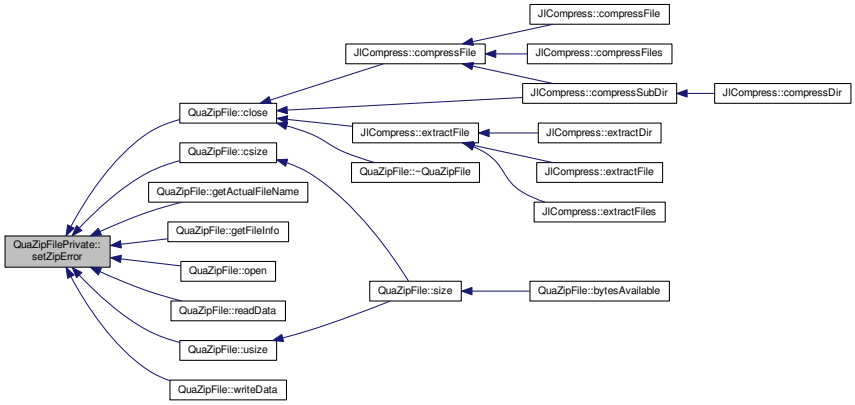
Definition at line 211 of file quazipfile.cpp.

References q, UNZ_OK, and zipError.

Referenced by QuaZipFile::close(), QuaZipFile::csize(), QuaZipFile::getActualFileName(), QuaZipFile::getFileInfo(), QuaZipFile::open(), QuaZipFile::readData(), QuaZipFile::usize(), and QuaZipFile::writeData().

```
212 {
213   QuaZipFilePrivate *fakeThis = const_cast<QuaZipFilePrivate*>(this); //
       non-const
214   fakeThis->zipError=zipError;
215   if(zipError==UNZ_OK)
216     q->setErrorString(QString());
217   else
218     q->setErrorString(QuaZipFile::tr("ZIP/UNZIP API error %1").arg(zipError));
219 }
```

Here is the caller graph for this function:



12.75.4 Friends And Related Function Documentation

12.75.4.1 friend class QuaZipFile [friend]

Definition at line 38 of file quazipfile.cpp.

12.75.5 Field Documentation

12.75.5.1 QuaZip::CaseSensitivity QuaZipFilePrivate::caseSensitivity [private]

Definition at line 47 of file quazipfile.cpp.

Referenced by QuaZipFile::getCaseSensitivity(), QuaZipFile::open(), and QuaZipFile::setFileName().

12.75.5.2 quint32 QuaZipFilePrivate::crc [private]

Definition at line 59 of file quazipfile.cpp.

Referenced by QuaZipFile::close(), and QuaZipFile::open().

12.75.5.3 QString QuaZipFilePrivate::fileName [private]

Definition at line 45 of file quazipfile.cpp.

Referenced by QuaZipFile::getFileName(), QuaZipFile::open(), QuaZipFile::setFileName(), and QuaZipFile::setZip().

12.75.5.4 bool QuaZipFilePrivate::internal [private]

This is true if the archive was opened by name, rather than by supplying an existing QuaZip instance.

Definition at line 65 of file quazipfile.cpp.

Referenced by QuaZipFile::close(), QuaZipFile::getZip(), QuaZipFile::open(), QuaZipFile::setFileName(), QuaZipFile::setZip(), and QuaZipFile::setZipName().

12.75.5.5 QuaZipFile* QuaZipFilePrivate::q [private]

Definition at line 41 of file quazipfile.cpp.

Referenced by setZipError().

12.75.5.6 bool QuaZipFilePrivate::raw [private]

Definition at line 49 of file quazipfile.cpp.

Referenced by QuaZipFile::isRaw(), QuaZipFile::open(), and QuaZipFile::size().

12.75.5.7 quint64 QuaZipFilePrivate::uncompressedSize [private]

Definition at line 57 of file quazipfile.cpp.

Referenced by QuaZipFile::close(), and QuaZipFile::open().

12.75.5.8 qint64 QuaZipFilePrivate::writePos [private]

QIODevice::pos() is broken for non-seekable devices, so we need our own position.

Definition at line 55 of file quazipfile.cpp.

Referenced by QuaZipFile::open(), QuaZipFile::pos(), QuaZipFile::size(), and QuaZipFile::writeData().

12.75.5.9 QuaZip* QuaZipFilePrivate::zip [private]

Definition at line 43 of file quazipfile.cpp.

Referenced by QuaZipFile::atEnd(), QuaZipFile::close(), QuaZipFile::csize(), QuaZipFile::getActualFileName(), QuaZipFile::getFileInfo(), QuaZipFile::getZip(), QuaZipFile::getZipName(), QuaZipFile::open(), QuaZipFile::pos(), QuaZipFile::readData(), QuaZipFile::setFileName(), QuaZipFile::setZip(), QuaZipFile::setZipName(), QuaZipFile::usize(), and QuaZipFile::writeData().

12.75.5.10 int QuaZipFilePrivate::zipError [private]

Definition at line 67 of file quazipfile.cpp.

Referenced by QuaZipFile::close(), QuaZipFile::csize(), QuaZipFile::getFileInfo(), QuaZipFile::getZipError(), QuaZipFile::open(), setZipError(), QuaZipFile::usize(), and QuaZipFile::writeData().

The documentation for this class was generated from the following file:

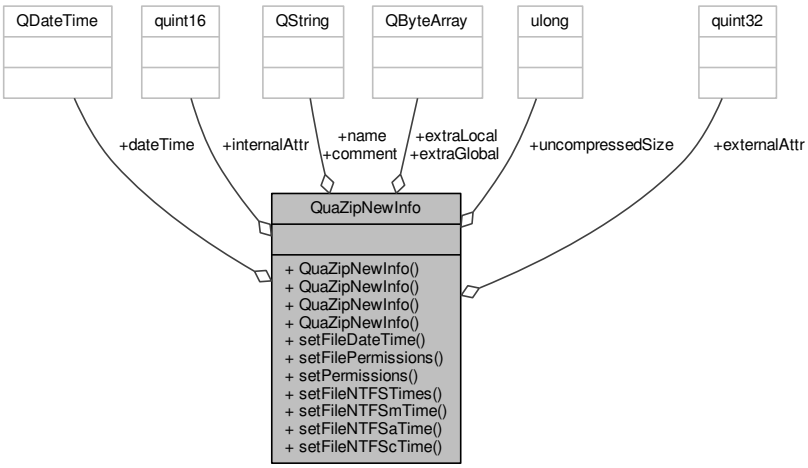
- src/ckb/quazip/quazipfile.cpp

12.76 QuaZipNewInfo Struct Reference

Information about a file to be created.

`#include <src/ckb/quazip/quazipnewinfo.h>`

Collaboration diagram for QuaZipNewInfo:



Public Member Functions

- QuaZipNewInfo (const QString &name)

  Constructs QuaZipNewInfo instance.
- QuaZipNewInfo (const QString &name, const QString &file)

  Constructs QuaZipNewInfo instance.
- QuaZipNewInfo (const QuaZipFileInfo &existing)

  Initializes the new instance from existing file info.
- QuaZipNewInfo (const QuaZipFileInfo64 &existing)

  Initializes the new instance from existing file info.
- void setFileDateTime (const QString &file)

  Sets the file timestamp from the existing file.
- void setFilePermissions (const QString &file)

  Sets the file permissions from the existing file.
- void setPermissions (QFile::Permissions permissions)

  Sets the file permissions.
- void setFileNTFSTimes (const QString &fileName)

  Sets the NTFS times from an existing file.
- void setFileNTFSmTime (const QDateTime &mTime, int fineTicks=0)

  Sets the NTFS modification time.
- void setFileNTFSaTime (const QDateTime &aTime, int fineTicks=0)

  Sets the NTFS access time.
- void setFileNTFScTime (const QDateTime &cTime, int fineTicks=0)

  Sets the NTFS creation time.

Data Fields

- QString name

  File name.
- QDateTime dateTime

  File timestamp.
- quint16 internalAttr

  File internal attributes.
- quint32 externalAttr

  File external attributes.
- QString comment

  File comment.
- QByteArray extraLocal

  File local extra field.
- QByteArray extraGlobal

  File global extra field.
- ulong uncompressedSize

  Uncompressed file size.

12.76.1 Detailed Description

This structure holds information about a file to be created inside ZIP archive. At least name should be set to something correct before passing this structure to QuaZipFile::open(OpenMode,const QuaZipNewInfo&,int,int,bool).

Zip64 support of this structure is slightly limited: in the raw mode (when a pre-compressed file is written into a ZIP file as-is), it is necessary to specify the uncompressed file size and the appropriate field is 32 bit. Since the raw mode is used extremely rare, there is no real need to have a separate QuaZipNewInfo64 structure like QuaZipFileInfo64. It may be added in the future though, if there is a demand for the raw mode with zip64 archives.

Definition at line 50 of file quazipnewinfo.h.

12.76.2 Constructor & Destructor Documentation

12.76.2.1 QuaZipNewInfo::QuaZipNewInfo (const QString & *name*)

Initializes name with *name*, dateTime with current date and time. Attributes are initialized with zeros, comment and extra field with null values.

Definition at line 79 of file quazipnewinfo.cpp.

```
79                                                  :
80   name(name), dateTime(QDateTime::currentDateTime()),
      internalAttr(0), externalAttr(0),
81   uncompressedSize(0)
82 {
83 }
```

12.76.2.2 QuaZipNewInfo::QuaZipNewInfo (const QString & *name*, const QString & *file*)

Initializes name with *name*. Timestamp and permissions are taken from the specified file. If the *file* does not exists or its timestamp is inaccessible (e. g. you do not have read permission for the directory file in), uses current time and zero permissions. Other attributes are initialized with zeros, comment and extra field with null values.

See Also

setFileDateTime()

Definition at line 85 of file quazipnewinfo.cpp.

References dateTime, and QuaZipNewInfo_setPermissions().

```
85                                                                    :
86   name(name), internalAttr(0), externalAttr(0),
      uncompressedSize(0)
87 {
88   QFileInfo info(file);
89   QDateTime lm = info.lastModified();
90   if (!info.exists()) {
91     dateTime = QDateTime::currentDateTime();
92   } else {
93     dateTime = lm;
94     QuaZipNewInfo_setPermissions(this, info.permissions(), info.isDir());
95   }
96 }
```

Here is the call graph for this function:

12.76.2.3 QuaZipNewInfo::QuaZipNewInfo (const QuaZipFileInfo & *existing*)

Mainly used when copying files between archives.

Both extra fields are initialized to existing.extra. QuaZipNewInfo

Parameters

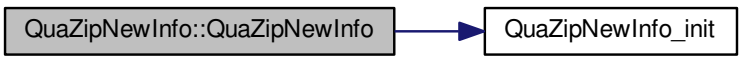
existing	

Definition at line 69 of file quazipnewinfo.cpp.

References QuaZipNewInfo_init().

```
70 {
71     QuaZipNewInfo_init(*this, existing);
72 }
```

Here is the call graph for this function:



12.76.2.4 QuaZipNewInfo::QuaZipNewInfo (const QuaZipFileInfo64 & *existing*)

Mainly used when copying files between archives.

Both extra fields are initialized to existing.extra. QuaZipNewInfo

Parameters

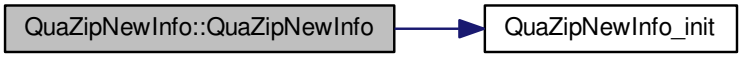
existing	

Definition at line 74 of file quazipnewinfo.cpp.

References QuaZipNewInfo_init().

```
75 {
76     QuaZipNewInfo_init(*this, existing);
77 }
```

Here is the call graph for this function:



12.76.3 Member Function Documentation

12.76.3.1 void QuaZipNewInfo::setFileDateTime (const QString & *file*)

Use this function to set the file timestamp from the existing file. Use it like this:

```
QuaZipFile zipFile(&zip);
QFile file("file-to-add");
file.open(QIODevice::ReadOnly);
QuaZipNewInfo info("file-name-in-archive");
info.setFileDateTime("file-to-add"); // take the timestamp from file
zipFile.open(QIODevice::WriteOnly, info);
```

This function does not change dateTime if some error occured (e. g. file is inaccessible).

Definition at line 98 of file quazipnewinfo.cpp.

References dateTime.

```
99 {
100   QFileInfo info(file);
101   QDateTime lm = info.lastModified();
102   if (info.exists())
103     dateTime = lm;
104 }
```

12.76.3.2 void QuaZipNewInfo::setFileNTFSaTime (const QDateTime & *aTime,* int *fineTicks* = 0)

The time is written into the NTFS record in both the local and the global extra fields, updating the existing record if there is one, or creating a new one and appending it to the end of each extra field. When updating an existing record, all other fields are left intact.

Parameters

aTime	The new access time.
fineTicks	The fractional part of milliseconds, in 100-nanosecond ticks (i. e. 9999 ticks = 999.9 microsecond). Values greater than 9999 will add milliseconds or even seconds, but this can be confusing and therefore is discouraged.

Definition at line 267 of file quazipnewinfo.cpp.

References extraGlobal, extraLocal, and setNTFSTime().

Referenced by setFileNTFSTimes().

```
268 {
269     setNTFSTime(extraLocal, aTime, 8, fineTicks);
270     setNTFSTime(extraGlobal, aTime, 8, fineTicks);
271 }
```

Here is the call graph for this function:

Here is the caller graph for this function:

12.76.3.3 void QuaZipNewInfo::setFileNTFScTime (const QDateTime & *cTime*, int *fineTicks* = 0)

The time is written into the NTFS record in both the local and the global extra fields, updating the existing record if there is one, or creating a new one and appending it to the end of each extra field. When updating an existing record, all other fields are left intact.

Parameters

cTime	The new creation time.
fineTicks	The fractional part of milliseconds, in 100-nanosecond ticks (i. e. 9999 ticks = 999.9 microsecond). Values greater than 9999 will add milliseconds or even seconds, but this can be confusing and therefore is discouraged.

Definition at line 273 of file quazipnewinfo.cpp.

References extraGlobal, extraLocal, and setNTFSTime().

Referenced by setFileNTFSTimes().

```
274 {
275     setNTFSTime(extraLocal, cTime, 16, fineTicks);
276     setNTFSTime(extraGlobal, cTime, 16, fineTicks);
277 }
```

Here is the call graph for this function:

Here is the caller graph for this function:

12.76.3.4 void QuaZipNewInfo::setFileNTFSmTime (const QDateTime & *mTime*, int *fineTicks* = 0)

The time is written into the NTFS record in both the local and the global extra fields, updating the existing record if there is one, or creating a new one and appending it to the end of each extra field. When updating an existing record, all other fields are left intact.

Parameters

mTime	The new modification time.
fineTicks	The fractional part of milliseconds, in 100-nanosecond ticks (i. e. 9999 ticks = 999.9 microsecond). Values greater than 9999 will add milliseconds or even seconds, but this can be confusing and therefore is discouraged.

Definition at line 261 of file quazipnewinfo.cpp.

References extraGlobal, extraLocal, and setNTFSTime().

Referenced by setFileNTFSTimes().

```
262 {
263     setNTFSTime(extraLocal, mTime, 0, fineTicks);
264     setNTFSTime(extraGlobal, mTime, 0, fineTicks);
265 }
```

Here is the call graph for this function:

Here is the caller graph for this function:

12.76.3.5 void QuaZipNewInfo::setFileNTFSTimes (const QString & *fileName*)

If the file doesn't exist, a warning is printed to the stderr and nothing is done. Otherwise, all three times, as reported by QFileInfo::lastModified(), QFileInfo::lastRead() and QFileInfo::created(), are written to the NTFS extra field record.

The NTFS record is written to both the local and the global extra fields, updating the existing record if there is one, or creating a new one and appending it to the end of each extra field.

The microseconds will be zero, as they aren't reported by QFileInfo.

Parameters

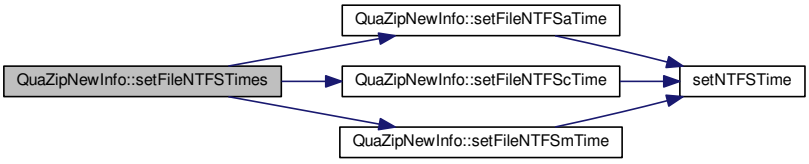
fileName	

Definition at line 118 of file quazipnewinfo.cpp.

References setFileNTFSaTime(), setFileNTFScTime(), and setFileNTFSmTime().

```
119 {
120     QFileInfo fi(fileName);
121     if (!fi.exists()) {
122         qWarning("QuaZipNewInfo::setFileNTFSTimes(): '%s' doesn't exist",
123                  fileName.toUtf8().constData());
124         return;
125     }
126     setFileNTFSmTime(fi.lastModified());
127     setFileNTFSaTime(fi.lastRead());
128     setFileNTFScTime(fi.created());
129 }
```

Here is the call graph for this function:



12.76.3.6 void QuaZipNewInfo::setFilePermissions (const QString & *file*)

Takes permissions from the file and sets the high 16 bits of external attributes. Uses QFileInfo to get permissions on all platforms.

Definition at line 106 of file quazipnewinfo.cpp.

References QuaZipNewInfo_setPermissions().

```
107 {
108     QFileInfo info = QFileInfo(file);
109     QFile::Permissions perm = info.permissions();
110     QuaZipNewInfo_setPermissions(this, perm, info.isDir());
111 }
```

Here is the call graph for this function:

12.76.3.7 void QuaZipNewInfo::setPermissions (QFile::Permissions *permissions*)

Modifies the highest 16 bits of external attributes. The type part is set to dir if the name ends with a slash, and to regular file otherwise.

Definition at line 113 of file quazipnewinfo.cpp.

References name, and QuaZipNewInfo_setPermissions().

```
114 {
115     QuaZipNewInfo_setPermissions(this, permissions,
    name.endsWith('/'));
116 }
```

Here is the call graph for this function:

12.76.4 Field Documentation

12.76.4.1 QString QuaZipNewInfo::comment

Will be encoded using QuaZip::getCommentCodec().

Definition at line 75 of file quazipnewinfo.h.

Referenced by QuaZipFile::open().

12.76.4.2 QDateTime QuaZipNewInfo::dateTime

This is the last file modification date and time. Will be stored in the archive central directory. It is a good practice to set it to the source file timestamp instead of archive creating time. Use setFileDateTime() or QuaZipNewInfo(const QString&, const QString&).

Definition at line 62 of file quazipnewinfo.h.

Referenced by QuaZipFile::open(), QuaZipNewInfo(), and setFileDateTime().

12.76.4.3 quint32 QuaZipNewInfo::externalAttr

The highest 16 bits contain Unix file permissions and type (dir or file). The constructor QuaZipNewInfo(const QString&, const QString&) takes permissions from the provided file.

Definition at line 71 of file quazipnewinfo.h.

Referenced by QuaZipFile::open(), and QuaZipNewInfo_setPermissions().

12.76.4.4 QByteArray QuaZipNewInfo::extraGlobal

Definition at line 79 of file quazipnewinfo.h.

Referenced by QuaZipFile::open(), setFileNTFSaTime(), setFileNTFScTime(), and setFileNTFSmTime().

12.76.4.5 QByteArray QuaZipNewInfo::extraLocal

Definition at line 77 of file quazipnewinfo.h.

Referenced by QuaZipFile::open(), setFileNTFSaTime(), setFileNTFScTime(), and setFileNTFSmTime().

12.76.4.6 quint16 QuaZipNewInfo::internalAttr

Definition at line 64 of file quazipnewinfo.h.

Referenced by QuaZipFile::open().

12.76.4.7 QString QuaZipNewInfo::name

This field holds file name inside archive, including path relative to archive root.

Definition at line 55 of file quazipnewinfo.h.

Referenced by QuaZipFile::open(), and setPermissions().

12.76.4.8 ulong QuaZipNewInfo::uncompressedSize

This is only needed if you are using raw file zipping mode, i. e. adding precompressed file in the zip archive.

Definition at line 84 of file quazipnewinfo.h.

Referenced by QuaZipFile::open().

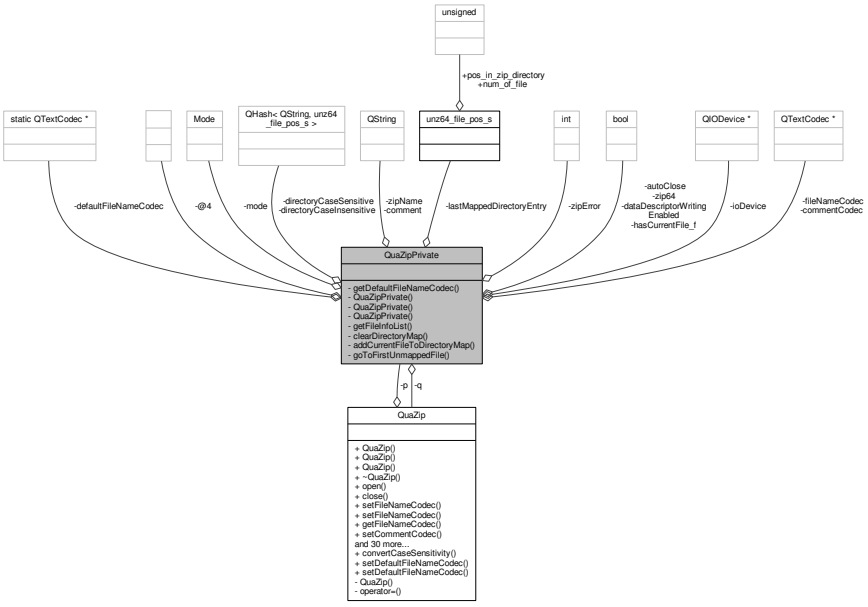
The documentation for this struct was generated from the following files:

- src/ckb/quazip/quazipnewinfo.h
- src/ckb/quazip/quazipnewinfo.cpp

12.77 QuaZipPrivate Class Reference

All the internal stuff for the QuaZip class.

Collaboration diagram for QuaZipPrivate:



Private Member Functions

- QTextCodec * getDefaultFileNameCodec ()
- QuaZipPrivate (QuaZip *q)

  The constructor for the corresponding QuaZip constructor.
- QuaZipPrivate (QuaZip *q, const QString &zipName)

  The constructor for the corresponding QuaZip constructor.
- QuaZipPrivate (QuaZip *q, QIODevice *ioDevice)

  The constructor for the corresponding QuaZip constructor.
- template<typename TFileInfo >
  bool getFileInfoList (QList< TFileInfo > *result) const

  Returns either a list of file names or a list of QuaZipFileInfo.
- void clearDirectoryMap ()

  Stores map of filenames and file locations for unzipping.
- void addCurrentFileToDirectoryMap (const QString &fileName)
- bool goToFirstUnmappedFile ()

Private Attributes

- QuaZip * q

  The pointer to the corresponding QuaZip instance.
- QTextCodec * fileNameCodec

  The codec for file names.
- QTextCodec * commentCodec

  The codec for comments.
- QString zipName

  The archive file name.
- QIODevice * ioDevice

  The device to access the archive.

- QString comment

  The global comment.
- QuaZip::Mode mode

  The open mode.
- union {
  unzFile **unzFile_f**
  The internal handle for UNZIP modes.
  zipFile **zipFile_f**
  The internal handle for ZIP modes.
  };

- bool hasCurrentFile_f

  Whether a current file is set.
- int zipError

  The last error.
- bool dataDescriptorWritingEnabled

  Whether the data descriptor writing mode is enabled.
- bool zip64

  The zip64 mode.
- bool autoClose

  The auto-close flag.
- QHash< QString, unz64_file_pos > directoryCaseSensitive
- QHash< QString, unz64_file_pos > directoryCaseInsensitive
- unz64_file_pos lastMappedDirectoryEntry

Static Private Attributes

- static QTextCodec * defaultFileNameCodec = NULL

Friends

- class QuaZip

12.77.1 Detailed Description

This class keeps all the private stuff for the QuaZip class so it can be changed without breaking binary compatibility, according to the Pimpl idiom.

Definition at line 39 of file quazip.cpp.

12.77.2 Constructor & Destructor Documentation

12.77.2.1 QuaZipPrivate::QuaZipPrivate (QuaZip * *q*) [inline], [private]

Definition at line 81 of file quazip.cpp.

References lastMappedDirectoryEntry, unz64_file_pos_s::num_of_file, and unz64_file_pos_s::pos_in_zip_directory.

```
81                                  :
82       q(q),
83       fileNameCodec(getDefaultFileNameCodec()),
84       commentCodec(QTextCodec::codecForLocale()),
85       ioDevice(NULL),
86       mode(QuaZip::mdNotOpen),
```

```
87        hasCurrentFile_f(false),
88        zipError(UNZ_OK),
89        dataDescriptorWritingEnabled(true),
90        zip64(false),
91        autoClose(true)
92    {
93        lastMappedDirectoryEntry.num_of_file = 0;
94        lastMappedDirectoryEntry.pos_in_zip_directory = 0;
95    }
```

12.77.2.2 QuaZipPrivate::QuaZipPrivate (QuaZip * *q*, const QString & *zipName*) [inline], [private]

Definition at line 97 of file quazip.cpp.

References lastMappedDirectoryEntry, unz64_file_pos_s::num_of_file, and unz64_file_pos_s::pos_in_zip_directory.

```
97                                                    :
98        q(q),
99        fileNameCodec(getDefaultFileNameCodec()),
100       commentCodec(QTextCodec::codecForLocale()),
101       zipName(zipName),
102       ioDevice(NULL),
103       mode(QuaZip::mdNotOpen),
104       hasCurrentFile_f(false),
105       zipError(UNZ_OK),
106       dataDescriptorWritingEnabled(true),
107       zip64(false),
108       autoClose(true)
109   {
110       lastMappedDirectoryEntry.num_of_file = 0;
111       lastMappedDirectoryEntry.pos_in_zip_directory = 0;
112   }
```

12.77.2.3 QuaZipPrivate::QuaZipPrivate (QuaZip * *q*, QIODevice * *ioDevice*) [inline], [private]

Definition at line 114 of file quazip.cpp.

References lastMappedDirectoryEntry, unz64_file_pos_s::num_of_file, and unz64_file_pos_s::pos_in_zip_directory.

```
114                                                    :
115       q(q),
116       fileNameCodec(getDefaultFileNameCodec()),
117       commentCodec(QTextCodec::codecForLocale()),
118       ioDevice(ioDevice),
119       mode(QuaZip::mdNotOpen),
120       hasCurrentFile_f(false),
121       zipError(UNZ_OK),
122       dataDescriptorWritingEnabled(true),
123       zip64(false),
124       autoClose(true)
125   {
126       lastMappedDirectoryEntry.num_of_file = 0;
127       lastMappedDirectoryEntry.pos_in_zip_directory = 0;
128   }
```

12.77.3 Member Function Documentation

12.77.3.1 void QuaZipPrivate::addCurrentFileToDirectoryMap (const QString & *fileName*) [inline], [private]

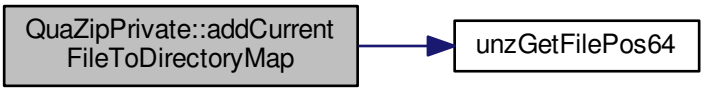
Definition at line 153 of file quazip.cpp.

References directoryCaseInsensitive, directoryCaseSensitive, hasCurrentFile_f, lastMappedDirectoryEntry, unz64_file_pos_s::pos_in_zip_directory, and unzGetFilePos64().

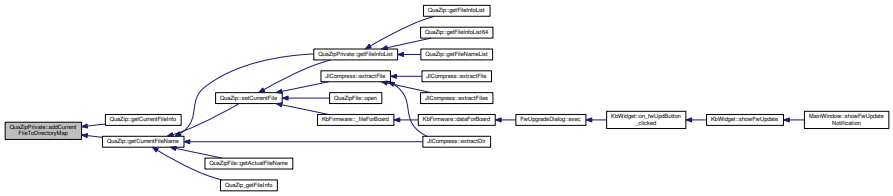
Referenced by QuaZip::getCurrentFileInfo(), and QuaZip::getCurrentFileName().

```
154 {
155     if (!hasCurrentFile_f || fileName.isEmpty()) {
156         return;
157     }
158     // Adds current file to filename map as fileName
159     unz64_file_pos fileDirectoryPos;
160     unzGetFilePos64(unzFile_f, &fileDirectoryPos);
161     directoryCaseSensitive.insert(fileName, fileDirectoryPos);
162     // Only add lowercase to directory map if not already there
163     // ensures only map the first one seen
164     QString lower = fileName.toLower();
165     if (!directoryCaseInsensitive.contains(lower))
166         directoryCaseInsensitive.insert(lower, fileDirectoryPos);
167     // Mark last one
168     if (fileDirectoryPos.pos_in_zip_directory >
      lastMappedDirectoryEntry.pos_in_zip_directory)
169         lastMappedDirectoryEntry = fileDirectoryPos;
170 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.77.3.2 void QuaZipPrivate::clearDirectoryMap ( ) [inline], [private]

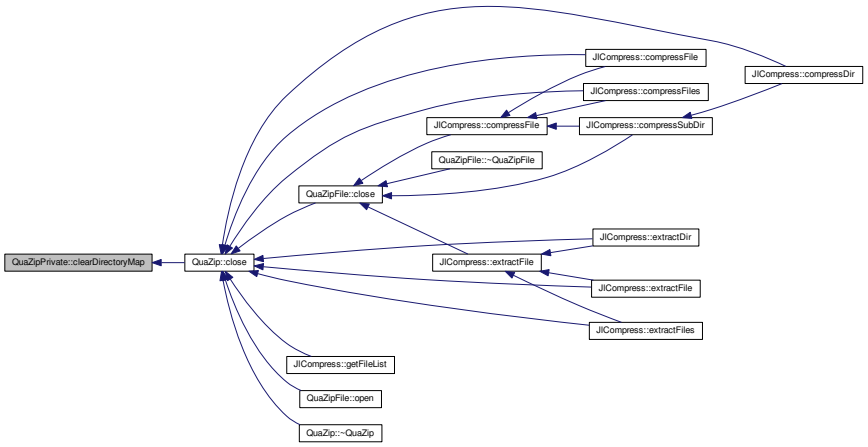
Definition at line 145 of file quazip.cpp.

References directoryCaseInsensitive, directoryCaseSensitive, lastMappedDirectoryEntry, unz64_file_pos_s::num_of_file, and unz64_file_pos_s::pos_in_zip_directory.

Referenced by QuaZip::close().

```
146 {
147     directoryCaseInsensitive.clear();
148     directoryCaseSensitive.clear();
149     lastMappedDirectoryEntry.num_of_file = 0;
150     lastMappedDirectoryEntry.pos_in_zip_directory = 0;
151 }
```

Here is the caller graph for this function:



12.77.3.3 QTextCodec* QuaZipPrivate::getDefaultFileNameCodec ( ) [inline], [private]

Definition at line 72 of file quazip.cpp.

References defaultFileNameCodec.

```
73     {
74         if (defaultFileNameCodec == NULL) {
75             return QTextCodec::codecForLocale();
76         } else {
77             return defaultFileNameCodec;
78         }
79     }
```

12.77.3.4 template<typename TFileInfo > bool QuaZipPrivate::getFileInfoList (QList< TFileInfo > * *result*) const [private]

Definition at line 687 of file quazip.cpp.

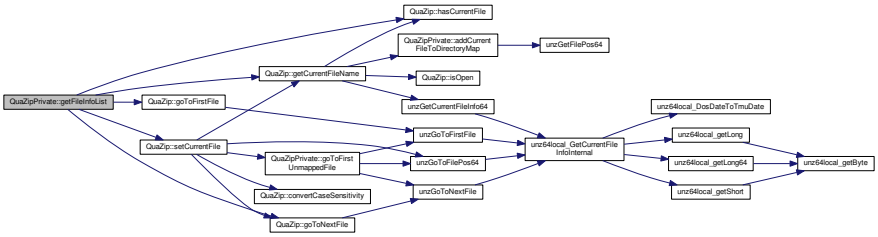
References QuaZip::getCurrentFileName(), QuaZip::goToFirstFile(), QuaZip::goToNextFile(), QuaZip::hasCurrentFile(), QuaZip::mdUnzip, mode, q, QuaZip::setCurrentFile(), UNZ_OK, and zipError.

Referenced by QuaZip::getFileInfoList(), QuaZip::getFileInfoList64(), and QuaZip::getFileNameList().

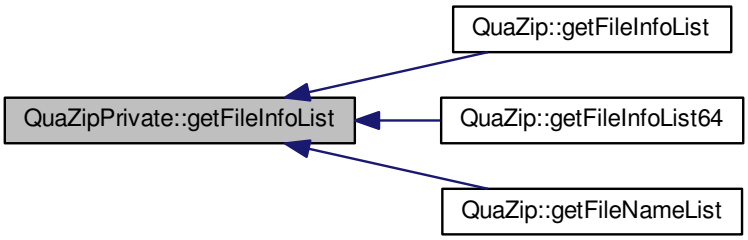
```
688 {
689   QuaZipPrivate *fakeThis=const_cast<QuaZipPrivate*>(this);
690   fakeThis->zipError=UNZ_OK;
691   if (mode!=QuaZip::mdUnzip) {
692     qWarning("QuaZip::getFileNameList/getFileInfoList(): "
693              "ZIP is not open in mdUnzip mode");
694     return false;
695   }
696   QString currentFile;
697   if (q->hasCurrentFile()) {
698       currentFile = q->getCurrentFileName();
699   }
700   if (q->goToFirstFile()) {
701       do {
702           bool ok;
703           result->append(QuaZip_getFileInfo<TFileInfo>(q, &ok));
704           if (!ok)
705               return false;
706       } while (q->goToNextFile());
707   }
708   if (zipError != UNZ_OK)
709       return false;
```

```
710   if (currentFile.isEmpty()) {
711       if (!q->goToFirstFile())
712           return false;
713   } else {
714       if (!q->setCurrentFile(currentFile))
715           return false;
716   }
717   return true;
718 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.77.3.5 bool QuaZipPrivate::goToFirstUnmappedFile ( ) [private]

Definition at line 172 of file quazip.cpp.

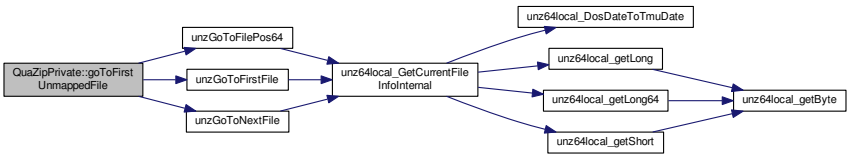
References hasCurrentFile_f, lastMappedDirectoryEntry, QuaZip::mdUnzip, mode, unz64_file_pos_s::pos_in_zip_directory, UNZ_END_OF_LIST_OF_FILE, UNZ_OK, unzGoToFilePos64(), unzGoToFirstFile(), unzGoToNextFile(), and zipError.

Referenced by QuaZip::setCurrentFile().

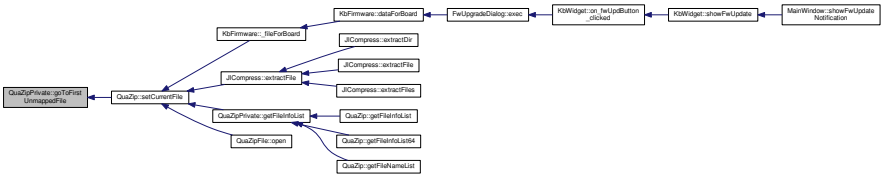
```
173 {
174     zipError = UNZ_OK;
175     if (mode != QuaZip::mdUnzip) {
176         qWarning("QuaZipPrivate::goToNextUnmappedFile(): ZIP is not open in mdUnzip mode");
177         return false;
178     }
179     // If not mapped anything, go to beginning
180     if (lastMappedDirectoryEntry.pos_in_zip_directory == 0) {
181         unzGoToFirstFile(unzFile_f);
182     } else {
183         // Goto the last one mapped, plus one
184         unzGoToFilePos64(unzFile_f, &lastMappedDirectoryEntry);
185         unzGoToNextFile(unzFile_f);
186     }
```

```
187     hasCurrentFile_f=zipError==UNZ_OK;
188     if(zipError==UNZ_END_OF_LIST_OF_FILE)
189       zipError=UNZ_OK;
190     return hasCurrentFile_f;
191 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.77.4 Friends And Related Function Documentation

12.77.4.1 friend class QuaZip [friend]

Definition at line 40 of file quazip.cpp.

12.77.5 Field Documentation

12.77.5.1 union { ... } [private]

12.77.5.2 bool QuaZipPrivate::autoClose [private]

Definition at line 71 of file quazip.cpp.

Referenced by QuaZip::isAutoClose(), QuaZip::open(), and QuaZip::setAutoClose().

12.77.5.3 QString QuaZipPrivate::comment [private]

Definition at line 53 of file quazip.cpp.

Referenced by QuaZip::close(), and QuaZip::setComment().

12.77.5.4 QTextCodec* QuaZipPrivate::commentCodec [private]

Definition at line 47 of file quazip.cpp.

Referenced by QuaZip::close(), QuaZip::getComment(), QuaZip::getCommentCodec(), QuaZip::getCurrentFileInfo(), and QuaZip::setCommentCodec().

12.77.5.5 bool QuaZipPrivate::dataDescriptorWritingEnabled [private]

Definition at line 67 of file quazip.cpp.

Referenced by QuaZip::isDataDescriptorWritingEnabled(), QuaZip::open(), and QuaZip::setDataDescriptorWritingEnabled().

12.77.5.6 QTextCodec * QuaZipPrivate::defaultFileNameCodec = NULL [static], [private]

Definition at line 140 of file quazip.cpp.

Referenced by getDefaultFileNameCodec(), and QuaZip::setDefaultFileNameCodec().

12.77.5.7 QHash<QString, unz64_file_pos> QuaZipPrivate::directoryCaseInsensitive [private]

Definition at line 138 of file quazip.cpp.

Referenced by addCurrentFileToDirectoryMap(), clearDirectoryMap(), and QuaZip::setCurrentFile().

12.77.5.8 QHash<QString, unz64_file_pos> QuaZipPrivate::directoryCaseSensitive [private]

Definition at line 137 of file quazip.cpp.

Referenced by addCurrentFileToDirectoryMap(), clearDirectoryMap(), and QuaZip::setCurrentFile().

12.77.5.9 QTextCodec* QuaZipPrivate::fileNameCodec [private]

Definition at line 45 of file quazip.cpp.

Referenced by QuaZip::getCurrentFileInfo(), QuaZip::getCurrentFileName(), QuaZip::getFileNameCodec(), and QuaZip::setFileNameCodec().

12.77.5.10 bool QuaZipPrivate::hasCurrentFile_f [private]

Definition at line 63 of file quazip.cpp.

Referenced by addCurrentFileToDirectoryMap(), QuaZip::goToFirstFile(), goToFirstUnmappedFile(), QuaZip::goToNextFile(), QuaZip::hasCurrentFile(), and QuaZip::setCurrentFile().

12.77.5.11 QIODevice* QuaZipPrivate::ioDevice [private]

Definition at line 51 of file quazip.cpp.

Referenced by QuaZip::close(), QuaZip::getIoDevice(), QuaZip::open(), QuaZip::setIoDevice(), and QuaZip::setZipName().

12.77.5.12 unz64_file_pos QuaZipPrivate::lastMappedDirectoryEntry [private]

Definition at line 139 of file quazip.cpp.

Referenced by addCurrentFileToDirectoryMap(), clearDirectoryMap(), goToFirstUnmappedFile(), and QuaZipPrivate().

12.77.5.13 QuaZip::Mode QuaZipPrivate::mode [private]

Definition at line 55 of file quazip.cpp.

Referenced by QuaZip::close(), QuaZip::getComment(), QuaZip::getCurrentFileInfo(), QuaZip::getCurrentFileName(), QuaZip::getEntriesCount(), getFileInfoList(), QuaZip::getMode(), QuaZip::goToFirstFile(), goToFirstUnmappedFile(), QuaZip::goToNextFile(), QuaZip::isOpen(), QuaZip::open(), and QuaZip::setCurrentFile().

12.77.5.14 QuaZip* QuaZipPrivate::q [private]

Definition at line 43 of file quazip.cpp.

Referenced by getFileInfoList().

12.77.5.15 bool QuaZipPrivate::zip64 [private]

Definition at line 69 of file quazip.cpp.

Referenced by QuaZip::isZip64Enabled(), and QuaZip::setZip64Enabled().

12.77.5.16 int QuaZipPrivate::zipError [private]

Definition at line 65 of file quazip.cpp.

Referenced by QuaZip::close(), QuaZip::getComment(), QuaZip::getCurrentFileInfo(), QuaZip::getCurrentFileName(), QuaZip::getEntriesCount(), getFileInfoList(), QuaZip::getZipError(), QuaZip::goToFirstFile(), goToFirstUnmappedFile(), QuaZip::goToNextFile(), QuaZip::open(), and QuaZip::setCurrentFile().

12.77.5.17 QString QuaZipPrivate::zipName [private]

Definition at line 49 of file quazip.cpp.

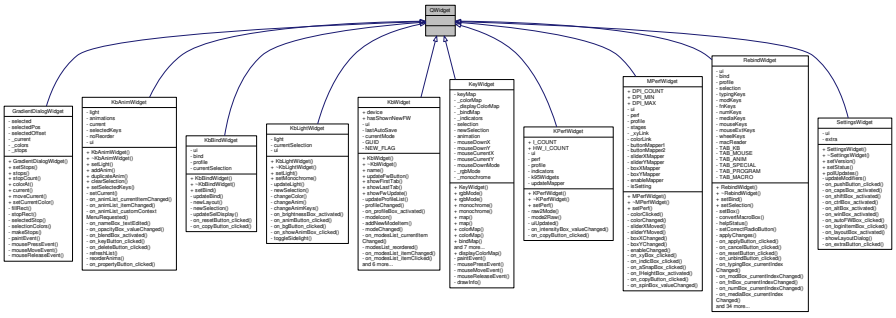
Referenced by QuaZip::close(), QuaZip::getIoDevice(), QuaZip::getZipName(), QuaZip::open(), QuaZip::setIoDevice(), and QuaZip::setZipName().

The documentation for this class was generated from the following file:

- src/ckb/quazip/quazip.cpp

12.78 QWidget Class Reference

Inheritance diagram for QWidget:



Collaboration diagram for QWidget:

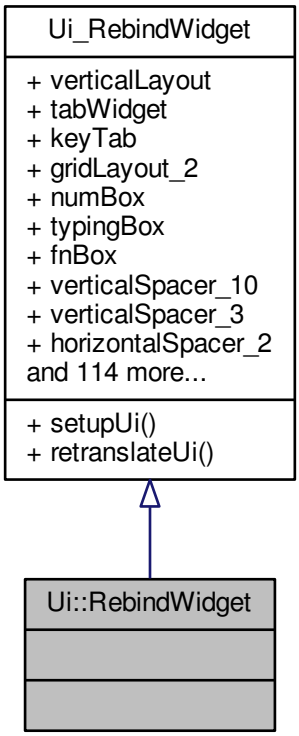
The documentation for this class was generated from the following file:

- src/ckb/keywidget.h

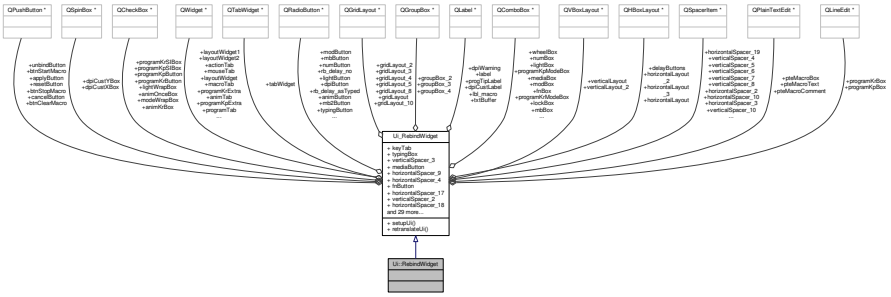
12.79 Ui::RebindWidget Class Reference

`#include <src/ckb/ui_rebindwidget.h>`

Inheritance diagram for Ui::RebindWidget:



Collaboration diagram for Ui::RebindWidget:



Public Member Functions

- void setupUi (QWidget *RebindWidget)
- void retranslateUi (QWidget *RebindWidget)

Data Fields

- QVBoxLayout * verticalLayout
- QTabWidget * tabWidget
- QWidget * keyTab
- QGridLayout * gridLayout_2
- QComboBox * numBox
- QComboBox * typingBox
- QComboBox * fnBox
- QSpacerItem * verticalSpacer_10
- QSpacerItem * verticalSpacer_3
- QSpacerItem * horizontalSpacer_2
- QRadioButton * typingButton
- QComboBox * modBox
- QSpacerItem * horizontalSpacer_3
- QRadioButton * mediaButton
- QSpacerItem * verticalSpacer_4
- QSpacerItem * horizontalSpacer_9
- QSpacerItem * horizontalSpacer_4
- QRadioButton * modButton
- QRadioButton * fnButton
- QComboBox * mediaBox
- QRadioButton * numButton
- QSpacerItem * horizontalSpacer_17
- QWidget * mouseTab
- QGridLayout * gridLayout
- QRadioButton * dpiButton
- QSpacerItem * verticalSpacer_2
- QSpacerItem * verticalSpacer_8
- QSpacerItem * horizontalSpacer_18
- QSpacerItem * horizontalSpacer_20
- QComboBox * mb2Box
- QComboBox * dpiBox
- QSpinBox * dpiCustYBox
- QComboBox * mbBox

- QComboBox * wheelBox
- QRadioButton * mbButton
- QSpinBox * dpiCustXBox
- QLabel * dpiWarning
- QRadioButton * wheelButton
- QSpacerItem * horizontalSpacer_22
- QRadioButton * mb2Button
- QLabel * dpiCustLabel
- QSpacerItem * horizontalSpacer_19
- QSpacerItem * horizontalSpacer_21
- QWidget * animTab
- QGridLayout * gridLayout_5
- QSpacerItem * verticalSpacer_11
- QComboBox * animBox
- QSpacerItem * horizontalSpacer_24
- QSpacerItem * horizontalSpacer_25
- QCheckBox * animKrBox
- QCheckBox * animOnceBox
- QSpacerItem * horizontalSpacer_26
- QRadioButton * animButton
- QSpacerItem * horizontalSpacer_23
- QWidget * actionTab
- QGridLayout * gridLayout_3
- QSpacerItem * horizontalSpacer_5
- QRadioButton * modeButton
- QCheckBox * modeWrapBox
- QCheckBox * lightWrapBox
- QSpacerItem * verticalSpacer_7
- QSpacerItem * verticalSpacer_6
- QRadioButton * lockButton
- QSpacerItem * verticalSpacer_5
- QComboBox * lightBox
- QComboBox * lockBox
- QRadioButton * lightButton
- QComboBox * modeBox
- QSpacerItem * horizontalSpacer_8
- QSpacerItem * horizontalSpacer_10
- QSpacerItem * horizontalSpacer_7
- QSpacerItem * horizontalSpacer_6
- QWidget * programTab
- QGridLayout * gridLayout_4
- QSpacerItem * verticalSpacer
- QSpacerItem * horizontalSpacer
- QCheckBox * programKpButton
- QLineEdit * programKpBox
- QSpacerItem * horizontalSpacer_11
- QSpacerItem * horizontalSpacer_13
- QLineEdit * programKrBox
- QCheckBox * programKrButton
- QSpacerItem * horizontalSpacer_12
- QWidget * programKpExtra
- QHBoxLayout * horizontalLayout_2
- QCheckBox * programKpSIBox
- QComboBox * programKpModeBox
- QSpacerItem * horizontalSpacer_15

- QWidget * programKrExtra
- QHBoxLayout * horizontalLayout_3
- QCheckBox * programKrSIBox
- QComboBox * programKrModeBox
- QSpacerItem * horizontalSpacer_16
- QLabel * progTipLabel
- QWidget * macroTab
- QWidget * layoutWidget1
- QGridLayout * gridLayout_8
- QGroupBox * groupBox_2
- QPlainTextEdit * pteMacroBox
- QPlainTextEdit * pteMacroComment
- QPlainTextEdit * pteMacroText
- QWidget * layoutWidget
- QVBoxLayout * verticalLayout_2
- QGroupBox * groupBox_3
- QGroupBox * groupBox_4
- QWidget * layoutWidget_2
- QGridLayout * gridLayout_10
- QPushButton * btnStopMacro
- QPushButton * btnClearMacro
- QPushButton * btnStartMacro
- QLabel * label
- QLabel * lbl_macro
- QWidget * layoutWidget2
- QHBoxLayout * delayButtons
- QRadioButton * rb_delay_no
- QRadioButton * rb_delay_default
- QRadioButton * rb_delay_asTyped
- QLabel * txtBuffer
- QHBoxLayout * horizontalLayout
- QSpacerItem * horizontalSpacer_14
- QPushButton * unbindButton
- QPushButton * resetButton
- QPushButton * cancelButton
- QPushButton * applyButton

12.79.1 Detailed Description

Definition at line 1062 of file ui_rebindwidget.h.

12.79.2 Member Function Documentation

12.79.2.1 void Ui_RebindWidget::retranslateUi (QWidget * *RebindWidget*) [inline], [inherited]

Definition at line 901 of file ui_rebindwidget.h.

References Ui_RebindWidget::actionTab, Ui_RebindWidget::animBox, Ui_RebindWidget::animButton, Ui_RebindWidget::animKrBox, Ui_RebindWidget::animOnceBox, Ui_RebindWidget::animTab, Ui_RebindWidget::applyButton, Ui_RebindWidget::btnClearMacro, Ui_RebindWidget::btnStartMacro, Ui_RebindWidget::btnStopMacro, Ui_RebindWidget::cancelButton, Ui_RebindWidget::dpiBox, Ui_RebindWidget::dpiButton, Ui_RebindWidget::dpiCustLabel, Ui_RebindWidget::dpiWarning, Ui_RebindWidget::fnBox, Ui_RebindWidget::fnButton, Ui_RebindWidget::groupBox_2, Ui_RebindWidget::groupBox_3, Ui_RebindWidget::groupBox_4, Ui_RebindWidget::keyTab, Ui_RebindWidget::label, Ui_RebindWidget::lbl_macro, Ui_RebindWidget::lightBox, Ui_RebindWidget::lightButton,

Ui_RebindWidget::lightWrapBox, Ui_RebindWidget::lockBox, Ui_RebindWidget::lockButton, Ui_RebindWidget::macroTab, Ui_RebindWidget::mb2Box, Ui_RebindWidget::mb2Button, Ui_RebindWidget::mbBox, Ui_RebindWidget::mbButton, Ui_RebindWidget::mediaBox, Ui_RebindWidget::mediaButton, Ui_RebindWidget::modBox, Ui_RebindWidget::modButton, Ui_RebindWidget::modeBox, Ui_RebindWidget::modeButton, Ui_RebindWidget::modeWrapBox, Ui_RebindWidget::mouseTab, Ui_RebindWidget::numBox, Ui_RebindWidget::numButton, Ui_RebindWidget::programKpButton, Ui_RebindWidget::programKpModeBox, Ui_RebindWidget::programKpSIBox, Ui_RebindWidget::programKrButton, Ui_RebindWidget::programKrModeBox, Ui_RebindWidget::programKrSIBox, Ui_RebindWidget::programTab, Ui_RebindWidget::progTipLabel, Ui_RebindWidget::pteMacroComment, Ui_RebindWidget::pteMacroText, Ui_RebindWidget::rb_delay_asTyped, Ui_RebindWidget::rb_delay_default, Ui_RebindWidget::rb_delay_no, Ui_RebindWidget::resetButton, Ui_RebindWidget::tabWidget, Ui_RebindWidget::txtBuffer, Ui_RebindWidget::typingBox, Ui_RebindWidget::typingButton, Ui_RebindWidget::unbindButton, Ui_RebindWidget::wheelBox, and Ui_RebindWidget::wheelButton.

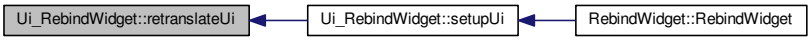
Referenced by Ui_RebindWidget::setupUi().

```
902     {
903         numBox->clear();
904         numBox->insertItems(0, QStringList()
905          << QString()
906         );
907         typingBox->clear();
908         typingBox->insertItems(0, QStringList()
909          << QString()
910         );
911         fnBox->clear();
912         fnBox->insertItems(0, QStringList()
913          << QString()
914         );
915         typingButton->setText(QApplication::translate("RebindWidget", "Typing:", Q_NULLPTR));
916         modBox->clear();
917         modBox->insertItems(0, QStringList()
918          << QString()
919         );
920         mediaButton->setText(QApplication::translate("RebindWidget", "Media:", Q_NULLPTR));
921         modButton->setText(QApplication::translate("RebindWidget", "Modifier:", Q_NULLPTR));
922         fnButton->setText(QApplication::translate("RebindWidget", "Miscellaneous:", Q_NULLPTR));
923         mediaBox->clear();
924         mediaBox->insertItems(0, QStringList()
925          << QString()
926         );
927         numButton->setText(QApplication::translate("RebindWidget", "Number pad:", Q_NULLPTR));
928         tabWidget->setTabText(tabWidget->indexOf(keyTab), QApplication::translate("
    RebindWidget", "Keyboard", Q_NULLPTR));
929         dpiButton->setText(QApplication::translate("RebindWidget", "Change DPI:", Q_NULLPTR));
930         mb2Box->clear();
931         mb2Box->insertItems(0, QStringList()
932          << QString()
933          << QApplication::translate("RebindWidget", "4 (Back)", Q_NULLPTR)
934          << QApplication::translate("RebindWidget", "5 (Forward)", Q_NULLPTR)
935          << QApplication::translate("RebindWidget", "6", Q_NULLPTR)
936          << QApplication::translate("RebindWidget", "7", Q_NULLPTR)
937          << QApplication::translate("RebindWidget", "8", Q_NULLPTR)
938         );
939         dpiBox->clear();
940         dpiBox->insertItems(0, QStringList()
941          << QString()
942          << QApplication::translate("RebindWidget", "(Cycle Up)", Q_NULLPTR)
943          << QApplication::translate("RebindWidget", "(Cycle Down)", Q_NULLPTR)
944          << QApplication::translate("RebindWidget", "(Up)", Q_NULLPTR)
945          << QApplication::translate("RebindWidget", "(Down)", Q_NULLPTR)
946          << QApplication::translate("RebindWidget", "Sniper", Q_NULLPTR)
947          << QApplication::translate("RebindWidget", "1:", Q_NULLPTR)
948          << QApplication::translate("RebindWidget", "2:", Q_NULLPTR)
949          << QApplication::translate("RebindWidget", "3:", Q_NULLPTR)
950          << QApplication::translate("RebindWidget", "4:", Q_NULLPTR)
951          << QApplication::translate("RebindWidget", "5:", Q_NULLPTR)
952          << QApplication::translate("RebindWidget", "Custom:", Q_NULLPTR)
953         );
954         mbBox->clear();
955         mbBox->insertItems(0, QStringList()
956          << QString()
957          << QApplication::translate("RebindWidget", "Left", Q_NULLPTR)
958          << QApplication::translate("RebindWidget", "Right", Q_NULLPTR)
959          << QApplication::translate("RebindWidget", "Middle", Q_NULLPTR)
960         );
961         wheelBox->clear();
962         wheelBox->insertItems(0, QStringList()
963          << QString()
964          << QApplication::translate("RebindWidget", "Up", Q_NULLPTR)
965          << QApplication::translate("RebindWidget", "Down", Q_NULLPTR)
966         );
```

```
967         mbButton->setText(QApplication::translate("RebindWidget", "Standard button:", Q_NULLPTR));
968         dpiWarning->setText(QApplication::translate("RebindWidget", "DPI is only available for
     mice.", Q_NULLPTR));
969         wheelButton->setText(QApplication::translate("RebindWidget", "Wheel:", Q_NULLPTR));
970         mb2Button->setText(QApplication::translate("RebindWidget", "Special button:", Q_NULLPTR));
971         dpiCustLabel->setText(QApplication::translate("RebindWidget", "x", Q_NULLPTR));
972         tabWidget->setTabText(tabWidget->indexOf(mouseTab),
    QApplication::translate("RebindWidget", "Mouse", Q_NULLPTR));
973         animBox->clear();
974         animBox->insertItems(0, QStringList()
975          << QString()
976         );
977         animKrBox->setText(QApplication::translate("RebindWidget", "Stop on key release",
    Q_NULLPTR));
978         animOnceBox->setText(QApplication::translate("RebindWidget", "Restart every time",
    Q_NULLPTR));
979         animButton->setText(QApplication::translate("RebindWidget", "Start animation on keypress:
    ", Q_NULLPTR));
980         tabWidget->setTabText(tabWidget->indexOf(animTab), QApplication::translate
    ("RebindWidget", "Animation", Q_NULLPTR));
981         modeButton->setText(QApplication::translate("RebindWidget", "Switch to mode:", Q_NULLPTR)
    );
982         modeWrapBox->setText(QApplication::translate("RebindWidget", "Wrap-around", Q_NULLPTR));
983         lightWrapBox->setText(QApplication::translate("RebindWidget", "Wrap-around", Q_NULLPTR)
    );
984         lockButton->setText(QApplication::translate("RebindWidget", "Windows lock:", Q_NULLPTR));
985         lightBox->clear();
986         lightBox->insertItems(0, QStringList()
987          << QString()
988          << QApplication::translate("RebindWidget", "Up", Q_NULLPTR)
989          << QApplication::translate("RebindWidget", "Down", Q_NULLPTR)
990         );
991         lockBox->clear();
992         lockBox->insertItems(0, QStringList()
993          << QString()
994          << QApplication::translate("RebindWidget", "Toggle", Q_NULLPTR)
995          << QApplication::translate("RebindWidget", "On", Q_NULLPTR)
996          << QApplication::translate("RebindWidget", "Off", Q_NULLPTR)
997         );
998         lightButton->setText(QApplication::translate("RebindWidget", "Brightness:", Q_NULLPTR));
999         modeBox->clear();
1000         modeBox->insertItems(0, QStringList()
1001          << QString()
1002          << QApplication::translate("RebindWidget", "(Previous)", Q_NULLPTR)
1003          << QApplication::translate("RebindWidget", "(Next)", Q_NULLPTR)
1004         );
1005         tabWidget->setTabText(tabWidget->indexOf(actionTab),
    QApplication::translate("RebindWidget", "Special", Q_NULLPTR));
1006         programKpButton->setText(QApplication::translate("RebindWidget", "Launch program on
     key press:", Q_NULLPTR));
1007         programKrButton->setText(QApplication::translate("RebindWidget", "Launch program on
     key release:", Q_NULLPTR));
1008         programKpSIBox->setText(QApplication::translate("RebindWidget", "Single instance",
    Q_NULLPTR));
1009         programKpModeBox->clear();
1010         programKpModeBox->insertItems(0, QStringList()
1011          << QApplication::translate("RebindWidget", "Run indefinitely", Q_NULLPTR)
1012          << QApplication::translate("RebindWidget", "Stop on release", Q_NULLPTR)
1013          << QApplication::translate("RebindWidget", "Stop on re-press", Q_NULLPTR)
1014         );
1015         programKrSIBox->setText(QApplication::translate("RebindWidget", "Single instance",
    Q_NULLPTR));
1016         programKrModeBox->clear();
1017         programKrModeBox->insertItems(0, QStringList()
1018          << QApplication::translate("RebindWidget", "Run indefinitely", Q_NULLPTR)
1019          << QApplication::translate("RebindWidget", "Stop on re-press", Q_NULLPTR)
1020         );
1021         progTipLabel->setText(QApplication::translate("RebindWidget", "Tip", Q_NULLPTR));
1022         tabWidget->setTabText(tabWidget->indexOf(programTab),
    QApplication::translate("RebindWidget", "Program", Q_NULLPTR));
1023         groupBox_2->setTitle(QApplication::translate("RebindWidget", "Macro Key Actions",
    Q_NULLPTR));
1024 #ifndef QT_NO_TOOLTIP
1025         pteMacroComment->setToolTip(QApplication::translate("RebindWidget", "What is this
     macro for?", Q_NULLPTR));
1026 #endif // QT_NO_TOOLTIP
1027         pteMacroComment->setDocumentTitle(QString());
1028         pteMacroComment->setPlainText(QString());
1029         pteMacroText->setDocumentTitle(QString());
1030         pteMacroText->setPlainText(QString());
1031         groupBox_3->setTitle(QApplication::translate("RebindWidget", "Macro Comment", Q_NULLPTR))
    ;
1032         groupBox_4->setTitle(QApplication::translate("RebindWidget", "Macro Text", Q_NULLPTR));
1033         btnStopMacro->setText(QApplication::translate("RebindWidget", "Stop", Q_NULLPTR));
1034         btnClearMacro->setText(QApplication::translate("RebindWidget", "Clear", Q_NULLPTR));
1035         btnStartMacro->setText(QApplication::translate("RebindWidget", "Start", Q_NULLPTR));
1036         label->setText(QApplication::translate("RebindWidget", "Record macro ", Q_NULLPTR));
```

```
1037         lbl_macro->setText(QApplication::translate("RebindWidget", "Comment label for help",
     Q_NULLPTR));
1038 #ifndef QT_NO_WHATSTHIS
1039         rb_delay_no->setWhatsThis(QApplication::translate("RebindWidget", "configure delays
     between keystrokes: Disable delay", Q_NULLPTR));
1040 #endif // QT_NO_WHATSTHIS
1041         rb_delay_no->setText(QApplication::translate("RebindWidget", "No delay", Q_NULLPTR));
1042 #ifndef QT_NO_WHATSTHIS
1043         rb_delay_default->setWhatsThis(QApplication::translate("RebindWidget", "configure
     delays between keystrokes: Set delay to default values: 20us up to 15 chars, 200us above", Q_NULLPTR));
1044 #endif // QT_NO_WHATSTHIS
1045         rb_delay_default->setText(QApplication::translate("RebindWidget", "default",
     Q_NULLPTR));
1046 #ifndef QT_NO_WHATSTHIS
1047         rb_delay_asTyped->setWhatsThis(QApplication::translate("RebindWidget", "configure
     delays between keystrokes: Remember delays as they had been typed while the macro definition process",
     Q_NULLPTR));
1048 #endif // QT_NO_WHATSTHIS
1049         rb_delay_asTyped->setText(QApplication::translate("RebindWidget", "as typed",
     Q_NULLPTR));
1050         txtBuffer->setText(QString());
1051         tabWidget->setTabText(tabWidget->indexOf(macroTab),
     QApplication::translate("RebindWidget", "Macro", Q_NULLPTR));
1052         unbindButton->setText(QApplication::translate("RebindWidget", "Unbind", Q_NULLPTR));
1053         resetButton->setText(QApplication::translate("RebindWidget", "Reset to Default",
     Q_NULLPTR));
1054         cancelButton->setText(QApplication::translate("RebindWidget", "Cancel", Q_NULLPTR));
1055         applyButton->setText(QApplication::translate("RebindWidget", "Apply", Q_NULLPTR));
1056         Q_UNUSED(RebindWidget);
1057     } // retranslateUi
```

Here is the caller graph for this function:



12.79.2.2 void Ui_RebindWidget::setupUi (QWidget * *RebindWidget*) [inline],[inherited]

Definition at line 163 of file ui_rebindwidget.h.

References Ui_RebindWidget::actionTab, Ui_RebindWidget::animBox, Ui_RebindWidget::animButton, Ui_RebindWidget::animKrBox, Ui_RebindWidget::animOnceBox, Ui_RebindWidget::animTab, Ui_RebindWidget::applyButton, Ui_RebindWidget::btnClearMacro, Ui_RebindWidget::btnStartMacro, Ui_RebindWidget::btnStopMacro, Ui_RebindWidget::cancelButton, Ui_RebindWidget::delayButtons, Ui_RebindWidget::dpiBox, Ui_RebindWidget::dpiButton, Ui_RebindWidget::dpiCustLabel, Ui_RebindWidget::dpiCustXBox, Ui_RebindWidget::dpiCustYBox, Ui_RebindWidget::dpiWarning, Ui_RebindWidget::fnBox, Ui_RebindWidget::fnButton, Ui_RebindWidget::gridLayout, Ui_RebindWidget::gridLayout_10, Ui_RebindWidget::gridLayout_2, Ui_RebindWidget::gridLayout_3, Ui_RebindWidget::gridLayout_4, Ui_RebindWidget::gridLayout_5, Ui_RebindWidget::gridLayout_8, Ui_RebindWidget::groupBox_2, Ui_RebindWidget::groupBox_3, Ui_RebindWidget::groupBox_4, Ui_RebindWidget::horizontalLayout, Ui_RebindWidget::horizontalLayout_2, Ui_RebindWidget::horizontalLayout_3, Ui_RebindWidget::horizontalSpacer, Ui_RebindWidget::horizontalSpacer_10, Ui_RebindWidget::horizontalSpacer_11, Ui_RebindWidget::horizontalSpacer_12, Ui_RebindWidget::horizontalSpacer_13, Ui_RebindWidget::horizontalSpacer_14, Ui_RebindWidget::horizontalSpacer_15, Ui_RebindWidget::horizontalSpacer_16, Ui_RebindWidget::horizontalSpacer_17, Ui_RebindWidget::horizontalSpacer_18, Ui_RebindWidget::horizontalSpacer_19, Ui_RebindWidget::horizontalSpacer_2, Ui_RebindWidget::horizontalSpacer_20, Ui_RebindWidget::horizontalSpacer_21, Ui_RebindWidget::horizontalSpacer_22, Ui_RebindWidget::horizontalSpacer_23, Ui_RebindWidget::horizontalSpacer_24, Ui_RebindWidget::horizontalSpacer_25, Ui_RebindWidget::horizontalSpacer_26, Ui_RebindWidget::horizontalSpacer_3, Ui_RebindWidget::horizontalSpacer_4, Ui_RebindWidget::horizontalSpacer_5, Ui_RebindWidget::horizontalSpacer_6, Ui_RebindWidget::horizontalSpacer_7, Ui_RebindWidget::horizontalSpacer_8, Ui_RebindWidget::horizontalSpacer_9, Ui_RebindWidget::keyTab, Ui_RebindWidget::label, Ui_RebindWidget::layoutWidget, Ui_RebindWidget::layoutWidget1, Ui_RebindWidget::layoutWidget2, Ui_RebindWidget::layoutWidget_2, Ui_RebindWidget::lbl_macro, Ui_RebindWidget::lightBox, Ui_RebindWidget::lightButton, Ui_RebindWidget::lightWrapBox, Ui_RebindWidget::lockBox, Ui_RebindWidget::lockButton, Ui_RebindWidget::macroTab, Ui_RebindWidget-

::mb2Box, Ui_RebindWidget::mb2Button, Ui_RebindWidget::mbBox, Ui_RebindWidget::mbButton, Ui_RebindWidget::mediaBox, Ui_RebindWidget::mediaButton, Ui_RebindWidget::modBox, Ui_RebindWidget::modButton, Ui_RebindWidget::modeBox, Ui_RebindWidget::modeButton, Ui_RebindWidget::modeWrapBox, Ui_RebindWidget::mouseTab, Ui_RebindWidget::numBox, Ui_RebindWidget::numButton, Ui_RebindWidget::programKpBox, Ui_RebindWidget::programKpButton, Ui_RebindWidget::programKpExtra, Ui_RebindWidget::programKpModeBox, Ui_RebindWidget::programKpSIBox, Ui_RebindWidget::programKrBox, Ui_RebindWidget::programKrButton, Ui_RebindWidget::programKrExtra, Ui_RebindWidget::programKrModeBox, Ui_RebindWidget::programKrSIBox, Ui_RebindWidget::programTab, Ui_RebindWidget::progTipLabel, Ui_RebindWidget::pteMacroBox, Ui_RebindWidget::pteMacroComment, Ui_RebindWidget::pteMacroText, Ui_RebindWidget::rb_delay_asTyped, Ui_RebindWidget::rb_delay_default, Ui_RebindWidget::rb_delay_no, Ui_RebindWidget::resetButton, Ui_RebindWidget::retranslateUi(), Ui_RebindWidget::tabWidget, Ui_RebindWidget::txtBuffer, Ui_RebindWidget::typingBox, Ui_RebindWidget::typingButton, Ui_RebindWidget::unbindButton, Ui_RebindWidget::verticalLayout, Ui_RebindWidget::verticalLayout_2, Ui_RebindWidget::verticalSpacer, Ui_RebindWidget::verticalSpacer_10, Ui_RebindWidget::verticalSpacer_11, Ui_RebindWidget::verticalSpacer_2, Ui_RebindWidget::verticalSpacer_3, Ui_RebindWidget::verticalSpacer_4, Ui_RebindWidget::verticalSpacer_5, Ui_RebindWidget::verticalSpacer_6, Ui_RebindWidget::verticalSpacer_7, Ui_RebindWidget::verticalSpacer_8, Ui_RebindWidget::wheelBox, and Ui_RebindWidget::wheelButton.

Referenced by RebindWidget::RebindWidget().

```
164     {
165         if (RebindWidget->objectName().isEmpty())
166             RebindWidget->setObjectName(QStringLiteral("RebindWidget"));
167         RebindWidget->resize(687, 342);
168         QSizePolicy sizePolicy(QSizePolicy::Expanding, QSizePolicy::Preferred);
169         sizePolicy.setHorizontalStretch(0);
170         sizePolicy.setVerticalStretch(0);
171         sizePolicy.setHeightForWidth(RebindWidget->sizePolicy().hasHeightForWidth());
172         RebindWidget->setSizePolicy(sizePolicy);
173         RebindWidget->setStyleSheet(QStringLiteral("font-size: 12px ;"));
174         verticalLayout = new QVBoxLayout(RebindWidget);
175         verticalLayout->setObjectName(QStringLiteral("verticalLayout"));
176         tabWidget = new QTabWidget(RebindWidget);
177         tabWidget->setObjectName(QStringLiteral("tabWidget"));
178         tabWidget->setStyleSheet(QStringLiteral(""));
179         tabWidget->setUsesScrollButtons(false);
180         tabWidget->setDocumentMode(false);
181         keyTab = new QWidget();
182         keyTab->setObjectName(QStringLiteral("keyTab"));
183         sizePolicy.setHeightForWidth(keyTab->sizePolicy().hasHeightForWidth());
184         keyTab->setSizePolicy(sizePolicy);
185         gridLayout_2 = new QGridLayout(keyTab);
186         gridLayout_2->setObjectName(QStringLiteral("gridLayout_2"));
187         gridLayout_2->setContentsMargins(-1, -1, -1, 0);
188         numBox = new QComboBox(keyTab);
189         numBox->setObjectName(QStringLiteral("numBox"));
190         QSizePolicy sizePolicy1(QSizePolicy::Expanding, QSizePolicy::Fixed);
191         sizePolicy1.setHorizontalStretch(0);
192         sizePolicy1.setVerticalStretch(0);
193         sizePolicy1.setHeightForWidth(numBox->sizePolicy().hasHeightForWidth());
194         numBox->setSizePolicy(sizePolicy1);
195 
196         gridLayout_2->addWidget(numBox, 6, 4, 1, 1);
197 
198         typingBox = new QComboBox(keyTab);
199         typingBox->setObjectName(QStringLiteral("typingBox"));
200         sizePolicy1.setHeightForWidth(typingBox->sizePolicy().hasHeightForWidth());
201         typingBox->setSizePolicy(sizePolicy1);
202 
203         gridLayout_2->addWidget(typingBox, 1, 1, 1, 1);
204 
205         fnBox = new QComboBox(keyTab);
206         fnBox->setObjectName(QStringLiteral("fnBox"));
207         sizePolicy1.setHeightForWidth(fnBox->sizePolicy().hasHeightForWidth());
208         fnBox->setSizePolicy(sizePolicy1);
209 
210         gridLayout_2->addWidget(fnBox, 6, 1, 1, 1);
211 
212         verticalSpacer_10 = new QSpacerItem(0, 5, QSizePolicy::Minimum, QSizePolicy::Fixed
    );
213 
214         gridLayout_2->addItem(verticalSpacer_10, 9, 1, 1, 1);
215 
216         verticalSpacer_3 = new QSpacerItem(20, 40, QSizePolicy::Minimum,
    QSizePolicy::Expanding);
217 
218         gridLayout_2->addItem(verticalSpacer_3, 15, 2, 1, 1);
219 
220         horizontalSpacer_2 = new QSpacerItem(5, 20, QSizePolicy::Fixed,
    QSizePolicy::Minimum);
```

```
221
222         gridLayout_2->addItem(horizontalSpacer_2, 1, 3, 1, 1);
223
224         typingButton = new QRadioButton(keyTab);
225         typingButton->setObjectName(QStringLiteral("typingButton"));
226         typingButton->setAutoExclusive(false);
227
228         gridLayout_2->addWidget(typingButton, 0, 0, 1, 2);
229
230         modBox = new QComboBox(keyTab);
231         modBox->setObjectName(QStringLiteral("modBox"));
232         sizePolicy1.setHeightForWidth(modBox->sizePolicy().hasHeightForWidth());
233         modBox->setSizePolicy(sizePolicy1);
234
235         gridLayout_2->addWidget(modBox, 1, 4, 1, 1);
236
237         horizontalSpacer_3 = new QSpacerItem(5, 20, QSizePolicy::Fixed,
    QSizePolicy::Minimum);
238
239         gridLayout_2->addItem(horizontalSpacer_3, 1, 0, 1, 1);
240
241         mediaButton = new QRadioButton(keyTab);
242         mediaButton->setObjectName(QStringLiteral("mediaButton"));
243         mediaButton->setAutoExclusive(false);
244
245         gridLayout_2->addWidget(mediaButton, 10, 0, 1, 2);
246
247         verticalSpacer_4 = new QSpacerItem(20, 5, QSizePolicy::Minimum, QSizePolicy::Fixed)
    ;
248
249         gridLayout_2->addItem(verticalSpacer_4, 2, 1, 1, 1);
250
251         horizontalSpacer_9 = new QSpacerItem(0, 20, QSizePolicy::Expanding,
    QSizePolicy::Minimum);
252
253         gridLayout_2->addItem(horizontalSpacer_9, 1, 5, 1, 2);
254
255         horizontalSpacer_4 = new QSpacerItem(0, 20, QSizePolicy::Expanding,
    QSizePolicy::Minimum);
256
257         gridLayout_2->addItem(horizontalSpacer_4, 1, 2, 1, 1);
258
259         modButton = new QRadioButton(keyTab);
260         modButton->setObjectName(QStringLiteral("modButton"));
261
262         gridLayout_2->addWidget(modButton, 0, 3, 1, 2);
263
264         fnButton = new QRadioButton(keyTab);
265         fnButton->setObjectName(QStringLiteral("fnButton"));
266         fnButton->setAutoExclusive(false);
267
268         gridLayout_2->addWidget(fnButton, 5, 0, 1, 2);
269
270         mediaBox = new QComboBox(keyTab);
271         mediaBox->setObjectName(QStringLiteral("mediaBox"));
272         sizePolicy1.setHeightForWidth(mediaBox->sizePolicy().hasHeightForWidth());
273         mediaBox->setSizePolicy(sizePolicy1);
274
275         gridLayout_2->addWidget(mediaBox, 11, 1, 1, 1);
276
277         numButton = new QRadioButton(keyTab);
278         numButton->setObjectName(QStringLiteral("numButton"));
279         numButton->setAutoExclusive(false);
280
281         gridLayout_2->addWidget(numButton, 5, 3, 1, 2);
282
283         horizontalSpacer_17 = new QSpacerItem(40, 20, QSizePolicy::Expanding,
    QSizePolicy::Minimum);
284
285         gridLayout_2->addItem(horizontalSpacer_17, 1, 7, 1, 1);
286
287         tabWidget->addTab(keyTab, QString());
288         mouseTab = new QWidget();
289         mouseTab->setObjectName(QStringLiteral("mouseTab"));
290         gridLayout = new QGridLayout(mouseTab);
291         gridLayout->setObjectName(QStringLiteral("gridLayout"));
292         dpiButton = new QRadioButton(mouseTab);
293         dpiButton->setObjectName(QStringLiteral("dpiButton"));
294         dpiButton->setAutoExclusive(false);
295
296         gridLayout->addWidget(dpiButton, 3, 3, 1, 5);
297
298         verticalSpacer_2 = new QSpacerItem(20, 40, QSizePolicy::Minimum,
    QSizePolicy::Expanding);
299
300         gridLayout->addItem(verticalSpacer_2, 5, 1, 1, 1);
301
```

```
302         verticalSpacer_8 = new QSpacerItem(20, 5, QSizePolicy::Minimum, QSizePolicy::Fixed)
    ;
303
304         gridLayout->addItem(verticalSpacer_8, 2, 1, 1, 1);
305
306         horizontalSpacer_18 = new QSpacerItem(5, 20, QSizePolicy::Fixed,
    QSizePolicy::Minimum);
307
308         gridLayout->addItem(horizontalSpacer_18, 1, 0, 1, 1);
309
310         horizontalSpacer_20 = new QSpacerItem(5, 20, QSizePolicy::Fixed,
    QSizePolicy::Minimum);
311
312         gridLayout->addItem(horizontalSpacer_20, 1, 3, 1, 1);
313
314         mb2Box = new QComboBox(mouseTab);
315         mb2Box->setObjectName(QStringLiteral("mb2Box"));
316         sizePolicy1.setHeightForWidth(mb2Box->sizePolicy().hasHeightForWidth());
317         mb2Box->setSizePolicy(sizePolicy1);
318
319         gridLayout->addWidget(mb2Box, 1, 4, 1, 4);
320
321         dpiBox = new QComboBox(mouseTab);
322         dpiBox->setObjectName(QStringLiteral("dpiBox"));
323
324         gridLayout->addWidget(dpiBox, 4, 4, 1, 1);
325
326         dpiCustYBox = new QSpinBox(mouseTab);
327         dpiCustYBox->setObjectName(QStringLiteral("dpiCustYBox"));
328         dpiCustYBox->setMinimum(400);
329         dpiCustYBox->setMaximum(8500);
330         dpiCustYBox->setSingleStep(50);
331
332         gridLayout->addWidget(dpiCustYBox, 4, 7, 1, 1);
333
334         mbBox = new QComboBox(mouseTab);
335         mbBox->setObjectName(QStringLiteral("mbBox"));
336         sizePolicy1.setHeightForWidth(mbBox->sizePolicy().hasHeightForWidth());
337         mbBox->setSizePolicy(sizePolicy1);
338
339         gridLayout->addWidget(mbBox, 1, 1, 1, 1);
340
341         wheelBox = new QComboBox(mouseTab);
342         wheelBox->setObjectName(QStringLiteral("wheelBox"));
343
344         gridLayout->addWidget(wheelBox, 4, 1, 1, 1);
345
346         mbButton = new QRadioButton(mouseTab);
347         mbButton->setObjectName(QStringLiteral("mbButton"));
348         mbButton->setAutoExclusive(false);
349
350         gridLayout->addWidget(mbButton, 0, 0, 1, 2);
351
352         dpiCustXBox = new QSpinBox(mouseTab);
353         dpiCustXBox->setObjectName(QStringLiteral("dpiCustXBox"));
354         dpiCustXBox->setMinimum(400);
355         dpiCustXBox->setMaximum(8500);
356         dpiCustXBox->setSingleStep(50);
357
358         gridLayout->addWidget(dpiCustXBox, 4, 5, 1, 1);
359
360         dpiWarning = new QLabel(mouseTab);
361         dpiWarning->setObjectName(QStringLiteral("dpiWarning"));
362         QSizePolicy sizePolicy2(QSizePolicy::Preferred, QSizePolicy::Fixed);
363         sizePolicy2.setHorizontalStretch(0);
364         sizePolicy2.setVerticalStretch(0);
365         sizePolicy2.setHeightForWidth(dpiWarning->sizePolicy().hasHeightForWidth());
366         dpiWarning->setSizePolicy(sizePolicy2);
367
368         gridLayout->addWidget(dpiWarning, 6, 0, 1, 10);
369
370         wheelButton = new QRadioButton(mouseTab);
371         wheelButton->setObjectName(QStringLiteral("wheelButton"));
372
373         gridLayout->addWidget(wheelButton, 3, 0, 1, 2);
374
375         horizontalSpacer_22 = new QSpacerItem(5, 20, QSizePolicy::Fixed,
    QSizePolicy::Minimum);
376
377         gridLayout->addItem(horizontalSpacer_22, 1, 8, 1, 1);
378
379         mb2Button = new QRadioButton(mouseTab);
380         mb2Button->setObjectName(QStringLiteral("mb2Button"));
381         mb2Button->setAutoExclusive(false);
382
383         gridLayout->addWidget(mb2Button, 0, 3, 1, 5);
384
```

```
385         dpiCustLabel = new QLabel(mouseTab);
386         dpiCustLabel->setObjectName(QStringLiteral("dpiCustLabel"));
387 
388         gridLayout->addWidget(dpiCustLabel, 4, 6, 1, 1);
389 
390         horizontalSpacer_19 = new QSpacerItem(0, 0, QSizePolicy::Expanding,
    QSizePolicy::Minimum);
391 
392         gridLayout->addItem(horizontalSpacer_19, 1, 2, 1, 1);
393 
394         horizontalSpacer_21 = new QSpacerItem(0, 20, QSizePolicy::Expanding,
    QSizePolicy::Minimum);
395 
396         gridLayout->addItem(horizontalSpacer_21, 1, 9, 1, 1);
397 
398         tabWidget->addTab(mouseTab, QString());
399         animTab = new QWidget();
400         animTab->setObjectName(QStringLiteral("animTab"));
401         gridLayout_5 = new QGridLayout(animTab);
402         gridLayout_5->setObjectName(QStringLiteral("gridLayout_5"));
403         verticalSpacer_11 = new QSpacerItem(20, 40, QSizePolicy::Minimum,
    QSizePolicy::Expanding);
404 
405         gridLayout_5->addItem(verticalSpacer_11, 6, 2, 1, 1);
406 
407         animBox = new QComboBox(animTab);
408         animBox->setObjectName(QStringLiteral("animBox"));
409         sizePolicy1.setHeightForWidth(animBox->sizePolicy().hasHeightForWidth());
410         animBox->setSizePolicy(sizePolicy1);
411 
412         gridLayout_5->addWidget(animBox, 4, 2, 1, 1);
413 
414         horizontalSpacer_24 = new QSpacerItem(20, 20, QSizePolicy::Fixed,
    QSizePolicy::Minimum);
415 
416         gridLayout_5->addItem(horizontalSpacer_24, 4, 3, 1, 1);
417 
418         horizontalSpacer_25 = new QSpacerItem(438, 20, QSizePolicy::Expanding,
    QSizePolicy::Minimum);
419 
420         gridLayout_5->addItem(horizontalSpacer_25, 4, 7, 1, 1);
421 
422         animKrBox = new QCheckBox(animTab);
423         animKrBox->setObjectName(QStringLiteral("animKrBox"));
424 
425         gridLayout_5->addWidget(animKrBox, 4, 6, 1, 1);
426 
427         animOnceBox = new QCheckBox(animTab);
428         animOnceBox->setObjectName(QStringLiteral("animOnceBox"));
429 
430         gridLayout_5->addWidget(animOnceBox, 4, 4, 1, 1);
431 
432         horizontalSpacer_26 = new QSpacerItem(20, 20, QSizePolicy::Fixed,
    QSizePolicy::Minimum);
433 
434         gridLayout_5->addItem(horizontalSpacer_26, 4, 5, 1, 1);
435 
436         animButton = new QRadioButton(animTab);
437         animButton->setObjectName(QStringLiteral("animButton"));
438         sizePolicy1.setHeightForWidth(animButton->sizePolicy().hasHeightForWidth());
439         animButton->setSizePolicy(sizePolicy1);
440 
441         gridLayout_5->addWidget(animButton, 0, 1, 1, 2);
442 
443         horizontalSpacer_23 = new QSpacerItem(5, 20, QSizePolicy::Fixed,
    QSizePolicy::Minimum);
444 
445         gridLayout_5->addItem(horizontalSpacer_23, 4, 1, 1, 1);
446 
447         tabWidget->addTab(animTab, QString());
448         actionTab = new QWidget();
449         actionTab->setObjectName(QStringLiteral("actionTab"));
450         sizePolicy.setHeightForWidth(actionTab->sizePolicy().hasHeightForWidth());
451         actionTab->setSizePolicy(sizePolicy);
452         gridLayout_3 = new QGridLayout(actionTab);
453         gridLayout_3->setObjectName(QStringLiteral("gridLayout_3"));
454         gridLayout_3->setContentsMargins(-1, -1, -1, 0);
455         horizontalSpacer_5 = new QSpacerItem(5, 20, QSizePolicy::Fixed,
    QSizePolicy::Minimum);
456 
457         gridLayout_3->addItem(horizontalSpacer_5, 1, 0, 1, 1);
458 
459         modeButton = new QRadioButton(actionTab);
460         modeButton->setObjectName(QStringLiteral("modeButton"));
461         modeButton->setAutoExclusive(false);
462 
463         gridLayout_3->addWidget(modeButton, 0, 0, 1, 2);
```

```
464
465         modeWrapBox = new QCheckBox(actionTab);
466         modeWrapBox->setObjectName(QStringLiteral("modeWrapBox"));
467         modeWrapBox->setChecked(true);
468
469         gridLayout_3->addWidget(modeWrapBox, 1, 3, 1, 1);
470
471         lightWrapBox = new QCheckBox(actionTab);
472         lightWrapBox->setObjectName(QStringLiteral("lightWrapBox"));
473         lightWrapBox->setChecked(true);
474
475         gridLayout_3->addWidget(lightWrapBox, 4, 3, 1, 1);
476
477         verticalSpacer_7 = new QSpacerItem(20, 5, QSizePolicy::Minimum, QSizePolicy::Fixed)
    ;
478
479         gridLayout_3->addItem(verticalSpacer_7, 2, 1, 1, 1);
480
481         verticalSpacer_6 = new QSpacerItem(20, 5, QSizePolicy::Minimum, QSizePolicy::Fixed)
    ;
482
483         gridLayout_3->addItem(verticalSpacer_6, 5, 1, 1, 1);
484
485         lockButton = new QRadioButton(actionTab);
486         lockButton->setObjectName(QStringLiteral("lockButton"));
487         lockButton->setAutoExclusive(false);
488
489         gridLayout_3->addWidget(lockButton, 6, 0, 1, 2);
490
491         verticalSpacer_5 = new QSpacerItem(20, 40, QSizePolicy::Minimum,
    QSizePolicy::Expanding);
492
493         gridLayout_3->addItem(verticalSpacer_5, 8, 1, 1, 1);
494
495         lightBox = new QComboBox(actionTab);
496         lightBox->setObjectName(QStringLiteral("lightBox"));
497         sizePolicy1.setHeightForWidth(lightBox->sizePolicy().hasHeightForWidth());
498         lightBox->setSizePolicy(sizePolicy1);
499
500         gridLayout_3->addWidget(lightBox, 4, 1, 1, 1);
501
502         lockBox = new QComboBox(actionTab);
503         lockBox->setObjectName(QStringLiteral("lockBox"));
504         sizePolicy1.setHeightForWidth(lockBox->sizePolicy().hasHeightForWidth());
505         lockBox->setSizePolicy(sizePolicy1);
506
507         gridLayout_3->addWidget(lockBox, 7, 1, 1, 1);
508
509         lightButton = new QRadioButton(actionTab);
510         lightButton->setObjectName(QStringLiteral("lightButton"));
511         lightButton->setAutoExclusive(false);
512
513         gridLayout_3->addWidget(lightButton, 3, 0, 1, 2);
514
515         modeBox = new QComboBox(actionTab);
516         modeBox->setObjectName(QStringLiteral("modeBox"));
517         sizePolicy1.setHeightForWidth(modeBox->sizePolicy().hasHeightForWidth());
518         modeBox->setSizePolicy(sizePolicy1);
519
520         gridLayout_3->addWidget(modeBox, 1, 1, 1, 1);
521
522         horizontalSpacer_8 = new QSpacerItem(5, 20, QSizePolicy::Fixed,
    QSizePolicy::Minimum);
523
524         gridLayout_3->addItem(horizontalSpacer_8, 1, 2, 1, 1);
525
526         horizontalSpacer_10 = new QSpacerItem(40, 20, QSizePolicy::Expanding,
    QSizePolicy::Minimum);
527
528         gridLayout_3->addItem(horizontalSpacer_10, 7, 3, 1, 1);
529
530         horizontalSpacer_7 = new QSpacerItem(40, 20, QSizePolicy::Expanding,
    QSizePolicy::Minimum);
531
532         gridLayout_3->addItem(horizontalSpacer_7, 1, 4, 1, 1);
533
534         horizontalSpacer_6 = new QSpacerItem(40, 20, QSizePolicy::Expanding,
    QSizePolicy::Minimum);
535
536         gridLayout_3->addItem(horizontalSpacer_6, 1, 5, 1, 1);
537
538         tabWidget->addTab(actionTab, QString());
539         programTab = new QWidget();
540         programTab->setObjectName(QStringLiteral("programTab"));
541         gridLayout_4 = new QGridLayout(programTab);
542         gridLayout_4->setObjectName(QStringLiteral("gridLayout_4"));
543         verticalSpacer = new QSpacerItem(20, 5, QSizePolicy::Minimum, QSizePolicy::Fixed);
```

```
544
545         gridLayout_4->addItem(verticalSpacer, 2, 1, 1, 1);
546
547         horizontalSpacer = new QSpacerItem(5, 20, QSizePolicy::Fixed, QSizePolicy::Minimum)
    ;
548
549         gridLayout_4->addItem(horizontalSpacer, 1, 0, 1, 1);
550
551         programKpButton = new QCheckBox(programTab);
552         programKpButton->setObjectName(QStringLiteral("programKpButton"));
553
554         gridLayout_4->addWidget(programKpButton, 0, 0, 1, 2);
555
556         programKpBox = new QLineEdit(programTab);
557         programKpBox->setObjectName(QStringLiteral("programKpBox"));
558
559         gridLayout_4->addWidget(programKpBox, 1, 1, 1, 1);
560
561         horizontalSpacer_11 = new QSpacerItem(40, 20, QSizePolicy::Expanding,
    QSizePolicy::Minimum);
562
563         gridLayout_4->addItem(horizontalSpacer_11, 0, 2, 1, 2);
564
565         horizontalSpacer_13 = new QSpacerItem(20, 20, QSizePolicy::Fixed,
    QSizePolicy::Minimum);
566
567         gridLayout_4->addItem(horizontalSpacer_13, 1, 2, 1, 1);
568
569         programKrBox = new QLineEdit(programTab);
570         programKrBox->setObjectName(QStringLiteral("programKrBox"));
571
572         gridLayout_4->addWidget(programKrBox, 4, 1, 1, 1);
573
574         programKrButton = new QCheckBox(programTab);
575         programKrButton->setObjectName(QStringLiteral("programKrButton"));
576
577         gridLayout_4->addWidget(programKrButton, 3, 0, 1, 2);
578
579         horizontalSpacer_12 = new QSpacerItem(40, 20, QSizePolicy::Expanding,
    QSizePolicy::Minimum);
580
581         gridLayout_4->addItem(horizontalSpacer_12, 0, 4, 1, 1);
582
583         programKpExtra = new QWidget(programTab);
584         programKpExtra->setObjectName(QStringLiteral("programKpExtra"));
585         horizontalLayout_2 = new QHBoxLayout(programKpExtra);
586         horizontalLayout_2->setSpacing(12);
587         horizontalLayout_2->setObjectName(QStringLiteral("horizontalLayout_2"));
588         horizontalLayout_2->setContentsMargins(0, 0, 0, 0);
589         programKpSIBox = new QCheckBox(programKpExtra);
590         programKpSIBox->setObjectName(QStringLiteral("programKpSIBox"));
591
592         horizontalLayout_2->addWidget(programKpSIBox);
593
594         programKpModeBox = new QComboBox(programKpExtra);
595         programKpModeBox->setObjectName(QStringLiteral("programKpModeBox"));
596
597         horizontalLayout_2->addWidget(programKpModeBox);
598
599         horizontalSpacer_15 = new QSpacerItem(40, 20, QSizePolicy::Expanding,
    QSizePolicy::Minimum);
600
601         horizontalLayout_2->addItem(horizontalSpacer_15);
602
603
604         gridLayout_4->addWidget(programKpExtra, 1, 3, 1, 2);
605
606         programKrExtra = new QWidget(programTab);
607         programKrExtra->setObjectName(QStringLiteral("programKrExtra"));
608         horizontalLayout_3 = new QHBoxLayout(programKrExtra);
609         horizontalLayout_3->setSpacing(12);
610         horizontalLayout_3->setObjectName(QStringLiteral("horizontalLayout_3"));
611         horizontalLayout_3->setContentsMargins(0, 0, 0, 0);
612         programKrSIBox = new QCheckBox(programKrExtra);
613         programKrSIBox->setObjectName(QStringLiteral("programKrSIBox"));
614
615         horizontalLayout_3->addWidget(programKrSIBox);
616
617         programKrModeBox = new QComboBox(programKrExtra);
618         programKrModeBox->setObjectName(QStringLiteral("programKrModeBox"));
619
620         horizontalLayout_3->addWidget(programKrModeBox);
621
622         horizontalSpacer_16 = new QSpacerItem(40, 20, QSizePolicy::Expanding,
    QSizePolicy::Minimum);
623
624         horizontalLayout_3->addItem(horizontalSpacer_16);
```

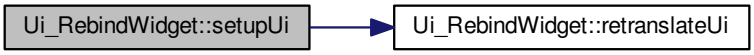
```
625
626
627         gridLayout_4->addWidget(programKrExtra, 4, 3, 1, 2);
628
629         progTipLabel = new QLabel(programTab);
630         progTipLabel->setObjectName(QStringLiteral("progTipLabel"));
631         QSizePolicy sizePolicy3(QSizePolicy::Preferred, QSizePolicy::Expanding);
632         sizePolicy3.setHorizontalStretch(0);
633         sizePolicy3.setVerticalStretch(0);
634         sizePolicy3.setHeightForWidth(progTipLabel->sizePolicy().hasHeightForWidth());
635         progTipLabel->setSizePolicy(sizePolicy3);
636         progTipLabel->setTextInteractionFlags(Qt::LinksAccessibleByMouse|
    Qt::TextSelectableByMouse);
637
638         gridLayout_4->addWidget(progTipLabel, 5, 1, 1, 4);
639
640         tabWidget->addTab(programTab, QString());
641         macroTab = new QWidget();
642         macroTab->setObjectName(QStringLiteral("macroTab"));
643         layoutWidget1 = new QWidget(macroTab);
644         layoutWidget1->setObjectName(QStringLiteral("layoutWidget1"));
645         layoutWidget1->setGeometry(QRect(40, 12, 241, 161));
646         gridLayout_8 = new QGridLayout(layoutWidget1);
647         gridLayout_8->setObjectName(QStringLiteral("gridLayout_8"));
648         gridLayout_8->setSizeConstraint(QLayout::SetNoConstraint);
649         gridLayout_8->setContentsMargins(0, 0, 0, 0);
650         groupBox_2 = new QGroupBox(layoutWidget1);
651         groupBox_2->setObjectName(QStringLiteral("groupBox_2"));
652         pteMacroBox = new QPlainTextEdit(groupBox_2);
653         pteMacroBox->setObjectName(QStringLiteral("pteMacroBox"));
654         pteMacroBox->setGeometry(QRect(0, 20, 241, 131));
655         pteMacroBox->setFocusPolicy(Qt::WheelFocus);
656         pteMacroBox->setTabChangesFocus(true);
657
658         gridLayout_8->addWidget(groupBox_2, 0, 0, 1, 1);
659
660         pteMacroComment = new QPlainTextEdit(macroTab);
661         pteMacroComment->setObjectName(QStringLiteral("pteMacroComment"));
662         pteMacroComment->setEnabled(true);
663         pteMacroComment->setGeometry(QRect(410, 10, 311, 31));
664         pteMacroComment->setFocusPolicy(Qt::WheelFocus);
665         pteMacroComment->setContextMenuPolicy(Qt::NoContextMenu);
666         pteMacroComment->setAcceptDrops(true);
667         pteMacroComment->setToolTipDuration(1);
668         pteMacroComment->setInputMethodHints(Qt::ImhNone);
669         pteMacroComment->setVerticalScrollBarPolicy(Qt::ScrollBarAlwaysOff);
670         pteMacroComment->setTabChangesFocus(true);
671         pteMacroComment->setUndoRedoEnabled(true);
672         pteMacroComment->setReadOnly(false);
673         pteMacroComment->setTabStopWidth(1);
674         pteMacroText = new QPlainTextEdit(macroTab);
675         pteMacroText->setObjectName(QStringLiteral("pteMacroText"));
676         pteMacroText->setGeometry(QRect(410, 50, 311, 66));
677         pteMacroText->setFocusPolicy(Qt::StrongFocus);
678         pteMacroText->setContextMenuPolicy(Qt::NoContextMenu);
679         pteMacroText->setAcceptDrops(false);
680         pteMacroText->setInputMethodHints(Qt::ImhNoAutoUppercase);
681         pteMacroText->setUndoRedoEnabled(false);
682         pteMacroText->setReadOnly(false);
683         pteMacroText->setTabStopWidth(1);
684         layoutWidget = new QWidget(macroTab);
685         layoutWidget->setObjectName(QStringLiteral("layoutWidget"));
686         layoutWidget->setGeometry(QRect(300, 10, 131, 74));
687         verticalLayout_2 = new QVBoxLayout(layoutWidget);
688         verticalLayout_2->setObjectName(QStringLiteral("verticalLayout_2"));
689         verticalLayout_2->setSizeConstraint(QLayout::SetNoConstraint);
690         verticalLayout_2->setContentsMargins(0, 0, 0, 0);
691         groupBox_3 = new QGroupBox(layoutWidget);
692         groupBox_3->setObjectName(QStringLiteral("groupBox_3"));
693
694         verticalLayout_2->addWidget(groupBox_3);
695
696         groupBox_4 = new QGroupBox(layoutWidget);
697         groupBox_4->setObjectName(QStringLiteral("groupBox_4"));
698
699         verticalLayout_2->addWidget(groupBox_4);
700
701         layoutWidget_2 = new QWidget(macroTab);
702         layoutWidget_2->setObjectName(QStringLiteral("layoutWidget_2"));
703         layoutWidget_2->setGeometry(QRect(300, 160, 407, 36));
704         gridLayout_10 = new QGridLayout(layoutWidget_2);
705         gridLayout_10->setObjectName(QStringLiteral("gridLayout_10"));
706         gridLayout_10->setContentsMargins(0, 0, 0, 0);
707         btnStopMacro = new QPushButton(layoutWidget_2);
708         btnStopMacro->setObjectName(QStringLiteral("btnStopMacro"));
709         btnStopMacro->setEnabled(false);
710
```

```
711         gridLayout_10->addWidget(btnStopMacro, 0, 2, 1, 1);
712 
713         btnClearMacro = new QPushButton(layoutWidget_2);
714         btnClearMacro->setObjectName(QStringLiteral("btnClearMacro"));
715 
716         gridLayout_10->addWidget(btnClearMacro, 0, 3, 1, 1);
717 
718         btnStartMacro = new QPushButton(layoutWidget_2);
719         btnStartMacro->setObjectName(QStringLiteral("btnStartMacro"));
720 
721         gridLayout_10->addWidget(btnStartMacro, 0, 1, 1, 1);
722 
723         label = new QLabel(layoutWidget_2);
724         label->setObjectName(QStringLiteral("label"));
725         label->setStyleSheet(QLatin1String("font: 75 10pt ; \n"
726 ""));
727 
728         gridLayout_10->addWidget(label, 0, 0, 1, 1);
729 
730         lbl_macro = new QLabel(macroTab);
731         lbl_macro->setObjectName(QStringLiteral("lbl_macro"));
732         lbl_macro->setGeometry(QRect(300, 130, 431, 27));
733         lbl_macro->setStyleSheet(QStringLiteral("font: 75 11pt ; color: darkblue;"));
734         layoutWidget2 = new QWidget(macroTab);
735         layoutWidget2->setObjectName(QStringLiteral("layoutWidget2"));
736         layoutWidget2->setGeometry(QRect(40, 170, 238, 28));
737         delayButtons = new QHBoxLayout(layoutWidget2);
738         delayButtons->setSpacing(0);
739         delayButtons->setObjectName(QStringLiteral("delayButtons"));
740         delayButtons->setSizeConstraint(QLayout::SetMinimumSize);
741         delayButtons->setContentsMargins(0, 0, 0, 0);
742         rb_delay_no = new QRadioButton(layoutWidget2);
743         rb_delay_no->setObjectName(QStringLiteral("rb_delay_no"));
744         rb_delay_no->setStyleSheet(QStringLiteral("font: 10pt ; color:  green;"));
745 
746         delayButtons->addWidget(rb_delay_no);
747 
748         rb_delay_default = new QRadioButton(layoutWidget2);
749         rb_delay_default->setObjectName(QStringLiteral("rb_delay_default"));
750         rb_delay_default->setStyleSheet(QStringLiteral("font: 10pt ; color:  green;"));
751         rb_delay_default->setChecked(false);
752         rb_delay_default->setAutoExclusive(true);
753 
754         delayButtons->addWidget(rb_delay_default);
755 
756         rb_delay_asTyped = new QRadioButton(layoutWidget2);
757         rb_delay_asTyped->setObjectName(QStringLiteral("rb_delay_asTyped"));
758         rb_delay_asTyped->setStyleSheet(QStringLiteral("font: 10pt ; color:  green;"));
759         rb_delay_asTyped->setChecked(true);
760 
761         delayButtons->addWidget(rb_delay_asTyped);
762 
763         txtBuffer = new QLabel(macroTab);
764         txtBuffer->setObjectName(QStringLiteral("txtBuffer"));
765         txtBuffer->setEnabled(false);
766         txtBuffer->setGeometry(QRect(300, 100, 0, 0));
767         QSizePolicy sizePolicy4(QSizePolicy::Fixed, QSizePolicy::Fixed);
768         sizePolicy4.setHorizontalStretch(0);
769         sizePolicy4.setVerticalStretch(0);
770         sizePolicy4.setHeightForWidth(txtBuffer->sizePolicy().hasHeightForWidth());
771         txtBuffer->setSizePolicy(sizePolicy4);
772         txtBuffer->setMaximumSize(QSize(0, 0));
773         txtBuffer->setContextMenuPolicy(Qt::NoContextMenu);
774 #ifndef QT_NO_TOOLTIP
775         txtBuffer->setToolTip(QStringLiteral(""));
776 #endif // QT_NO_TOOLTIP
777         txtBuffer->setToolTipDuration(0);
778 #ifndef QT_NO_STATUSTIP
779         txtBuffer->setStatusTip(QStringLiteral(""));
780 #endif // QT_NO_STATUSTIP
781 #ifndef QT_NO_WHATSTHIS
782         txtBuffer->setWhatsThis(QStringLiteral(""));
783 #endif // QT_NO_WHATSTHIS
784         txtBuffer->setFrameShadow(QFrame::Plain);
785         txtBuffer->setLineWidth(0);
786         txtBuffer->setTextFormat(Qt::PlainText);
787         txtBuffer->setIndent(0);
788         txtBuffer->setTextInteractionFlags(Qt::NoTextInteraction);
789         tabWidget->addTab(macroTab, QString());
790         pteMacroComment->raise();
791         layoutWidget->raise();
792         layoutWidget1->raise();
793         layoutWidget->raise();
794         pteMacroText->raise();
795         layoutWidget_2->raise();
796         lbl_macro->raise();
797         txtBuffer->raise();
```

```
798
799         verticalLayout->addWidget(tabWidget);
800
801         horizontalLayout = new QHBoxLayout();
802         horizontalLayout->setObjectName(QStringLiteral("horizontalLayout"));
803         horizontalSpacer_14 = new QSpacerItem(40, 20, QSizePolicy::Expanding,
    QSizePolicy::Minimum);
804
805         horizontalLayout->addItem(horizontalSpacer_14);
806
807         unbindButton = new QPushButton(RebindWidget);
808         unbindButton->setObjectName(QStringLiteral("unbindButton"));
809
810         horizontalLayout->addWidget(unbindButton);
811
812         resetButton = new QPushButton(RebindWidget);
813         resetButton->setObjectName(QStringLiteral("resetButton"));
814
815         horizontalLayout->addWidget(resetButton);
816
817         cancelButton = new QPushButton(RebindWidget);
818         cancelButton->setObjectName(QStringLiteral("cancelButton"));
819
820         horizontalLayout->addWidget(cancelButton);
821
822         applyButton = new QPushButton(RebindWidget);
823         applyButton->setObjectName(QStringLiteral("applyButton"));
824
825         horizontalLayout->addWidget(applyButton);
826
827
828         verticalLayout->addLayout(horizontalLayout);
829
830         QWidget::setTabOrder(tabWidget, typingButton);
831         QWidget::setTabOrder(typingButton, typingBox);
832         QWidget::setTabOrder(typingBox, modButton);
833         QWidget::setTabOrder(modButton, modBox);
834         QWidget::setTabOrder(modBox, fnButton);
835         QWidget::setTabOrder(fnButton, fnBox);
836         QWidget::setTabOrder(fnBox, numButton);
837         QWidget::setTabOrder(numButton, numBox);
838         QWidget::setTabOrder(numBox, mediaButton);
839         QWidget::setTabOrder(mediaButton, mediaBox);
840         QWidget::setTabOrder(mediaBox, mbButton);
841         QWidget::setTabOrder(mbButton, mbBox);
842         QWidget::setTabOrder(mbBox, mb2Button);
843         QWidget::setTabOrder(mb2Button, mb2Box);
844         QWidget::setTabOrder(mb2Box, wheelButton);
845         QWidget::setTabOrder(wheelButton, wheelBox);
846         QWidget::setTabOrder(wheelBox, dpiButton);
847         QWidget::setTabOrder(dpiButton, dpiBox);
848         QWidget::setTabOrder(dpiBox, dpiCustXBox);
849         QWidget::setTabOrder(dpiCustXBox, dpiCustYBox);
850         QWidget::setTabOrder(dpiCustYBox, animButton);
851         QWidget::setTabOrder(animButton, animBox);
852         QWidget::setTabOrder(animBox, animOnceBox);
853         QWidget::setTabOrder(animOnceBox, animKrBox);
854         QWidget::setTabOrder(animKrBox, modeButton);
855         QWidget::setTabOrder(modeButton, modeBox);
856         QWidget::setTabOrder(modeBox, modeWrapBox);
857         QWidget::setTabOrder(modeWrapBox, lightButton);
858         QWidget::setTabOrder(lightButton, lightBox);
859         QWidget::setTabOrder(lightBox, lightWrapBox);
860         QWidget::setTabOrder(lightWrapBox, lockButton);
861         QWidget::setTabOrder(lockButton, lockBox);
862         QWidget::setTabOrder(lockBox, programKpButton);
863         QWidget::setTabOrder(programKpButton, programKpBox);
864         QWidget::setTabOrder(programKpBox, programKpSIBox);
865         QWidget::setTabOrder(programKpSIBox, programKpModeBox);
866         QWidget::setTabOrder(programKpModeBox, programKrButton);
867         QWidget::setTabOrder(programKrButton, programKrBox);
868         QWidget::setTabOrder(programKrBox, programKrSIBox);
869         QWidget::setTabOrder(programKrSIBox, programKrModeBox);
870         QWidget::setTabOrder(programKrModeBox, pteMacroBox);
871         QWidget::setTabOrder(pteMacroBox, rb_delay_no);
872         QWidget::setTabOrder(rb_delay_no, rb_delay_default);
873         QWidget::setTabOrder(rb_delay_default, rb_delay_asTyped);
874         QWidget::setTabOrder(rb_delay_asTyped, pteMacroComment);
875         QWidget::setTabOrder(pteMacroComment, pteMacroText);
876         QWidget::setTabOrder(pteMacroText, btnStartMacro);
877         QWidget::setTabOrder(btnStartMacro, btnStopMacro);
878         QWidget::setTabOrder(btnStopMacro, btnClearMacro);
879         QWidget::setTabOrder(btnClearMacro, unbindButton);
880         QWidget::setTabOrder(unbindButton, resetButton);
881         QWidget::setTabOrder(resetButton, cancelButton);
882         QWidget::setTabOrder(cancelButton, applyButton);
883
```

```
884         retranslateUi(RebindWidget);
885         QObject::connect(btnClearMacro, SIGNAL(clicked()),
    pteMacroText, SLOT(clear()));
886         QObject::connect(btnClearMacro, SIGNAL(clicked()),
    pteMacroComment, SLOT(clear()));
887         QObject::connect(btnClearMacro, SIGNAL(clicked()),
    btnStopMacro, SLOT(animateClick()));
888         QObject::connect(cancelButton, SIGNAL(clicked()), txtBuffer, SLOT(clear()));
889         QObject::connect(btnClearMacro, SIGNAL(clicked()),
    txtBuffer, SLOT(clear()));
890         QObject::connect(btnStartMacro, SIGNAL(clicked()),
    txtBuffer, SLOT(clear()));
891         QObject::connect(unbindButton, SIGNAL(clicked()), txtBuffer, SLOT(clear()));
892         QObject::connect(resetButton, SIGNAL(clicked()), txtBuffer, SLOT(clear()));
893
894         tabWidget->setCurrentIndex(5);
895         applyButton->setDefault(true);
896
897
898         QMetaObject::connectSlotsByName(RebindWidget);
899     } // setupUi
```

Here is the call graph for this function:



Here is the caller graph for this function:

12.79.3 Field Documentation

12.79.3.1 QWidget* Ui_RebindWidget::actionTab [inherited]

Definition at line 92 of file ui_rebindwidget.h.

Referenced by Ui_RebindWidget::retranslateUi(), and Ui_RebindWidget::setupUi().

12.79.3.2 QComboBox* Ui_RebindWidget::animBox [inherited]

Definition at line 84 of file ui_rebindwidget.h.

Referenced by RebindWidget::applyChanges(), RebindWidget::on_animBox_currentIndexChanged(), RebindWidget::on_animButton_clicked(), Ui_RebindWidget::retranslateUi(), RebindWidget::setBox(), RebindWidget::setSelection(), and Ui_RebindWidget::setupUi().

12.79.3.3 QRadioButton* Ui_RebindWidget::animButton [inherited]

Definition at line 90 of file ui_rebindwidget.h.

Referenced by RebindWidget::on_animBox_currentIndexChanged(), Ui_RebindWidget::retranslateUi(), and Ui_RebindWidget::setupUi().

12.79.3.4 QCheckBox* Ui_RebindWidget::animKrBox [inherited]

Definition at line 87 of file ui_rebindwidget.h.

Referenced by RebindWidget::applyChanges(), Ui_RebindWidget::retranslateUi(), RebindWidget::setSelection(), and Ui_RebindWidget::setupUi().

12.79.3.5 QCheckBox* Ui_RebindWidget::animOnceBox [inherited]

Definition at line 88 of file ui_rebindwidget.h.

Referenced by RebindWidget::applyChanges(), Ui_RebindWidget::retranslateUi(), RebindWidget::setSelection(), and Ui_RebindWidget::setupUi().

12.79.3.6 QWidget* Ui_RebindWidget::animTab [inherited]

Definition at line 81 of file ui_rebindwidget.h.

Referenced by Ui_RebindWidget::retranslateUi(), and Ui_RebindWidget::setupUi().

12.79.3.7 QPushButton* Ui_RebindWidget::applyButton [inherited]

Definition at line 161 of file ui_rebindwidget.h.

Referenced by RebindWidget::on_btnStartMacro_clicked(), RebindWidget::on_btnStopMacro_clicked(), Ui_RebindWidget::retranslateUi(), and Ui_RebindWidget::setupUi().

12.79.3.8 QPushButton* Ui_RebindWidget::btnClearMacro [inherited]

Definition at line 146 of file ui_rebindwidget.h.

Referenced by Ui_RebindWidget::retranslateUi(), and Ui_RebindWidget::setupUi().

12.79.3.9 QPushButton* Ui_RebindWidget::btnStartMacro [inherited]

Definition at line 147 of file ui_rebindwidget.h.

Referenced by RebindWidget::on_btnStartMacro_clicked(), RebindWidget::on_btnStopMacro_clicked(), Ui_RebindWidget::retranslateUi(), and Ui_RebindWidget::setupUi().

12.79.3.10 QPushButton* Ui_RebindWidget::btnStopMacro [inherited]

Definition at line 145 of file ui_rebindwidget.h.

Referenced by RebindWidget::on_btnStartMacro_clicked(), RebindWidget::on_btnStopMacro_clicked(), Ui_RebindWidget::retranslateUi(), and Ui_RebindWidget::setupUi().

12.79.3.11 QPushButton* Ui_RebindWidget::cancelButton [inherited]

Definition at line 160 of file ui_rebindwidget.h.

Referenced by Ui_RebindWidget::retranslateUi(), and Ui_RebindWidget::setupUi().

12.79.3.12 QHBoxLayout* Ui_RebindWidget::delayButtons [inherited]

Definition at line 151 of file ui_rebindwidget.h.

Referenced by Ui_RebindWidget::setupUi().

12.79.3.13 QComboBox* Ui_RebindWidget::dpiBox [inherited]

Definition at line 68 of file ui_rebindwidget.h.

Referenced by RebindWidget::applyChanges(), RebindWidget::on_dpiBox_currentIndexChanged(), RebindWidget::on_dpiButton_clicked(), Ui_RebindWidget::retranslateUi(), RebindWidget::setBind(), RebindWidget::setBox(), RebindWidget::setSelection(), and Ui_RebindWidget::setupUi().

12.79.3.14 QRadioButton* Ui_RebindWidget::dpiButton [inherited]

Definition at line 62 of file ui_rebindwidget.h.

Referenced by RebindWidget::on_dpiBox_currentIndexChanged(), Ui_RebindWidget::retranslateUi(), RebindWidget::setBind(), and Ui_RebindWidget::setupUi().

12.79.3.15 QLabel* Ui_RebindWidget::dpiCustLabel [inherited]

Definition at line 78 of file ui_rebindwidget.h.

Referenced by RebindWidget::on_dpiBox_currentIndexChanged(), Ui_RebindWidget::retranslateUi(), RebindWidget::setBind(), and Ui_RebindWidget::setupUi().

12.79.3.16 QSpinBox* Ui_RebindWidget::dpiCustXBox [inherited]

Definition at line 73 of file ui_rebindwidget.h.

Referenced by RebindWidget::applyChanges(), RebindWidget::on_dpiBox_currentIndexChanged(), RebindWidget::setBind(), RebindWidget::setSelection(), and Ui_RebindWidget::setupUi().

12.79.3.17 QSpinBox* Ui_RebindWidget::dpiCustYBox [inherited]

Definition at line 69 of file ui_rebindwidget.h.

Referenced by RebindWidget::applyChanges(), RebindWidget::on_dpiBox_currentIndexChanged(), RebindWidget::setBind(), RebindWidget::setSelection(), and Ui_RebindWidget::setupUi().

12.79.3.18 QLabel* Ui_RebindWidget::dpiWarning [inherited]

Definition at line 74 of file ui_rebindwidget.h.

Referenced by Ui_RebindWidget::retranslateUi(), RebindWidget::setBind(), and Ui_RebindWidget::setupUi().

12.79.3.19 QComboBox* Ui_RebindWidget::fnBox [inherited]

Definition at line 44 of file ui_rebindwidget.h.

Referenced by RebindWidget::applyChanges(), RebindWidget::on_fnBox_currentIndexChanged(), RebindWidget::on_fnButton_clicked(), RebindWidget::RebindWidget(), Ui_RebindWidget::retranslateUi(), RebindWidget::setBox(), RebindWidget::setSelection(), and Ui_RebindWidget::setupUi().

12.79.3.20 QRadioButton* Ui_RebindWidget::fnButton [inherited]

Definition at line 56 of file ui_rebindwidget.h.

Referenced by RebindWidget::on_fnBox_currentIndexChanged(), Ui_RebindWidget::retranslateUi(), and Ui_-RebindWidget::setupUi().

12.79.3.21 QGridLayout* Ui_RebindWidget::gridLayout [inherited]

Definition at line 61 of file ui_rebindwidget.h.

Referenced by Ui_RebindWidget::setupUi().

12.79.3.22 QGridLayout* Ui_RebindWidget::gridLayout_10 [inherited]

Definition at line 144 of file ui_rebindwidget.h.

Referenced by Ui_RebindWidget::setupUi().

12.79.3.23 QGridLayout* Ui_RebindWidget::gridLayout_2 [inherited]

Definition at line 41 of file ui_rebindwidget.h.

Referenced by Ui_RebindWidget::setupUi().

12.79.3.24 QGridLayout* Ui_RebindWidget::gridLayout_3 [inherited]

Definition at line 93 of file ui_rebindwidget.h.

Referenced by Ui_RebindWidget::setupUi().

12.79.3.25 QGridLayout* Ui_RebindWidget::gridLayout_4 [inherited]

Definition at line 111 of file ui_rebindwidget.h.

Referenced by Ui_RebindWidget::setupUi().

12.79.3.26 QGridLayout* Ui_RebindWidget::gridLayout_5 [inherited]

Definition at line 82 of file ui_rebindwidget.h.

Referenced by Ui_RebindWidget::setupUi().

12.79.3.27 QGridLayout* Ui_RebindWidget::gridLayout_8 [inherited]

Definition at line 134 of file ui_rebindwidget.h.

Referenced by Ui_RebindWidget::setupUi().

12.79.3.28 QGroupBox* Ui_RebindWidget::groupBox_2 [inherited]

Definition at line 135 of file ui_rebindwidget.h.

Referenced by Ui_RebindWidget::retranslateUi(), and Ui_RebindWidget::setupUi().

12.79.3.29 QGroupBox* Ui_RebindWidget::groupBox_3 [inherited]

Definition at line 141 of file ui_rebindwidget.h.

Referenced by Ui_RebindWidget::retranslateUi(), and Ui_RebindWidget::setupUi().

12.79.3.30 QGroupBox* Ui_RebindWidget::groupBox_4 [inherited]

Definition at line 142 of file ui_rebindwidget.h.

Referenced by Ui_RebindWidget::retranslateUi(), and Ui_RebindWidget::setupUi().

12.79.3.31 QHBoxLayout* Ui_RebindWidget::horizontalLayout [inherited]

Definition at line 156 of file ui_rebindwidget.h.

Referenced by Ui_RebindWidget::setupUi().

12.79.3.32 QHBoxLayout* Ui_RebindWidget::horizontalLayout_2 [inherited]

Definition at line 122 of file ui_rebindwidget.h.

Referenced by Ui_RebindWidget::setupUi().

12.79.3.33 QHBoxLayout* Ui_RebindWidget::horizontalLayout_3 [inherited]

Definition at line 127 of file ui_rebindwidget.h.

Referenced by Ui_RebindWidget::setupUi().

12.79.3.34 QSpacerItem* Ui_RebindWidget::horizontalSpacer [inherited]

Definition at line 113 of file ui_rebindwidget.h.

Referenced by Ui_RebindWidget::setupUi().

12.79.3.35 QSpacerItem* Ui_RebindWidget::horizontalSpacer_10 [inherited]

Definition at line 107 of file ui_rebindwidget.h.

Referenced by Ui_RebindWidget::setupUi().

12.79.3.36 QSpacerItem* Ui_RebindWidget::horizontalSpacer_11 [inherited]

Definition at line 116 of file ui_rebindwidget.h.

Referenced by Ui_RebindWidget::setupUi().

12.79.3.37 QSpacerItem* Ui_RebindWidget::horizontalSpacer_12 [inherited]

Definition at line 120 of file ui_rebindwidget.h.

Referenced by Ui_RebindWidget::setupUi().

12.79.3.38 QSpacerItem* Ui_RebindWidget::horizontalSpacer_13 [inherited]

Definition at line 117 of file ui_rebindwidget.h.

Referenced by Ui_RebindWidget::setupUi().

12.79.3.39 QSpacerItem* Ui_RebindWidget::horizontalSpacer_14 [inherited]

Definition at line 157 of file ui_rebindwidget.h.

Referenced by Ui_RebindWidget::setupUi().

12.79.3.40 QSpacerItem* Ui_RebindWidget::horizontalSpacer_15 [inherited]

Definition at line 125 of file ui_rebindwidget.h.

Referenced by Ui_RebindWidget::setupUi().

12.79.3.41 QSpacerItem* Ui_RebindWidget::horizontalSpacer_16 [inherited]

Definition at line 130 of file ui_rebindwidget.h.

Referenced by Ui_RebindWidget::setupUi().

12.79.3.42 QSpacerItem* Ui_RebindWidget::horizontalSpacer_17 [inherited]

Definition at line 59 of file ui_rebindwidget.h.

Referenced by Ui_RebindWidget::setupUi().

12.79.3.43 QSpacerItem* Ui_RebindWidget::horizontalSpacer_18 [inherited]

Definition at line 65 of file ui_rebindwidget.h.

Referenced by Ui_RebindWidget::setupUi().

12.79.3.44 QSpacerItem* Ui_RebindWidget::horizontalSpacer_19 [inherited]

Definition at line 79 of file ui_rebindwidget.h.

Referenced by Ui_RebindWidget::setupUi().

12.79.3.45 QSpacerItem* Ui_RebindWidget::horizontalSpacer_2 [inherited]

Definition at line 47 of file ui_rebindwidget.h.

Referenced by Ui_RebindWidget::setupUi().

12.79.3.46 QSpacerItem* Ui_RebindWidget::horizontalSpacer_20 [inherited]

Definition at line 66 of file ui_rebindwidget.h.

Referenced by Ui_RebindWidget::setupUi().

12.79.3.47 QSpacerItem* Ui_RebindWidget::horizontalSpacer_21 [inherited]

Definition at line 80 of file ui_rebindwidget.h.

Referenced by Ui_RebindWidget::setupUi().

12.79.3.48 QSpacerItem* Ui_RebindWidget::horizontalSpacer_22 [inherited]

Definition at line 76 of file ui_rebindwidget.h.

Referenced by Ui_RebindWidget::setupUi().

12.79.3.49 QSpacerItem* Ui_RebindWidget::horizontalSpacer_23 [inherited]

Definition at line 91 of file ui_rebindwidget.h.

Referenced by Ui_RebindWidget::setupUi().

12.79.3.50 QSpacerItem* Ui_RebindWidget::horizontalSpacer_24 [inherited]

Definition at line 85 of file ui_rebindwidget.h.

Referenced by Ui_RebindWidget::setupUi().

12.79.3.51 QSpacerItem* Ui_RebindWidget::horizontalSpacer_25 [inherited]

Definition at line 86 of file ui_rebindwidget.h.

Referenced by Ui_RebindWidget::setupUi().

12.79.3.52 QSpacerItem* Ui_RebindWidget::horizontalSpacer_26 [inherited]

Definition at line 89 of file ui_rebindwidget.h.

Referenced by Ui_RebindWidget::setupUi().

12.79.3.53 QSpacerItem* Ui_RebindWidget::horizontalSpacer_3 [inherited]

Definition at line 50 of file ui_rebindwidget.h.

Referenced by Ui_RebindWidget::setupUi().

12.79.3.54 QSpacerItem* Ui_RebindWidget::horizontalSpacer_4 [inherited]

Definition at line 54 of file ui_rebindwidget.h.

Referenced by Ui_RebindWidget::setupUi().

12.79.3.55 QSpacerItem* Ui_RebindWidget::horizontalSpacer_5 [inherited]

Definition at line 94 of file ui_rebindwidget.h.

Referenced by Ui_RebindWidget::setupUi().

12.79.3.56 QSpacerItem* Ui_RebindWidget::horizontalSpacer_6 [inherited]

Definition at line 109 of file ui_rebindwidget.h.

Referenced by Ui_RebindWidget::setupUi().

12.79.3.57 QSpacerItem* Ui_RebindWidget::horizontalSpacer_7 [inherited]

Definition at line 108 of file ui_rebindwidget.h.

Referenced by Ui_RebindWidget::setupUi().

12.79.3.58 QSpacerItem* Ui_RebindWidget::horizontalSpacer_8 [inherited]

Definition at line 106 of file ui_rebindwidget.h.

Referenced by Ui_RebindWidget::setupUi().

12.79.3.59 QSpacerItem* Ui_RebindWidget::horizontalSpacer_9 [inherited]

Definition at line 53 of file ui_rebindwidget.h.

Referenced by Ui_RebindWidget::setupUi().

12.79.3.60 QWidget* Ui_RebindWidget::keyTab [inherited]

Definition at line 40 of file ui_rebindwidget.h.

Referenced by Ui_RebindWidget::retranslateUi(), and Ui_RebindWidget::setupUi().

12.79.3.61 QLabel* Ui_RebindWidget::label [inherited]

Definition at line 148 of file ui_rebindwidget.h.

Referenced by Ui_RebindWidget::retranslateUi(), and Ui_RebindWidget::setupUi().

12.79.3.62 QWidget* Ui_RebindWidget::layoutWidget [inherited]

Definition at line 139 of file ui_rebindwidget.h.

Referenced by Ui_RebindWidget::setupUi().

12.79.3.63 QWidget* Ui_RebindWidget::layoutWidget1 [inherited]

Definition at line 133 of file ui_rebindwidget.h.

Referenced by Ui_RebindWidget::setupUi().

12.79.3.64 QWidget* Ui_RebindWidget::layoutWidget2 [inherited]

Definition at line 150 of file ui_rebindwidget.h.

Referenced by Ui_RebindWidget::setupUi().

12.79.3.65 QWidget* Ui_RebindWidget::layoutWidget_2 [inherited]

Definition at line 143 of file ui_rebindwidget.h.

Referenced by Ui_RebindWidget::setupUi().

12.79.3.66 QLabel* Ui_RebindWidget::lbl_macro [inherited]

Definition at line 149 of file ui_rebindwidget.h.

Referenced by RebindWidget::helpStatus(), Ui_RebindWidget::retranslateUi(), and Ui_RebindWidget::setupUi().

12.79.3.67 QComboBox* Ui_RebindWidget::lightBox [inherited]

Definition at line 102 of file ui_rebindwidget.h.

Referenced by RebindWidget::applyChanges(), RebindWidget::on_lightBox_currentIndexChanged(), RebindWidget::on_lightButton_clicked(), Ui_RebindWidget::retranslateUi(), RebindWidget::setBox(), RebindWidget::setSelection(), and Ui_RebindWidget::setupUi().

12.79.3.68 QRadioButton* Ui_RebindWidget::lightButton [inherited]

Definition at line 104 of file ui_rebindwidget.h.

Referenced by RebindWidget::on_lightBox_currentIndexChanged(), Ui_RebindWidget::retranslateUi(), and Ui_RebindWidget::setupUi().

12.79.3.69 QCheckBox* Ui_RebindWidget::lightWrapBox [inherited]

Definition at line 97 of file ui_rebindwidget.h.

Referenced by RebindWidget::applyChanges(), RebindWidget::on_lightBox_currentIndexChanged(), RebindWidget::RebindWidget(), Ui_RebindWidget::retranslateUi(), RebindWidget::setSelection(), and Ui_RebindWidget::setupUi().

12.79.3.70 QComboBox* Ui_RebindWidget::lockBox [inherited]

Definition at line 103 of file ui_rebindwidget.h.

Referenced by RebindWidget::applyChanges(), RebindWidget::on_lockBox_currentIndexChanged(), RebindWidget::on_lockButton_clicked(), Ui_RebindWidget::retranslateUi(), RebindWidget::setBox(), RebindWidget::setSelection(), and Ui_RebindWidget::setupUi().

12.79.3.71 QRadioButton* Ui_RebindWidget::lockButton [inherited]

Definition at line 100 of file ui_rebindwidget.h.

Referenced by RebindWidget::on_lockBox_currentIndexChanged(), Ui_RebindWidget::retranslateUi(), and Ui_RebindWidget::setupUi().

12.79.3.72 QWidget* Ui_RebindWidget::macroTab [inherited]

Definition at line 132 of file ui_rebindwidget.h.

Referenced by Ui_RebindWidget::retranslateUi(), and Ui_RebindWidget::setupUi().

12.79.3.73 QComboBox* Ui_RebindWidget::mb2Box [inherited]

Definition at line 67 of file ui_rebindwidget.h.

Referenced by RebindWidget::applyChanges(), RebindWidget::on_mb2Box_currentIndexChanged(), RebindWidget::on_mb2Button_clicked(), Ui_RebindWidget::retranslateUi(), RebindWidget::setBox(), RebindWidget::setSelection(), and Ui_RebindWidget::setupUi().

12.79.3.74 QRadioButton* Ui_RebindWidget::mb2Button [inherited]

Definition at line 77 of file ui_rebindwidget.h.

Referenced by RebindWidget::on_mb2Box_currentIndexChanged(), Ui_RebindWidget::retranslateUi(), and Ui_RebindWidget::setupUi().

12.79.3.75 QComboBox* Ui_RebindWidget::mbBox [inherited]

Definition at line 70 of file ui_rebindwidget.h.

Referenced by RebindWidget::applyChanges(), RebindWidget::on_mbBox_currentIndexChanged(), RebindWidget::on_mbButton_clicked(), Ui_RebindWidget::retranslateUi(), RebindWidget::setBox(), RebindWidget::setSelection(), and Ui_RebindWidget::setupUi().

12.79.3.76 QRadioButton* Ui_RebindWidget::mbButton [inherited]

Definition at line 72 of file ui_rebindwidget.h.

Referenced by RebindWidget::on_mbBox_currentIndexChanged(), Ui_RebindWidget::retranslateUi(), and Ui_RebindWidget::setupUi().

12.79.3.77 QComboBox* Ui_RebindWidget::mediaBox [inherited]

Definition at line 57 of file ui_rebindwidget.h.

Referenced by RebindWidget::applyChanges(), RebindWidget::on_mediaBox_currentIndexChanged(), RebindWidget::on_mediaButton_clicked(), RebindWidget::RebindWidget(), Ui_RebindWidget::retranslateUi(), RebindWidget::setBox(), RebindWidget::setSelection(), and Ui_RebindWidget::setupUi().

12.79.3.78 QRadioButton* Ui_RebindWidget::mediaButton [inherited]

Definition at line 51 of file ui_rebindwidget.h.

Referenced by RebindWidget::on_mediaBox_currentIndexChanged(), Ui_RebindWidget::retranslateUi(), and Ui_RebindWidget::setupUi().

12.79.3.79 QComboBox* Ui_RebindWidget::modBox [inherited]

Definition at line 49 of file ui_rebindwidget.h.

Referenced by RebindWidget::applyChanges(), RebindWidget::on_modBox_currentIndexChanged(), RebindWidget::on_modButton_clicked(), RebindWidget::RebindWidget(), Ui_RebindWidget::retranslateUi(), RebindWidget::setBox(), RebindWidget::setSelection(), and Ui_RebindWidget::setupUi().

12.79.3.80 QRadioButton* Ui_RebindWidget::modButton [inherited]

Definition at line 55 of file ui_rebindwidget.h.

Referenced by RebindWidget::on_modBox_currentIndexChanged(), Ui_RebindWidget::retranslateUi(), and Ui_-RebindWidget::setupUi().

12.79.3.81 QComboBox* Ui_RebindWidget::modeBox [inherited]

Definition at line 105 of file ui_rebindwidget.h.

Referenced by RebindWidget::applyChanges(), RebindWidget::on_modeBox_currentIndexChanged(), Rebind-Widget::on_modeButton_clicked(), Ui_RebindWidget::retranslateUi(), RebindWidget::setBind(), RebindWidget::set-Box(), RebindWidget::setSelection(), and Ui_RebindWidget::setupUi().

12.79.3.82 QRadioButton* Ui_RebindWidget::modeButton [inherited]

Definition at line 95 of file ui_rebindwidget.h.

Referenced by RebindWidget::on_modeBox_currentIndexChanged(), Ui_RebindWidget::retranslateUi(), and Ui_-RebindWidget::setupUi().

12.79.3.83 QCheckBox* Ui_RebindWidget::modeWrapBox [inherited]

Definition at line 96 of file ui_rebindwidget.h.

Referenced by RebindWidget::applyChanges(), RebindWidget::on_modeBox_currentIndexChanged(), Rebind-Widget::RebindWidget(), Ui_RebindWidget::retranslateUi(), RebindWidget::setSelection(), and Ui_RebindWidget-::setupUi().

12.79.3.84 QWidget* Ui_RebindWidget::mouseTab [inherited]

Definition at line 60 of file ui_rebindwidget.h.

Referenced by Ui_RebindWidget::retranslateUi(), and Ui_RebindWidget::setupUi().

12.79.3.85 QComboBox* Ui_RebindWidget::numBox [inherited]

Definition at line 42 of file ui_rebindwidget.h.

Referenced by RebindWidget::applyChanges(), RebindWidget::on_numBox_currentIndexChanged(), Rebind-Widget::on_numButton_clicked(), RebindWidget::RebindWidget(), Ui_RebindWidget::retranslateUi(), Rebind-Widget::setBox(), RebindWidget::setSelection(), and Ui_RebindWidget::setupUi().

12.79.3.86 QRadioButton* Ui_RebindWidget::numButton [inherited]

Definition at line 58 of file ui_rebindwidget.h.

Referenced by RebindWidget::on_numBox_currentIndexChanged(), Ui_RebindWidget::retranslateUi(), and Ui_-RebindWidget::setupUi().

12.79.3.87 QLineEdit* Ui_RebindWidget::programKpBox [inherited]

Definition at line 115 of file ui_rebindwidget.h.

Referenced by RebindWidget::applyChanges(), RebindWidget::on_programKpBox_textChanged(), RebindWidget-::on_programKpButton_clicked(), RebindWidget::setBox(), RebindWidget::setSelection(), and Ui_RebindWidget-::setupUi().

12.79.3.88 QCheckBox* Ui_RebindWidget::programKpButton [inherited]

Definition at line 114 of file ui_rebindwidget.h.

Referenced by RebindWidget::on_programKpBox_textChanged(), Ui_RebindWidget::retranslateUi(), RebindWidget::setBox(), and Ui_RebindWidget::setupUi().

12.79.3.89 QWidget* Ui_RebindWidget::programKpExtra [inherited]

Definition at line 121 of file ui_rebindwidget.h.

Referenced by RebindWidget::on_programKpBox_textChanged(), RebindWidget::RebindWidget(), and Ui_RebindWidget::setupUi().

12.79.3.90 QComboBox* Ui_RebindWidget::programKpModeBox [inherited]

Definition at line 124 of file ui_rebindwidget.h.

Referenced by RebindWidget::applyChanges(), RebindWidget::on_programKpSIBox_clicked(), Ui_RebindWidget::retranslateUi(), RebindWidget::setSelection(), and Ui_RebindWidget::setupUi().

12.79.3.91 QCheckBox* Ui_RebindWidget::programKpSIBox [inherited]

Definition at line 123 of file ui_rebindwidget.h.

Referenced by RebindWidget::applyChanges(), Ui_RebindWidget::retranslateUi(), RebindWidget::setSelection(), and Ui_RebindWidget::setupUi().

12.79.3.92 QLineEdit* Ui_RebindWidget::programKrBox [inherited]

Definition at line 118 of file ui_rebindwidget.h.

Referenced by RebindWidget::applyChanges(), RebindWidget::on_programKrBox_textChanged(), RebindWidget::on_programKrButton_clicked(), RebindWidget::setBox(), RebindWidget::setSelection(), and Ui_RebindWidget::setupUi().

12.79.3.93 QCheckBox* Ui_RebindWidget::programKrButton [inherited]

Definition at line 119 of file ui_rebindwidget.h.

Referenced by RebindWidget::on_programKrBox_textChanged(), Ui_RebindWidget::retranslateUi(), RebindWidget::setBox(), and Ui_RebindWidget::setupUi().

12.79.3.94 QWidget* Ui_RebindWidget::programKrExtra [inherited]

Definition at line 126 of file ui_rebindwidget.h.

Referenced by RebindWidget::on_programKrBox_textChanged(), RebindWidget::RebindWidget(), and Ui_RebindWidget::setupUi().

12.79.3.95 QComboBox* Ui_RebindWidget::programKrModeBox [inherited]

Definition at line 129 of file ui_rebindwidget.h.

Referenced by RebindWidget::applyChanges(), RebindWidget::on_programKrSIBox_clicked(), Ui_RebindWidget::retranslateUi(), RebindWidget::setSelection(), and Ui_RebindWidget::setupUi().

12.79.3.96 QCheckBox* Ui_RebindWidget::programKrSlBox [inherited]

Definition at line 128 of file ui_rebindwidget.h.

Referenced by RebindWidget::applyChanges(), Ui_RebindWidget::retranslateUi(), RebindWidget::setSelection(), and Ui_RebindWidget::setupUi().

12.79.3.97 QWidget* Ui_RebindWidget::programTab [inherited]

Definition at line 110 of file ui_rebindwidget.h.

Referenced by Ui_RebindWidget::retranslateUi(), and Ui_RebindWidget::setupUi().

12.79.3.98 QLabel* Ui_RebindWidget::progTipLabel [inherited]

Definition at line 131 of file ui_rebindwidget.h.

Referenced by RebindWidget::RebindWidget(), Ui_RebindWidget::retranslateUi(), and Ui_RebindWidget::setupUi().

12.79.3.99 QPlainTextEdit* Ui_RebindWidget::pteMacroBox [inherited]

Definition at line 136 of file ui_rebindwidget.h.

Referenced by RebindWidget::applyChanges(), RebindWidget::convertMacroBox(), RebindWidget::on_btnStartMacro_clicked(), RebindWidget::on_pteMacroBox_textChanged(), RebindWidget::setBox(), RebindWidget::setSelection(), and Ui_RebindWidget::setupUi().

12.79.3.100 QPlainTextEdit* Ui_RebindWidget::pteMacroComment [inherited]

Definition at line 137 of file ui_rebindwidget.h.

Referenced by RebindWidget::applyChanges(), Ui_RebindWidget::retranslateUi(), RebindWidget::setSelection(), and Ui_RebindWidget::setupUi().

12.79.3.101 QPlainTextEdit* Ui_RebindWidget::pteMacroText [inherited]

Definition at line 138 of file ui_rebindwidget.h.

Referenced by RebindWidget::applyChanges(), RebindWidget::on_btnStartMacro_clicked(), Ui_RebindWidget::retranslateUi(), RebindWidget::setSelection(), and Ui_RebindWidget::setupUi().

12.79.3.102 QRadioButton* Ui_RebindWidget::rb_delay_asTyped [inherited]

Definition at line 154 of file ui_rebindwidget.h.

Referenced by RebindWidget::applyChanges(), RebindWidget::on_btnStartMacro_clicked(), RebindWidget::on_btnStopMacro_clicked(), Ui_RebindWidget::retranslateUi(), RebindWidget::setCorrectRadioButton(), and Ui_RebindWidget::setupUi().

12.79.3.103 QRadioButton* Ui_RebindWidget::rb_delay_default [inherited]

Definition at line 153 of file ui_rebindwidget.h.

Referenced by RebindWidget::convertMacroBox(), RebindWidget::on_btnStartMacro_clicked(), RebindWidget::on_btnStopMacro_clicked(), Ui_RebindWidget::retranslateUi(), RebindWidget::setCorrectRadioButton(), and Ui_RebindWidget::setupUi().

12.79.3.104 QRadioButton* Ui_RebindWidget::rb_delay_no [inherited]

Definition at line 152 of file ui_rebindwidget.h.

Referenced by RebindWidget::convertMacroBox(), RebindWidget::on_btnStartMacro_clicked(), RebindWidget::on_btnStopMacro_clicked(), Ui_RebindWidget::retranslateUi(), RebindWidget::setCorrectRadioButton(), and Ui_RebindWidget::setupUi().

12.79.3.105 QPushButton* Ui_RebindWidget::resetButton [inherited]

Definition at line 159 of file ui_rebindwidget.h.

Referenced by RebindWidget::on_btnStartMacro_clicked(), RebindWidget::on_btnStopMacro_clicked(), Ui_RebindWidget::retranslateUi(), and Ui_RebindWidget::setupUi().

12.79.3.106 QTabWidget* Ui_RebindWidget::tabWidget [inherited]

Definition at line 39 of file ui_rebindwidget.h.

Referenced by Ui_RebindWidget::retranslateUi(), RebindWidget::setSelection(), and Ui_RebindWidget::setupUi().

12.79.3.107 QLabel* Ui_RebindWidget::txtBuffer [inherited]

Definition at line 155 of file ui_rebindwidget.h.

Referenced by RebindWidget::applyChanges(), RebindWidget::convertMacroBox(), Ui_RebindWidget::retranslateUi(), RebindWidget::setBox(), RebindWidget::setSelection(), and Ui_RebindWidget::setupUi().

12.79.3.108 QComboBox* Ui_RebindWidget::typingBox [inherited]

Definition at line 43 of file ui_rebindwidget.h.

Referenced by RebindWidget::applyChanges(), RebindWidget::on_typingBox_currentIndexChanged(), RebindWidget::on_typingButton_clicked(), Ui_RebindWidget::retranslateUi(), RebindWidget::setBind(), RebindWidget::setBox(), RebindWidget::setSelection(), and Ui_RebindWidget::setupUi().

12.79.3.109 QRadioButton* Ui_RebindWidget::typingButton [inherited]

Definition at line 48 of file ui_rebindwidget.h.

Referenced by RebindWidget::on_typingBox_currentIndexChanged(), Ui_RebindWidget::retranslateUi(), and Ui_RebindWidget::setupUi().

12.79.3.110 QPushButton* Ui_RebindWidget::unbindButton [inherited]

Definition at line 158 of file ui_rebindwidget.h.

Referenced by RebindWidget::on_btnStartMacro_clicked(), RebindWidget::on_btnStopMacro_clicked(), Ui_RebindWidget::retranslateUi(), and Ui_RebindWidget::setupUi().

12.79.3.111 QVBoxLayout* Ui_RebindWidget::verticalLayout [inherited]

Definition at line 38 of file ui_rebindwidget.h.

Referenced by Ui_RebindWidget::setupUi().

12.79.3.112 QVBoxLayout* Ui_RebindWidget::verticalLayout_2 [inherited]

Definition at line 140 of file ui_rebindwidget.h.

Referenced by Ui_RebindWidget::setupUi().

12.79.3.113 QSpacerItem* Ui_RebindWidget::verticalSpacer [inherited]

Definition at line 112 of file ui_rebindwidget.h.

Referenced by Ui_RebindWidget::setupUi().

12.79.3.114 QSpacerItem* Ui_RebindWidget::verticalSpacer_10 [inherited]

Definition at line 45 of file ui_rebindwidget.h.

Referenced by Ui_RebindWidget::setupUi().

12.79.3.115 QSpacerItem* Ui_RebindWidget::verticalSpacer_11 [inherited]

Definition at line 83 of file ui_rebindwidget.h.

Referenced by Ui_RebindWidget::setupUi().

12.79.3.116 QSpacerItem* Ui_RebindWidget::verticalSpacer_2 [inherited]

Definition at line 63 of file ui_rebindwidget.h.

Referenced by Ui_RebindWidget::setupUi().

12.79.3.117 QSpacerItem* Ui_RebindWidget::verticalSpacer_3 [inherited]

Definition at line 46 of file ui_rebindwidget.h.

Referenced by Ui_RebindWidget::setupUi().

12.79.3.118 QSpacerItem* Ui_RebindWidget::verticalSpacer_4 [inherited]

Definition at line 52 of file ui_rebindwidget.h.

Referenced by Ui_RebindWidget::setupUi().

12.79.3.119 QSpacerItem* Ui_RebindWidget::verticalSpacer_5 [inherited]

Definition at line 101 of file ui_rebindwidget.h.

Referenced by Ui_RebindWidget::setupUi().

12.79.3.120 QSpacerItem* Ui_RebindWidget::verticalSpacer_6 [inherited]

Definition at line 99 of file ui_rebindwidget.h.

Referenced by Ui_RebindWidget::setupUi().

12.79.3.121 QSpacerItem* Ui_RebindWidget::verticalSpacer_7 [inherited]

Definition at line 98 of file ui_rebindwidget.h.

Referenced by Ui_RebindWidget::setupUi().

12.79.3.122 QSpacerItem* Ui_RebindWidget::verticalSpacer_8 [inherited]

Definition at line 64 of file ui_rebindwidget.h.

Referenced by Ui_RebindWidget::setupUi().

12.79.3.123 QComboBox* Ui_RebindWidget::wheelBox [inherited]

Definition at line 71 of file ui_rebindwidget.h.

Referenced by RebindWidget::applyChanges(), RebindWidget::on_wheelBox_currentIndexChanged(), RebindWidget::on_wheelButton_clicked(), Ui_RebindWidget::retranslateUi(), RebindWidget::setBox(), RebindWidget::setSelection(), and Ui_RebindWidget::setupUi().

12.79.3.124 QRadioButton* Ui_RebindWidget::wheelButton [inherited]

Definition at line 75 of file ui_rebindwidget.h.

Referenced by RebindWidget::on_wheelBox_currentIndexChanged(), Ui_RebindWidget::retranslateUi(), and Ui_RebindWidget::setupUi().

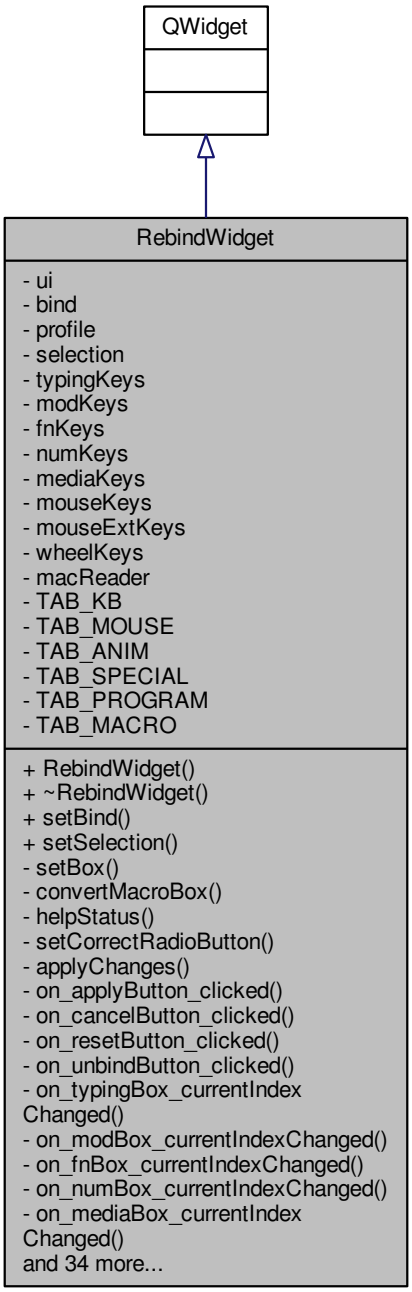
The documentation for this class was generated from the following file:

- src/ckb/ui_rebindwidget.h

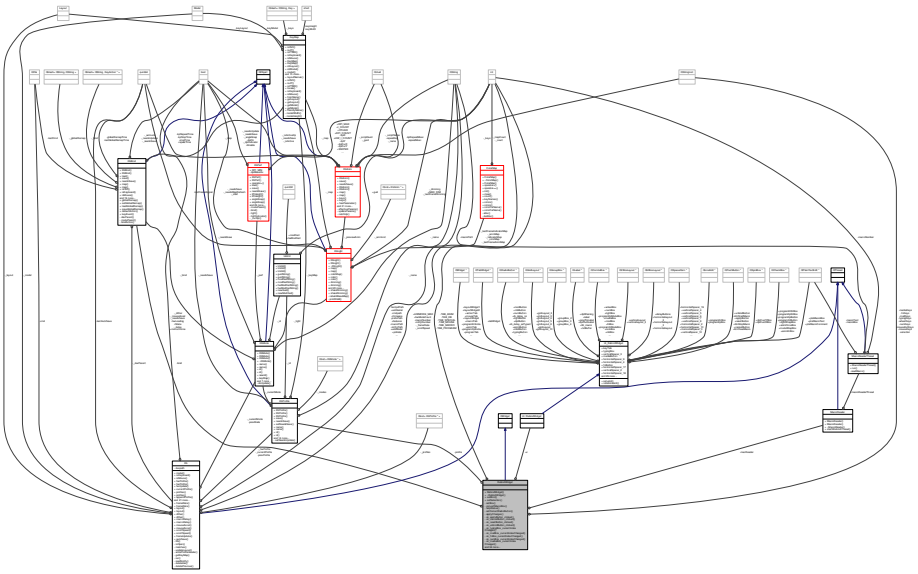
12.80 RebindWidget Class Reference

`#include <src/ckb/rebindwidget.h>`

Inheritance diagram for RebindWidget:



Collaboration diagram for RebindWidget:



Public Member Functions

- RebindWidget (QWidget *parent)
- ~RebindWidget ()
- void setBind (KbBind *newBind, KbProfile *newProfile)
- void setSelection (const QStringList &newSelection, bool applyPrevious=false)

Private Slots

- void applyChanges (const QStringList &keys, bool doUnbind)
- void on_applyButton_clicked ()
- void on_cancelButton_clicked ()
- void on_resetButton_clicked ()
- void on_unbindButton_clicked ()
- void on_typingBox_currentIndexChanged (int index)
- void on_modBox_currentIndexChanged (int index)
- void on_fnBox_currentIndexChanged (int index)
- void on_numBox_currentIndexChanged (int index)
- void on_mediaBox_currentIndexChanged (int index)
- void on_mbBox_currentIndexChanged (int index)
- void on_mb2Box_currentIndexChanged (int index)
- void on_wheelBox_currentIndexChanged (int index)
- void on_dpiBox_currentIndexChanged (int index)
- void on_modeBox_currentIndexChanged (int index)
- void on_lightBox_currentIndexChanged (int index)
- void on_lockBox_currentIndexChanged (int index)
- void on_programKpBox_textChanged (const QString &arg1)
- void on_programKrBox_textChanged (const QString &arg1)
- void on_animBox_currentIndexChanged (int index)
- void on_typingButton_clicked (bool checked)
- void on_modButton_clicked (bool checked)
- void on_fnButton_clicked (bool checked)

- void on_numButton_clicked (bool checked)
- void on_mediaButton_clicked (bool checked)
- void on_mbButton_clicked (bool checked)
- void on_mb2Button_clicked (bool checked)
- void on_wheelButton_clicked (bool checked)
- void on_dpiButton_clicked (bool checked)
- void on_modeButton_clicked (bool checked)
- void on_lightButton_clicked (bool checked)
- void on_lockButton_clicked (bool checked)
- void on_programKpButton_clicked (bool checked)
- void on_programKrButton_clicked (bool checked)
- void on_programKpSIBox_clicked (bool checked)
- void on_programKrSIBox_clicked (bool checked)
- void on_animButton_clicked (bool checked)
- void on_pteMacroBox_textChanged ()
- void on_btnStartMacro_clicked ()

  RebindWidget::on_btnStartMacro_clicked starts macro recording. A new notification channel and MacroReader are created to do the job.
- void on_btnStopMacro_clicked ()

  RebindWidget::on_btnStopMacro_clicked ends the macro recording. Notify channel ist closed, the ReaderThread is deleted when the notification is really down.
- void on_btnClearMacro_clicked ()

  RebindWidget::on_btnClearMacro_clicked changes the help info an the panel. The job of clearing the input panels is triggerd with signal/slot via the RebindWidget.ui file.
- void on_rb_delay_no_toggled (bool checked)

  RebindWidget::on_rb_delay_no_toggled.
- void on_rb_delay_asTyped_toggled (bool checked)

  RebindWidget::on_rb_delay_asTyped_toggled.
- void on_rb_delay_default_toggled (bool checked)

  RebindWidget::on_rb_delay_default_toggled.

Private Member Functions

- void setBox (QWidget *box)
- void convertMacroBox ()

  RebindWidget::convertMacroBox converts the macroBox content. The KB sends each keypress as "key [+|-]<keyname><newline>" This is followed by timing information (delays between keystrokes).
- void helpStatus (int status)

  RebindWidget::helpStatus shows a help line in the ui.
- void setCorrectRadioButton (QString macdef)

  RebindWidget::setCorrectRadioButton.

Private Attributes

- Ui::RebindWidget * ui
- KbBind * bind
- KbProfile * profile
- QStringList selection
- QStringList typingKeys
- QStringList modKeys
- QStringList fnKeys
- QStringList numKeys
- QStringList mediaKeys

- QStringList mouseKeys
- QStringList mouseExtKeys
- QStringList wheelKeys
- MacroReader * macReader

  macReader holds the MacroReader when macro recording starts.

Static Private Attributes

- static const int TAB_KB = 0
- static const int TAB_MOUSE = 1
- static const int TAB_ANIM = 2
- static const int TAB_SPECIAL = 3
- static const int TAB_PROGRAM = 4
- static const int TAB_MACRO = 5

12.80.1 Detailed Description

Definition at line 15 of file rebindwidget.h.

12.80.2 Constructor & Destructor Documentation

12.80.2.1 RebindWidget::RebindWidget (QWidget * *parent*) [explicit]

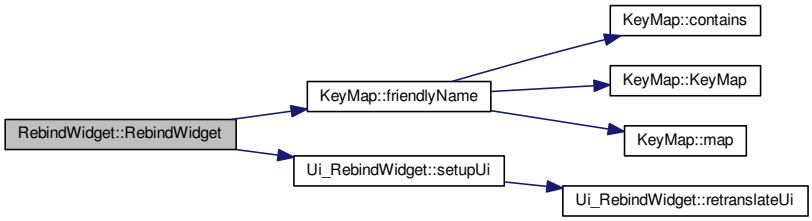
Definition at line 9 of file rebindwidget.cpp.

References Ui_RebindWidget::fnBox, fnKeys, KeyMap::friendlyName(), Ui_RebindWidget::lightWrapBox, Ui_RebindWidget::mediaBox, mediaKeys, Ui_RebindWidget::modBox, Ui_RebindWidget::modeWrapBox, modKeys, mouseExtKeys, mouseKeys, Ui_RebindWidget::numBox, numKeys, Ui_RebindWidget::programKpExtra, Ui_RebindWidget::programKrExtra, Ui_RebindWidget::progTipLabel, Ui_RebindWidget::setupUi(), ui, and wheelKeys.

```
9                                           :
10     QWidget(parent),
11     ui(new Ui::RebindWidget),
12     bind(0), profile(0), macReader(0)
13 {
14     ui->setupUi(this);
15     ui->lightWrapBox->hide();
16     ui->modeWrapBox->hide();
17     ui->programKpExtra->hide();
18     ui->programKrExtra->hide();
19
20     // Populate key lists
21     modKeys << "caps" << "lshift" << "rshift" << "lctrl" << "rctrl" << "lwin" << "rwin" << "lalt" <<
       "ralt" << "rmenu" << "fn";
22     fnKeys << "esc" << "f1" << "f2" << "f3" << "f4" << "f5" << "f6" << "f7" << "f8" << "f9" << "f10"
      << "f11" << "f12" << "prtscn" << "scroll" << "pause" << "f13" << "f14" << "f15" << "f16" << "f17" << "f18" <
      < "f19" << "f20" << "ins" << "del" << "home" << "end" << "pgup" << "pgdn" << "power" << "lightup" << "
      lightdn" << "up" << "down" << "left" << "right";
23     numKeys << "numlock" << "num0" << "num1" << "num2" << "num3" << "num4" << "num5" << "num6" << "
      num7" << "num8" << "num9" << "numslash" << "numstar" << "numminus" << "numplus" << "numdot" << "numenter";
24     mediaKeys << "stop" << "prev" << "play" << "next" << "volup" << "voldn" << "mute" << "eject";
25     mouseKeys << "mouse1" << "mouse2" << "mouse3";
26     mouseExtKeys << "mouse4" << "mouse5" << "mouse6" << "mouse7" << "mouse8";
27     wheelKeys << "wheelup" << "wheeldn";
28     // Populate boxes
29     foreach(const QString& key, modKeys)
30         ui->modBox->addItem(KeyMap::friendlyName(key));
31     foreach(const QString& key, fnKeys)
32         ui->fnBox->addItem(KeyMap::friendlyName(key).replace("\n", " / "));
33     foreach(const QString& key, numKeys)
34         ui->numBox->addItem(KeyMap::friendlyName(key).remove("NumPad "));
35     foreach(const QString& key, mediaKeys)
36         ui->mediaBox->addItem(KeyMap::friendlyName(key));
37 #ifdef Q_OS_MACX
38     // Replace some OSX keys with their actual meanings
39     ui->modBox->setItemText(modKeys.indexOf("lwin") + 1, "Left Cmd");
40     ui->modBox->setItemText(modKeys.indexOf("rwin") + 1, "Right Cmd");
41     ui->modBox->setItemText(modKeys.indexOf("lalt") + 1, "Left Option");
```

```
42     ui->modBox->setItemText(modKeys.indexOf("ralt") + 1, "Right Option");
43     ui->fnBox->setItemText(fnKeys.indexOf("prtscn") + 1, "F13");
44     ui->fnBox->setItemText(fnKeys.indexOf("scroll") + 1, "F14");
45     ui->fnBox->setItemText(fnKeys.indexOf("pause") + 1, "F15");
46     ui->fnBox->setItemText(fnKeys.indexOf("f13") + 1, "F13 (Windows/Linux)");
47     ui->fnBox->setItemText(fnKeys.indexOf("f14") + 1, "F14 (Windows/Linux)");
48     ui->fnBox->setItemText(fnKeys.indexOf("f15") + 1, "F15 (Windows/Linux)");
49     ui->fnBox->setItemText(fnKeys.indexOf("ins") + 1, "Help");
50     ui->numBox->setItemText(numKeys.indexOf("numlock") + 1, "Clear");
51
52     // Add tip label
53     ui->progTipLabel->setText("<p style=\"line-height:150%\">Tip: use the <font face=\"
       monospace\">open</font> command to launch a file, directory, or app. For instance, to start Safari:<br /><font
        face=\"monospace\">  open /Applications/Safari.app</font></p>");
54 #else
55     ui->progTipLabel->setText("<p style=\"line-height:150%\">Tip: use <font face=\"monospace
       \">xdg-open</font> to launch a file or directory. For instance, to open your home folder:<br /><font face=\"
       monospace\">  xdg-open " + QStandardPaths::writableLocation(QStandardPaths::HomeLocation) + "
       </font></p>");
56 #endif
57 }
```

Here is the call graph for this function:



12.80.2.2 RebindWidget::~RebindWidget ( )

Definition at line 59 of file rebindwidget.cpp.

References ui.

```
59                                {
60     delete ui;
61 }
```

12.80.3 Member Function Documentation

12.80.3.1 void RebindWidget::applyChanges (const QStringList & *keys*, bool *doUnbind*) [private], [slot]

G-key macro handling: Set the macro definiton for all keys selected (indeed, it may be multiple keys). First, concat the Macro Key Definion and the Macro plain text after escaping possible colos in the parts for Macro Text and Macro Comment.

But first, there is a special condition to handle: You have recorded a macro with timing infos. Afterwards you changed manually the timing infos in the pteMacroBox and press Apply. In that case we must overwrite the txtBuffer to remember your changes.

Todo There is still a bug in the state machine: If you record a macro in asTyped-mode, switch to another mode and change the vontent of the pteMacroBox manually, then the changes are not saved in the timing buffer. But anyhow, let's do more relevant things...

Definition at line 334 of file rebindwidget.cpp.

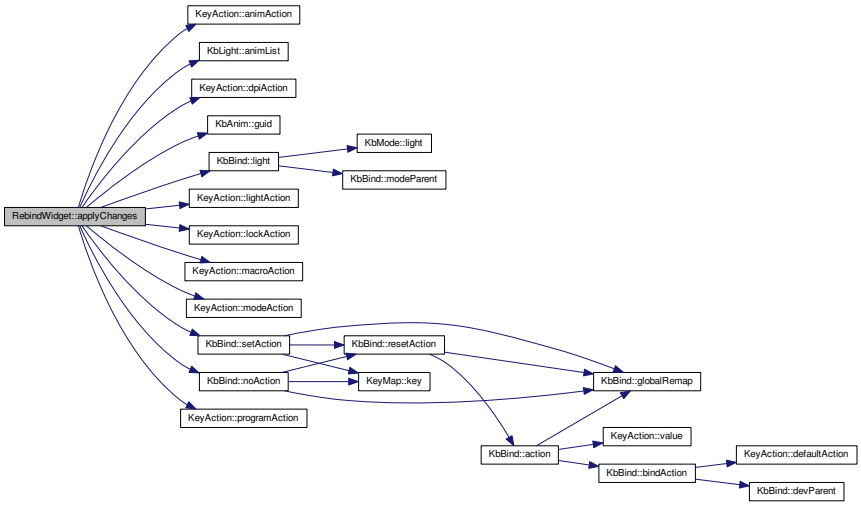
References KeyAction::animAction(), Ui_RebindWidget::animBox, Ui_RebindWidget::animKrBox, KbLight::animList(), Ui_RebindWidget::animOnceBox, bind, DPI_OFFSET, KeyAction::dpiAction(), Ui_RebindWidget::dpiBox, Ui_RebindWidget::dpiCustXBox, Ui_RebindWidget::dpiCustYBox, Ui_RebindWidget::fnBox, fnKeys, KbAnim::guid(), KbBind::light(), KeyAction::lightAction(), Ui_RebindWidget::lightBox, Ui_RebindWidget::lightWrapBox, KeyAction::lockAction(), Ui_RebindWidget::lockBox, KeyAction::macroAction(), Ui_RebindWidget::mb2Box, Ui_RebindWidget::mbBox, Ui_RebindWidget::mediaBox, mediaKeys, Ui_RebindWidget::modBox, KeyAction::modeAction(), Ui_RebindWidget::modeBox, Ui_RebindWidget::modeWrapBox, modKeys, mouseExtKeys, mouseKeys, KbBind::noAction(), Ui_RebindWidget::numBox, numKeys, KeyAction::PROGRAM_PR_MULTI, KeyAction::PROGRAM_RE_INDEF, KeyAction::PROGRAM_RE_KPSTOP, KeyAction::PROGRAM_RE_MULTI, KeyAction::programAction(), Ui_RebindWidget::programKpBox, Ui_RebindWidget::programKpModeBox, Ui_RebindWidget::programKpSIBox, Ui_RebindWidget::programKrBox, Ui_RebindWidget::programKrModeBox, Ui_RebindWidget::programKrSIBox, Ui_RebindWidget::pteMacroBox, Ui_RebindWidget::pteMacroComment, Ui_RebindWidget::pteMacroText, Ui_RebindWidget::rb_delay_asTyped, KbBind::setAction(), Ui_RebindWidget::txtBuffer, Ui_RebindWidget::typingBox, typingKeys, ui, Ui_RebindWidget::wheelBox, and wheelKeys.

Referenced by on_applyButton_clicked(), and setSelection().

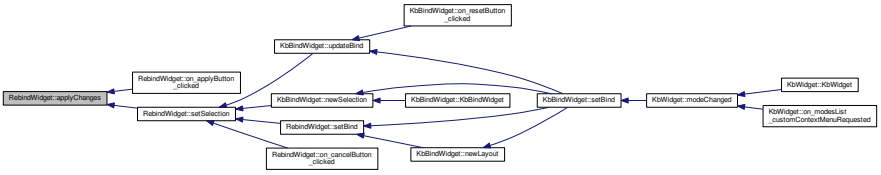
```
334                                                                         {
335     if(ui->typingBox->currentIndex() > 0)
336         bind->setAction(keys, typingKeys[ui->typingBox->currentIndex() -
    1]);
337     else if(ui->modBox->currentIndex() > 0)
338         bind->setAction(keys, modKeys[ui->modBox->currentIndex() - 1]);
339     else if(ui->fnBox->currentIndex() > 0)
340         bind->setAction(keys, fnKeys[ui->fnBox->currentIndex() - 1]);
341     else if(ui->numBox->currentIndex() > 0)
342         bind->setAction(keys, numKeys[ui->numBox->currentIndex() - 1]);
343     else if(ui->mediaBox->currentIndex() > 0)
344         bind->setAction(keys, mediaKeys[ui->mediaBox->currentIndex() - 1]);
345     else if(ui->mbBox->currentIndex() > 0)
346         bind->setAction(keys, mouseKeys[ui->mbBox->currentIndex() - 1]);
347     else if(ui->mb2Box->currentIndex() > 0)
348         bind->setAction(keys, mouseExtKeys[ui->mb2Box->currentIndex() - 1]
    );
349     else if(ui->wheelBox->currentIndex() > 0)
350         bind->setAction(keys, wheelKeys[ui->wheelBox->currentIndex() - 1]);
351     else if(ui->dpiBox->currentIndex() > 0)
352         bind->setAction(keys, KeyAction::dpiAction(
    ui->dpiBox->currentIndex() - DPI_OFFSET, ui->dpiCustXBox->value(),
    ui->dpiCustYBox->value()));
353     else if(ui->modeBox->currentIndex() > 0)
354         bind->setAction(keys, KeyAction::modeAction(
    ui->modeBox->currentIndex() - 3 - (ui->modeWrapBox->isChecked() &&
    ui->modeBox->currentIndex() < 3 ? 2 : 0)));
355     else if(ui->lightBox->currentIndex() > 0)
356         bind->setAction(keys, KeyAction::lightAction(
    ui->lightBox->currentIndex() - 1 + (ui->lightWrapBox->isChecked() ? 2 : 0)));
357     else if(ui->lockBox->currentIndex() > 0)
358         bind->setAction(keys, KeyAction::lockAction(
    ui->lockBox->currentIndex() - 1));
359     else if(ui->animBox->currentIndex() > 0){
360         KbAnim* anim = bind->light()->animList().at(ui->
    animBox->currentIndex() - 1);
361         bind->setAction(keys, KeyAction::animAction(anim->
    guid(), !ui->animOnceBox->isChecked(), ui->animKrBox->isChecked()));
362     } else if(!ui->programKpBox->text().isEmpty() || !ui->
    programKrBox->text().isEmpty()){
363         int kpStop = 0, krStop = 0;
364         if(!ui->programKpBox->text().isEmpty()){
365             if(!ui->programKpSIBox->isChecked())
366                 kpStop = KeyAction::PROGRAM_PR_MULTI;
367             else
368                 kpStop = ui->programKpModeBox->currentIndex();
369         }
370         if(!ui->programKrBox->text().isEmpty()){
371             if(!ui->programKrSIBox->isChecked())
372                 krStop = KeyAction::PROGRAM_RE_MULTI;
373             else if(ui->programKrModeBox->currentIndex() == 0)
374                 krStop = KeyAction::PROGRAM_RE_INDEF;
375             else
376                 krStop = KeyAction::PROGRAM_RE_KPSTOP;
377         }
378         bind->setAction(keys, KeyAction::programAction(
    ui->programKpBox->text(), ui->programKrBox->text(), kpStop | krStop));
379     } else if (ui->pteMacroBox->toPlainText().length() > 0) {
384
389         if (ui->rb_delay_asTyped->isChecked()) ui->txtBuffer->setText(
    ui->pteMacroBox->toPlainText());
390
396         QString mac;
```

```
397         mac = ui->txtBuffer->text();
398         mac = ui->pteMacroComment->toPlainText().replace(":", "&das_IST_31N_col0n;") + ":"
      + mac;
399         mac = ui->pteMacroText->toPlainText().replace(":", "&das_IST_31N_col0n;") + ":" + mac
    ;
400         mac = ui->pteMacroBox->toPlainText() + ":" + mac;
401         bind->setAction(keys, KeyAction::macroAction(mac));
402     } else if(doUnbind)
403         bind->noAction(keys);
404 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.80.3.2 void RebindWidget::convertMacroBox ( ) [private]

The ckb-daemon needs a shorter format, only " [+|-]<keyname>=<delay>", multiple entries are separated by comma.

That function does the conversion.

Definition at line 808 of file rebindwidget.cpp.

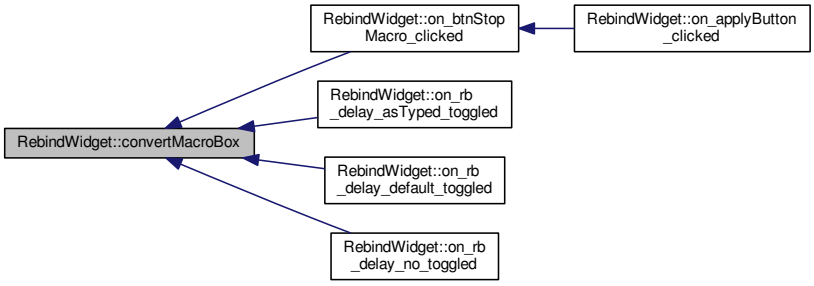
References Ui_RebindWidget::pteMacroBox, Ui_RebindWidget::rb_delay_default, Ui_RebindWidget::rb_delay_no, Ui_RebindWidget::txtBuffer, and ui.

Referenced by on_btnStopMacro_clicked(), on_rb_delay_asTyped_toggled(), on_rb_delay_default_toggled(), and on_rb_delay_no_toggled().

```
808                                 {
809     QString in;
810
```

```
811     // Remember the original input stream before it is converted.
812     // In case of new choice of delay mode we have to restore it.
813     if (ui->txtBuffer->text() == "") {
814         ui->txtBuffer->setText(ui->pteMacroBox->toPlainText());
815         in = ui->pteMacroBox->toPlainText();
816     } else in = ui->txtBuffer->text();
817
818     in.replace (QRegExp("\n"), ",");    // first join all in one line
819     in.replace (QRegExp("key "), "");   // then delete keyword "key" followed by space
820     in.replace (QRegExp(",="), "=");    // at last join each keystroke with its delay parameter
821
822     // How to deal with the delay params?
823     // Because the three radio buttons are mututally exclusive,
824     // we can run through the if-chain w/o conflicts.
825     // If rb_delay_asTyped is checked, do nothing, because that's the standard.
826
827     if (ui->rb_delay_default->isChecked()) {
828         in.replace(QRegExp("=\\d+,"), ",");  // Delete the timing infos, use default value
829         in.replace(QRegExp("=\\d+$"), "");   // The last entry is without comma
830     }
831     if (ui->rb_delay_no->isChecked()) {
832         in.replace(QRegExp("=\\d+,"), "=0,");  // Set timing infos to zero for no delay
833         in.replace(QRegExp("=\\d+$"), "=0");   // Again the last entry w/o comma
834         in.replace(QRegExp("([\\+\\-]\\w+),"), "\\1=0,");  // If no delay is given, force it to zero
835         in.replace(QRegExp("([\\+\\-]\\w+)$"), "\\1=0");
836     }
837
838     // Show the new format by replacing the older one.
839     ui->pteMacroBox->setPlainText(in);
840 }
```

Here is the caller graph for this function:



12.80.3.3 void RebindWidget::helpStatus (int *status*) [private]

Parameters

status	determines what to display.

Definition at line 782 of file rebindwidget.cpp.

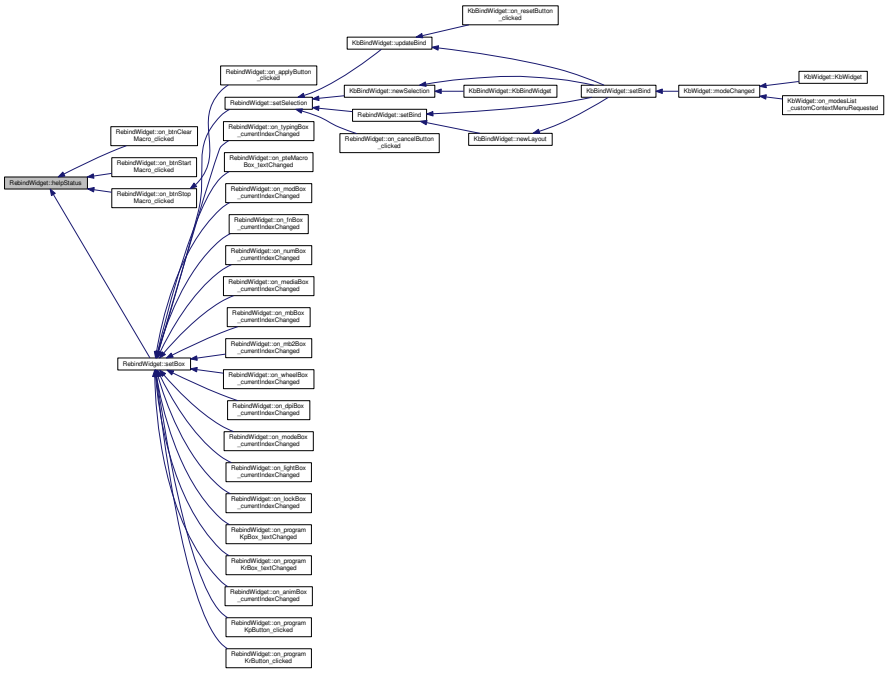
References Ui_RebindWidget::lbl_macro, and ui.

Referenced by on_btnClearMacro_clicked(), on_btnStartMacro_clicked(), on_btnStopMacro_clicked(), and setBox().

```
782                                     {
783     switch (status) {
784     case 1:
785         ui->lbl_macro->setText("Type in a macro name in the comment box and click start.");
786         break;
787     case 2:
788         ui->lbl_macro->setText("Type your macro and click stop when finished.");
789         break;
790     case 3:
791         ui->lbl_macro->setText("Click Apply or change values in Macro Key Actions in advance.");
792         break;
```

```
793     default:
794         ui->lbl_macro->setText(QString("Oops: Some magic in RebindWidget::helpStatus (%1)").arg(
    status));
795     }
796 }
```

Here is the caller graph for this function:



12.80.3.4 void RebindWidget::on_animBox_currentIndexChanged (int *index*) [private], [slot]

Definition at line 613 of file rebindwidget.cpp.

References Ui_RebindWidget::animBox, Ui_RebindWidget::animButton, setBox(), and ui.

```
613                                                     {
614     if(index == 0)
615         ui->animButton->setChecked(false);
616     else {
617         ui->animButton->setChecked(true);
618         setBox(ui->animBox);
619     }
620 }
```

Here is the call graph for this function:

12.80.3.5 void RebindWidget::on_animButton_clicked (bool *checked*) [private], [slot]

Definition at line 710 of file rebindwidget.cpp.

References Ui_RebindWidget::animBox, and ui.

```
710                                                 {
711     if(checked && ui->animBox->currentIndex() == 0)
712         ui->animBox->setCurrentIndex(1);
713 }
```

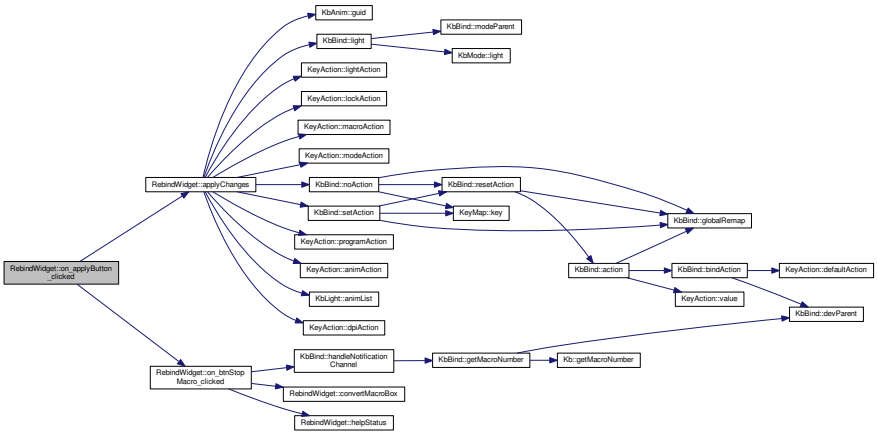
12.80.3.6 void RebindWidget::on_applyButton_clicked ( ) [private], [slot]

Definition at line 406 of file rebindwidget.cpp.

References applyChanges(), on_btnStopMacro_clicked(), and selection.

```
406                                         {
407     // Normally, this should be done via signalling.
408     // Because there is no serarate thread, we have to call it directly
409     // (otherwise we could do Key char conversion step by step,
410     // but so it is more easy to change the key definition):
411     on_btnStopMacro_clicked();
412     applyChanges(selection, true);
413 }
```

Here is the call graph for this function:



12.80.3.7 void RebindWidget::on_btnClearMacro_clicked ( ) [private], [slot]

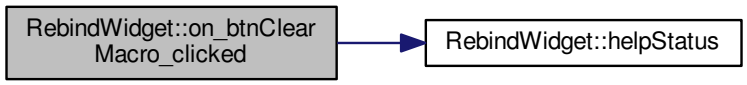
Todo I do not know what is the better solution with the delay-buttons in case of clicking clear: Reset the button to the default value or do not touch it? Not clear is ignored.

Definition at line 774 of file rebindwidget.cpp.

References helpStatus().

```
774                                             {
775     helpStatus(1);
776 }
```

Here is the call graph for this function:



12.80.3.8 void RebindWidget::on_btnStartMacro_clicked ( ) [private],[slot]

The UI is protected against false clicking (e.g. if you type start and than Apply, the channel is closed in wrong order).

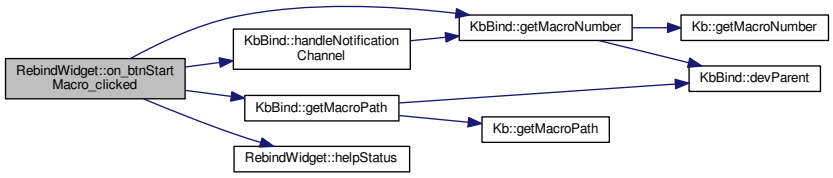
At this time, all neccessary params like macroNumber, macroPath, cmdFile etc. had been cached.

Definition at line 724 of file rebindwidget.cpp.

References Ui_RebindWidget::applyButton, bind, Ui_RebindWidget::btnStartMacro, Ui_RebindWidget::btnStopMacro, KbBind::getMacroNumber(), KbBind::getMacroPath(), KbBind::handleNotificationChannel(), helpStatus(), macReader, Ui_RebindWidget::pteMacroBox, Ui_RebindWidget::pteMacroText, Ui_RebindWidget::rb_delay_asTyped, Ui_RebindWidget::rb_delay_default, Ui_RebindWidget::rb_delay_no, Ui_RebindWidget::resetButton, ui, and Ui_RebindWidget::unbindButton.

```
724                                             {
725     if (!macReader) {
726         bind->handleNotificationChannel(true);
727         macReader = new MacroReader(bind->getMacroNumber(),
      bind->getMacroPath(), ui->pteMacroBox, ui->
      pteMacroText);
728         // because of the second thread we need to disable three of the four bottom buttons.
729         // Clicking "Stop" will enable them again.
730         ui->applyButton->setEnabled(false);
731         ui->resetButton->setEnabled(false);
732         ui->unbindButton->setEnabled(false);
733         ui->btnStartMacro->setEnabled(false);
734         ui->btnStopMacro->setEnabled(true);
735         ui->rb_delay_asTyped->setEnabled(false);
736         ui->rb_delay_no->setEnabled(false);
737         ui->rb_delay_default->setEnabled(false);
738         helpStatus(2);
739     }
740 }
```

Here is the call graph for this function:



12.80.3.9 void RebindWidget::on_btnStopMacro_clicked ( ) [private],[slot]

Afterwards, the characters in the MacroBox are changed from KB-out format to cmd-in format. At last the UI changes to the new state.

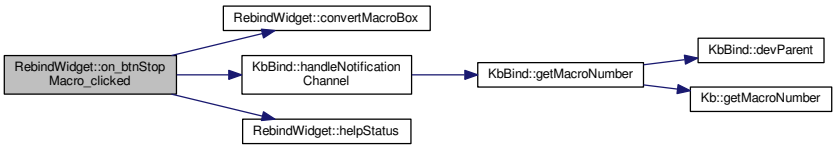
Definition at line 749 of file rebindwidget.cpp.

References Ui_RebindWidget::applyButton, bind, Ui_RebindWidget::btnStartMacro, Ui_RebindWidget::btnStopMacro, convertMacroBox(), KbBind::handleNotificationChannel(), helpStatus(), macReader, Ui_RebindWidget::rb_delay_asTyped, Ui_RebindWidget::rb_delay_default, Ui_RebindWidget::rb_delay_no, Ui_RebindWidget::resetButton, ui, and Ui_RebindWidget::unbindButton.

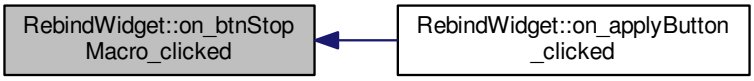
Referenced by on_applyButton_clicked().

```
749                                         {
750     if (macReader) {
751         bind->handleNotificationChannel(false);
752         delete macReader;
753         macReader = 0;
754         convertMacroBox();
755         ui->applyButton->setEnabled(true);
756         ui->resetButton->setEnabled(true);
757         ui->unbindButton->setEnabled(true);
758         ui->btnStartMacro->setEnabled(true);
759         ui->btnStopMacro->setEnabled(false);
760         ui->rb_delay_asTyped->setEnabled(true);
761         ui->rb_delay_no->setEnabled(true);
762         ui->rb_delay_default->setEnabled(true);
763         helpStatus(3);
764     }
765 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



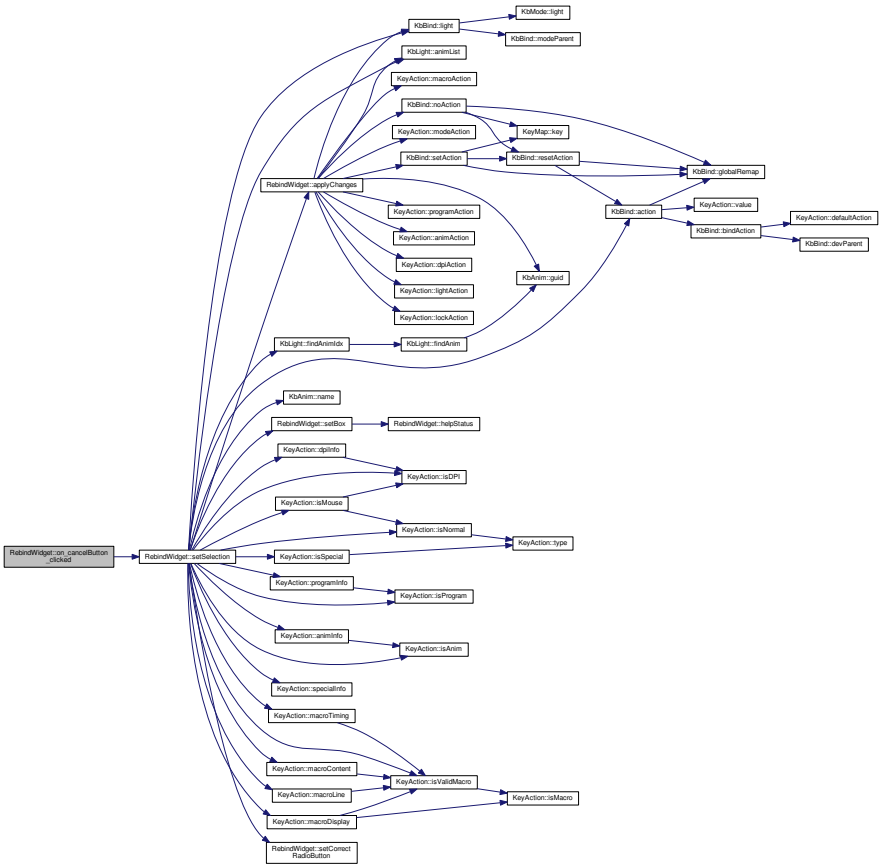
12.80.3.10 void RebindWidget::on_cancelButton_clicked ( ) [private], [slot]

Definition at line 415 of file rebindwidget.cpp.

References selection, and setSelection().

```
415                                         {
416     // Re-load selection
417     setSelection(selection);
418 }
```

Here is the call graph for this function:



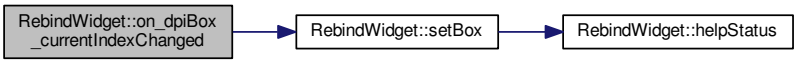
12.80.3.11 void RebindWidget::on_dpiBox_currentIndexChanged (int *index*) [private], [slot]

Definition at line 539 of file rebindwidget.cpp.

References DPI_CUST_IDX, Ui_RebindWidget::dpiBox, Ui_RebindWidget::dpiButton, Ui_RebindWidget::dpiCustLabel, Ui_RebindWidget::dpiCustXBox, Ui_RebindWidget::dpiCustYBox, setBox(), and ui.

```
539                                                     {
540     if(index != DPI_CUST_IDX){
541         // Clear custom DPI
542         ui->dpiCustXBox->setEnabled(false);
543         ui->dpiCustYBox->setEnabled(false);
544         ui->dpiCustLabel->setEnabled(false);
545     } else {
546         ui->dpiCustXBox->setEnabled(true);
547         ui->dpiCustYBox->setEnabled(true);
548         ui->dpiCustLabel->setEnabled(true);
549     }
550     if(index == 0)
551         ui->dpiButton->setChecked(false);
552     else {
553         ui->dpiButton->setChecked(true);
554         setBox(ui->dpiBox);
555     }
556 }
```

Here is the call graph for this function:



12.80.3.12 void RebindWidget::on_dpiButton_clicked (bool *checked*) [private], [slot]

Definition at line 695 of file rebindwidget.cpp.

References Ui_RebindWidget::dpiBox, and ui.

```
695                                                 {
696     if(checked && ui->dpiBox->currentIndex() == 0)
697         ui->dpiBox->setCurrentIndex(1);
698 }
```

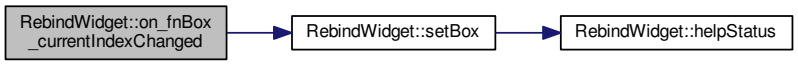
12.80.3.13 void RebindWidget::on_fnBox_currentIndexChanged (int *index*) [private], [slot]

Definition at line 485 of file rebindwidget.cpp.

References Ui_RebindWidget::fnBox, Ui_RebindWidget::fnButton, setBox(), and ui.

```
485                                                     {
486     if(index == 0)
487         ui->fnButton->setChecked(false);
488     else {
489         ui->fnButton->setChecked(true);
490         setBox(ui->fnBox);
491     }
492 }
```

Here is the call graph for this function:



12.80.3.14 void RebindWidget::on_fnButton_clicked (bool *checked*) [private], [slot]

Definition at line 632 of file rebindwidget.cpp.

References Ui_RebindWidget::fnBox, and ui.

```
632                                                 {
633     if(checked && ui->fnBox->currentIndex() == 0)
634         ui->fnBox->setCurrentIndex(1);
635 }
```

12.80.3.15 void RebindWidget::on_lightBox_currentIndexChanged (int *index*) [private], [slot]

Definition at line 571 of file rebindwidget.cpp.

References Ui_RebindWidget::lightBox, Ui_RebindWidget::lightButton, Ui_RebindWidget::lightWrapBox, setBox(), and ui.

```
571                                                     {
572     if(index == 0){
573         ui->lightButton->setChecked(false);
574         ui->lightWrapBox->hide();
575     } else {
576         ui->lightWrapBox->show();
577         ui->lightButton->setChecked(true);
578         setBox(ui->lightBox);
579     }
580 }
```

Here is the call graph for this function:

12.80.3.16 void RebindWidget::on_lightButton_clicked (bool *checked*) [private], [slot]

Definition at line 652 of file rebindwidget.cpp.

References Ui_RebindWidget::lightBox, and ui.

```
652                                                 {
653     if(checked && ui->lightBox->currentIndex() == 0)
654         ui->lightBox->setCurrentIndex(1);
655 }
```

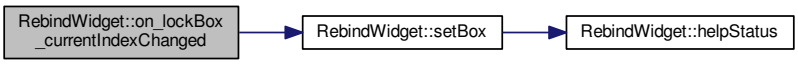
12.80.3.17 void RebindWidget::on_lockBox_currentIndexChanged (int *index*) [private], [slot]

Definition at line 582 of file rebindwidget.cpp.

References Ui_RebindWidget::lockBox, Ui_RebindWidget::lockButton, setBox(), and ui.

```
582                                                     {
583     if(index == 0)
584         ui->lockButton->setChecked(false);
585     else {
586         ui->lockButton->setChecked(true);
587         setBox(ui->lockBox);
588     }
589 }
```

Here is the call graph for this function:



12.80.3.18 void RebindWidget::on_lockButton_clicked (bool *checked*) [private], [slot]

Definition at line 657 of file rebindwidget.cpp.

References Ui_RebindWidget::lockBox, and ui.

```
657                                                 {
658     if(checked && ui->lockBox->currentIndex() == 0)
659         ui->lockBox->setCurrentIndex(1);
660 }
```

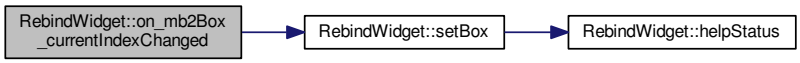
12.80.3.19 void RebindWidget::on_mb2Box_currentIndexChanged (int *index*) [private], [slot]

Definition at line 521 of file rebindwidget.cpp.

References Ui_RebindWidget::mb2Box, Ui_RebindWidget::mb2Button, setBox(), and ui.

```
521                                                     {
522     if(index == 0)
523         ui->mb2Button->setChecked(false);
524     else {
525         ui->mb2Button->setChecked(true);
526         setBox(ui->mb2Box);
527     }
528 }
```

Here is the call graph for this function:



12.80.3.20 void RebindWidget::on_mb2Button_clicked (bool *checked*) [private], [slot]

Definition at line 685 of file rebindwidget.cpp.

References Ui_RebindWidget::mb2Box, and ui.

```
685                                                 {
686     if(checked && ui->mb2Box->currentIndex() == 0)
687         ui->mb2Box->setCurrentIndex(1);
688 }
```

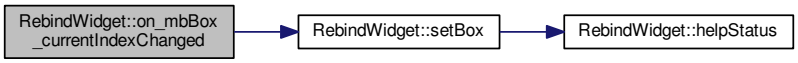
12.80.3.21 void RebindWidget::on_mbBox_currentIndexChanged (int *index*) [private], [slot]

Definition at line 512 of file rebindwidget.cpp.

References Ui_RebindWidget::mbBox, Ui_RebindWidget::mbButton, setBox(), and ui.

```
512                                                     {
513     if(index == 0)
514         ui->mbButton->setChecked(false);
515     else {
516         ui->mbButton->setChecked(true);
517         setBox(ui->mbBox);
518     }
519 }
```

Here is the call graph for this function:



12.80.3.22 void RebindWidget::on_mbButton_clicked (bool *checked*) [private], [slot]

Definition at line 680 of file rebindwidget.cpp.

References Ui_RebindWidget::mbBox, and ui.

```
680                                                 {
681     if(checked && ui->mbBox->currentIndex() == 0)
682         ui->mbBox->setCurrentIndex(1);
683 }
```

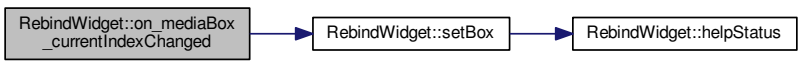
12.80.3.23 void RebindWidget::on_mediaBox_currentIndexChanged (int *index*) [private], [slot]

Definition at line 503 of file rebindwidget.cpp.

References Ui_RebindWidget::mediaBox, Ui_RebindWidget::mediaButton, setBox(), and ui.

```
503                                                     {
504     if(index == 0)
505         ui->mediaButton->setChecked(false);
506     else {
507         ui->mediaButton->setChecked(true);
508         setBox(ui->mediaBox);
509     }
510 }
```

Here is the call graph for this function:



12.80.3.24 void RebindWidget::on_mediaButton_clicked (bool *checked*) [private], [slot]

Definition at line 642 of file rebindwidget.cpp.

References Ui_RebindWidget::mediaBox, and ui.

```
642                                                 {
643     if(checked && ui->mediaBox->currentIndex() == 0)
644         ui->mediaBox->setCurrentIndex(1);
645 }
```

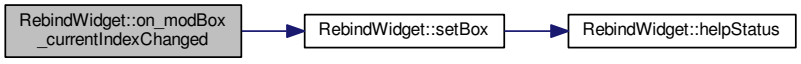
12.80.3.25 void RebindWidget::on_modBox_currentIndexChanged (int *index*) [private], [slot]

Definition at line 476 of file rebindwidget.cpp.

References Ui_RebindWidget::modBox, Ui_RebindWidget::modButton, setBox(), and ui.

```
476                                                     {
477     if(index == 0)
478         ui->modButton->setChecked(false);
479     else {
480         ui->modButton->setChecked(true);
481         setBox(ui->modBox);
482     }
483 }
```

Here is the call graph for this function:



12.80.3.26 void RebindWidget::on_modButton_clicked (bool *checked*) [private], [slot]

Definition at line 627 of file rebindwidget.cpp.

References Ui_RebindWidget::modBox, and ui.

```
627                                                 {
628     if(checked && ui->modBox->currentIndex() == 0)
629         ui->modBox->setCurrentIndex(1);
630 }
```

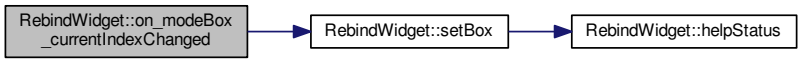
12.80.3.27 void RebindWidget::on_modeBox_currentIndexChanged (int *index*) [private], [slot]

Definition at line 558 of file rebindwidget.cpp.

References Ui_RebindWidget::modeBox, Ui_RebindWidget::modeButton, Ui_RebindWidget::modeWrapBox, setBox(), and ui.

```
558                                                     {
559     if(index == 1 || index == 2)
560         ui->modeWrapBox->show();
561     else
562         ui->modeWrapBox->hide();
563     if(index == 0)
564         ui->modeButton->setChecked(false);
565     else {
566         ui->modeButton->setChecked(true);
567         setBox(ui->modeBox);
568     }
569 }
```

Here is the call graph for this function:



12.80.3.28 void RebindWidget::on_modeButton_clicked (bool *checked*) [private], [slot]

Definition at line 647 of file rebindwidget.cpp.

References Ui_RebindWidget::modeBox, and ui.

```
647                                                 {
648     if(checked && ui->modeBox->currentIndex() == 0)
649         ui->modeBox->setCurrentIndex(1);
650 }
```

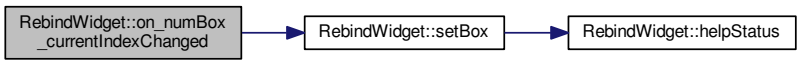
12.80.3.29 void RebindWidget::on_numBox_currentIndexChanged (int *index*) [private], [slot]

Definition at line 494 of file rebindwidget.cpp.

References Ui_RebindWidget::numBox, Ui_RebindWidget::numButton, setBox(), and ui.

```
494                                                     {
495     if(index == 0)
496         ui->numButton->setChecked(false);
497     else {
498         ui->numButton->setChecked(true);
499         setBox(ui->numBox);
500     }
501 }
```

Here is the call graph for this function:



12.80.3.30 void RebindWidget::on_numButton_clicked (bool *checked*) [private], [slot]

Definition at line 637 of file rebindwidget.cpp.

References Ui_RebindWidget::numBox, and ui.

```
637                                                 {
638     if(checked && ui->numBox->currentIndex() == 0)
639         ui->numBox->setCurrentIndex(1);
640 }
```

12.80.3.31 void RebindWidget::on_programKpBox_textChanged (const QString & *arg1*) [private], [slot]

Definition at line 591 of file rebindwidget.cpp.

References Ui_RebindWidget::programKpBox, Ui_RebindWidget::programKpButton, Ui_RebindWidget::programKpExtra, setBox(), and ui.

```
591                                                         {
592     if(arg1.isEmpty()){
593         ui->programKpButton->setChecked(false);
594         ui->programKpExtra->hide();
595     } else {
596         ui->programKpButton->setChecked(true);
597         setBox(ui->programKpBox);
598         ui->programKpExtra->show();
599     }
600 }
```

Here is the call graph for this function:

12.80.3.32 void RebindWidget::on_programKpButton_clicked (bool *checked*) [private], [slot]

Definition at line 662 of file rebindwidget.cpp.

References Ui_RebindWidget::programKpBox, setBox(), and ui.

```
662                                                     {
663     if(!checked){
664         ui->programKpBox->setText("");
665     } else if(ui->programKpBox->text().isEmpty()){
666         ui->programKpBox->setFocus();
667         setBox(ui->programKpBox);
668     }
669 }
```

Here is the call graph for this function:

12.80.3.33 void RebindWidget::on_programKpSlBox_clicked (bool *checked*) [private], [slot]

Definition at line 700 of file rebindwidget.cpp.

References Ui_RebindWidget::programKpModeBox, and ui.

```
700                                                     {
701     ui->programKpModeBox->setCurrentIndex(0);
702     ui->programKpModeBox->setEnabled(checked);
703 }
```

12.80.3.34 void RebindWidget::on_programKrBox_textChanged (const QString & *arg1*) [private], [slot]

Definition at line 602 of file rebindwidget.cpp.

References Ui_RebindWidget::programKrBox, Ui_RebindWidget::programKrButton, Ui_RebindWidget::programKrExtra, setBox(), and ui.

```
602                                                     {
603     if(arg1.isEmpty()){
604         ui->programKrButton->setChecked(false);
605         ui->programKrExtra->hide();
606     } else {
607         ui->programKrButton->setChecked(true);
608         setBox(ui->programKrBox);
609         ui->programKrExtra->show();
610     }
611 }
```

Here is the call graph for this function:

12.80.3.35 void RebindWidget::on_programKrButton_clicked (bool *checked*) [private], [slot]

Definition at line 671 of file rebindwidget.cpp.

References Ui_RebindWidget::programKrBox, setBox(), and ui.

```
671                                                     {
672     if(!checked){
673         ui->programKrBox->setText("");
674     } else if(ui->programKrBox->text().isEmpty()){
675         ui->programKrBox->setFocus();
676         setBox(ui->programKrBox);
677     }
678 }
```

Here is the call graph for this function:

12.80.3.36 void RebindWidget::on_programKrSlBox_clicked (bool *checked*) [private], [slot]

Definition at line 705 of file rebindwidget.cpp.

References Ui_RebindWidget::programKrModeBox, and ui.

```
705                                                     {
706     ui->programKrModeBox->setCurrentIndex(0);
707     ui->programKrModeBox->setEnabled(checked);
708 }
```

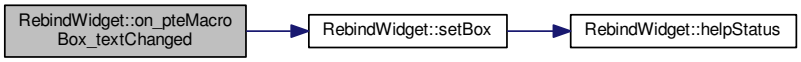
12.80.3.37 void RebindWidget::on_pteMacroBox_textChanged ( ) [private], [slot]

Definition at line 470 of file rebindwidget.cpp.

References Ui_RebindWidget::pteMacroBox, setBox(), and ui.

```
470                                             {
471     if (ui->pteMacroBox->toPlainText().length() > 0) {
472         setBox(ui->pteMacroBox);
473     }
474 }
```

Here is the call graph for this function:



12.80.3.38 void RebindWidget::on_rb_delay_asTyped_toggled (bool *checked*) [private], [slot]

Parameters

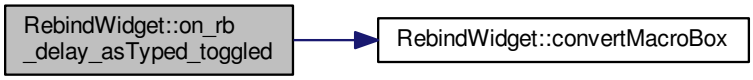
checked	This button ist clicked to use the delay times, as they are recorded. Returs a warning message, if we are not in the recording phase, because then we don't have the delay times any more.

Definition at line 862 of file rebindwidget.cpp.

References convertMacroBox().

```
863 {
864     convertMacroBox();
865 }
```

Here is the call graph for this function:



12.80.3.39 void RebindWidget::on_rb_delay_default_toggled (bool *checked*) [private], [slot]

Parameters

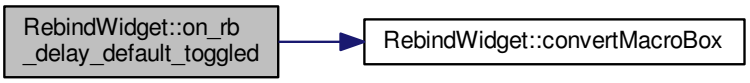
checked	This is as easy as the no-delay-button, because this means take the default values.

Definition at line 873 of file rebindwidget.cpp.

References convertMacroBox().

```
874 {
875     convertMacroBox();
876 }
```

Here is the call graph for this function:



12.80.3.40 void RebindWidget::on_rb_delay_no_toggled (bool *checked*) [private],[slot]

Parameters

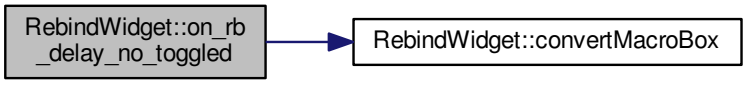
checked	The following slots are triggerd by changing the mutual exclusive radio buttons when choosing the delay. They are called, if the button ist enabled. This first one should disable all delay.

Definition at line 850 of file rebindwidget.cpp.

References convertMacroBox().

```
851 {
852     convertMacroBox();
853 }
```

Here is the call graph for this function:



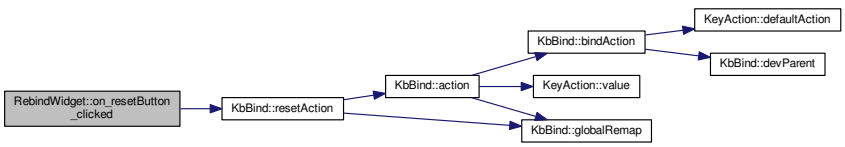
12.80.3.41 void RebindWidget::on_resetButton_clicked ( ) [private],[slot]

Definition at line 420 of file rebindwidget.cpp.

References bind, KbBind::resetAction(), and selection.

```
420                                     {
421     bind->resetAction(selection);
422 }
```

Here is the call graph for this function:



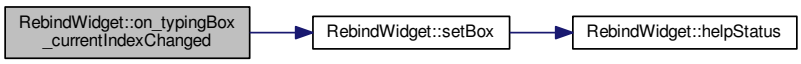
12.80.3.42 void RebindWidget::on_typingBox_currentIndexChanged (int *index*) [private], [slot]

Definition at line 461 of file rebindwidget.cpp.

References setBox(), Ui_RebindWidget::typingBox, Ui_RebindWidget::typingButton, and ui.

```
461                                                          {
462     if(index == 0)
463         ui->typingButton->setChecked(false);
464     else {
465         ui->typingButton->setChecked(true);
466         setBox(ui->typingBox);
467     }
468 }
```

Here is the call graph for this function:



12.80.3.43 void RebindWidget::on_typingButton_clicked (bool *checked*) [private], [slot]

Definition at line 622 of file rebindwidget.cpp.

References Ui_RebindWidget::typingBox, and ui.

```
622                                                    {
623     if(checked && ui->typingBox->currentIndex() == 0)
624         ui->typingBox->setCurrentIndex(1);
625 }
```

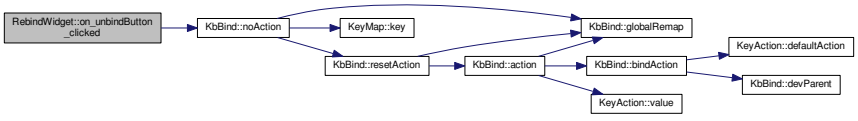
12.80.3.44 void RebindWidget::on_unbindButton_clicked ( ) [private], [slot]

Definition at line 424 of file rebindwidget.cpp.

References bind, KbBind::noAction(), and selection.

```
424                                             {
425     bind->noAction(selection);
426 }
```

Here is the call graph for this function:



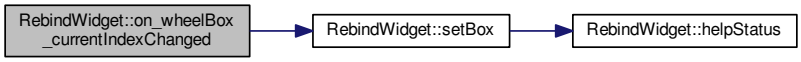
12.80.3.45 void RebindWidget::on_wheelBox_currentIndexChanged (int *index*) [private], [slot]

Definition at line 530 of file rebindwidget.cpp.

References setBox(), ui, Ui_RebindWidget::wheelBox, and Ui_RebindWidget::wheelButton.

```
530                                                         {
531     if(index == 0)
532         ui->wheelButton->setChecked(false);
533     else {
534         ui->wheelButton->setChecked(true);
535         setBox(ui->wheelBox);
536     }
537 }
```

Here is the call graph for this function:



12.80.3.46 void RebindWidget::on_wheelButton_clicked (bool *checked*) [private], [slot]

Definition at line 690 of file rebindwidget.cpp.

References ui, and Ui_RebindWidget::wheelBox.

```
690                                                     {
691     if(checked && ui->wheelBox->currentIndex() == 0)
692         ui->wheelBox->setCurrentIndex(1);
693 }
```

12.80.3.47 void RebindWidget::setBind (KbBind * *newBind*, KbProfile * *newProfile*)

Definition at line 63 of file rebindwidget.cpp.

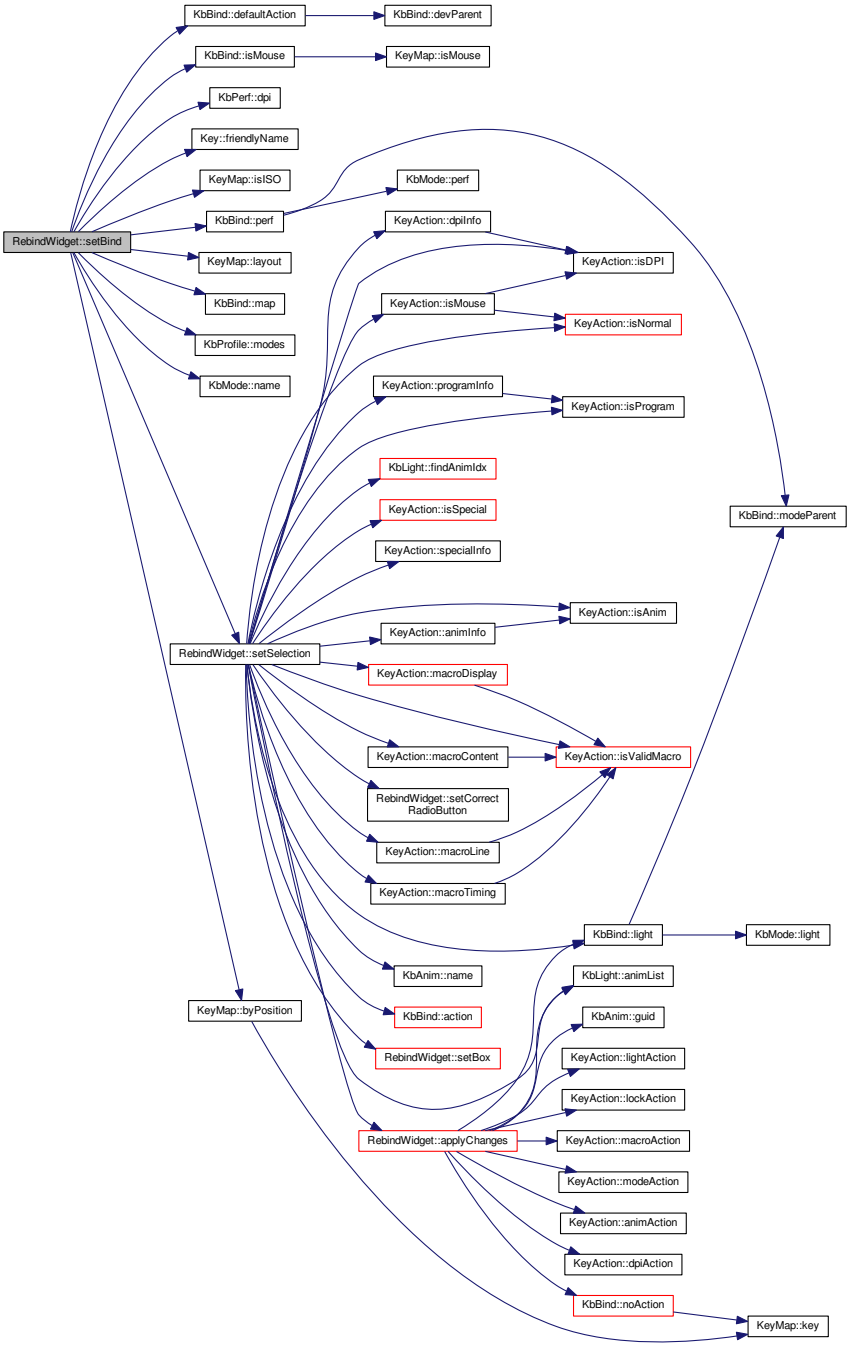
References bind, KeyMap::byPosition(), KbBind::defaultAction(), KbPerf::dpi(), KbPerf::DPI_COUNT, DPI_OFFSET, Ui_RebindWidget::dpiBox, Ui_RebindWidget::dpiButton, Ui_RebindWidget::dpiCustLabel, Ui_RebindWidget::dpiCustXBox, Ui_RebindWidget::dpiCustYBox, Ui_RebindWidget::dpiWarning, fnKeys, Key::friendlyName(), KeyMap::isISO(), KbBind::isMouse(), KeyMap::K95, KeyMap::layout(), KbBind::map(), mediaKeys, Ui_RebindWidget::modeBox, KbProfile::modes(), modKeys, KbMode::name(), numKeys, KbBind::perf(), profile, setSelection(), Ui_RebindWidget::typingBox, typingKeys, and ui.

Referenced by KbBindWidget::newLayout(), and KbBindWidget::setBind().

```
63                                                          {
64     bind = newBind;
65     profile = newProfile;
66     setSelection(QStringList());
67
68     // Populate typing keys by position
69     ui->typingBox->clear();
70     ui->typingBox->addItem(" ");
71     typingKeys.clear();
72     // Use the K95 map as it has all keys
73     const KeyMap& map = KeyMap(KeyMap::K95, bind->map().
       layout());
74     foreach(const QString& name, map.byPosition()){
75         KeyAction action(bind->defaultAction(name));
76         if(action.isNormal() && !modKeys.contains(name) && !fnKeys.contains(name) && !
       numKeys.contains(name) && !mediaKeys.contains(name) && name != "enter" && name != "tab" &&
       name != "bspace"){
```

```
77              const Key& pos = map[name];
78              QString friendly = pos.friendlyName();
79              ui->typingBox->addItem(friendly);
80              typingKeys.append(name);
81          }
82      }
83      typingKeys << "enter" << "tab" << "bspace";
84      ui->typingBox->addItem("Enter");
85      ui->typingBox->addItem("Tab");
86      ui->typingBox->addItem("Backspace");
87      if(!map.isISO()){
88          // Add ISO backslash (aka KEY_102ND) to ANSI options
89          typingKeys << "bslash_iso";
90          ui->typingBox->addItem("Backslash (ISO layout)");
91      }
92
93      // Populate mode list
94      ui->modeBox->clear();
95      ui->modeBox->addItem(" ");
96      ui->modeBox->addItem("(Previous)");
97      ui->modeBox->addItem("(Next)");
98      int idx = 1;
99      foreach(KbMode* mode, newProfile->modes())
100         ui->modeBox->addItem(QString("%1: %2").arg(idx++).arg(mode->
    name()));
101
102     // Enable/disable DPI based on device
103     if(bind->isMouse()){
104         ui->dpiButton->setEnabled(true);
105         ui->dpiBox->setEnabled(true);
106         ui->dpiWarning->hide();
107         // Fill DPI slots
108         const KbPerf* perf = bind->perf();
109         for(int i = 0; i < KbPerf::DPI_COUNT; i++){
110             bool sniper = (i == 0);
111             int boxIdx = i + DPI_OFFSET;
112             QPoint dpi = perf->dpi(i);
113             QString text = tr(sniper ? "Sniper:\t%1 x %2" : "%3:\t%1 x %2").arg(dpi.x()).arg(dpi.y());
114             if(!sniper) text = text.arg(i);
115             ui->dpiBox->setItemText(boxIdx, text);
116         }
117     } else {
118         ui->dpiButton->setEnabled(false);
119         ui->dpiBox->setEnabled(false);
120         ui->dpiWarning->show();
121     }
122     // Always disable custom DPI boxes until selected
123     ui->dpiCustXBox->setEnabled(false);
124     ui->dpiCustYBox->setEnabled(false);
125     ui->dpiCustLabel->setEnabled(false);
126 }
```

Here is the call graph for this function:



Here is the caller graph for this function:

12.80.3.48 void RebindWidget::setBox (QWidget * box) [private]

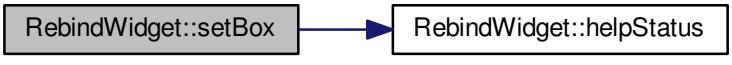
Definition at line 428 of file rebindwidget.cpp.

References Ui_RebindWidget::animBox, Ui_RebindWidget::dpiBox, Ui_RebindWidget::fnBox, helpStatus(), Ui_RebindWidget::lightBox, Ui_RebindWidget::lockBox, Ui_RebindWidget::mb2Box, Ui_RebindWidget::mbBox, Ui_RebindWidget::mediaBox, Ui_RebindWidget::modBox, Ui_RebindWidget::modeBox, Ui_RebindWidget::numBox, Ui_RebindWidget::programKpBox, Ui_RebindWidget::programKpButton, Ui_RebindWidget::programKrBox, Ui_RebindWidget::programKrButton, Ui_RebindWidget::pteMacroBox, Ui_RebindWidget::txtBuffer, Ui_RebindWidget::typingBox, ui, and Ui_RebindWidget::wheelBox.

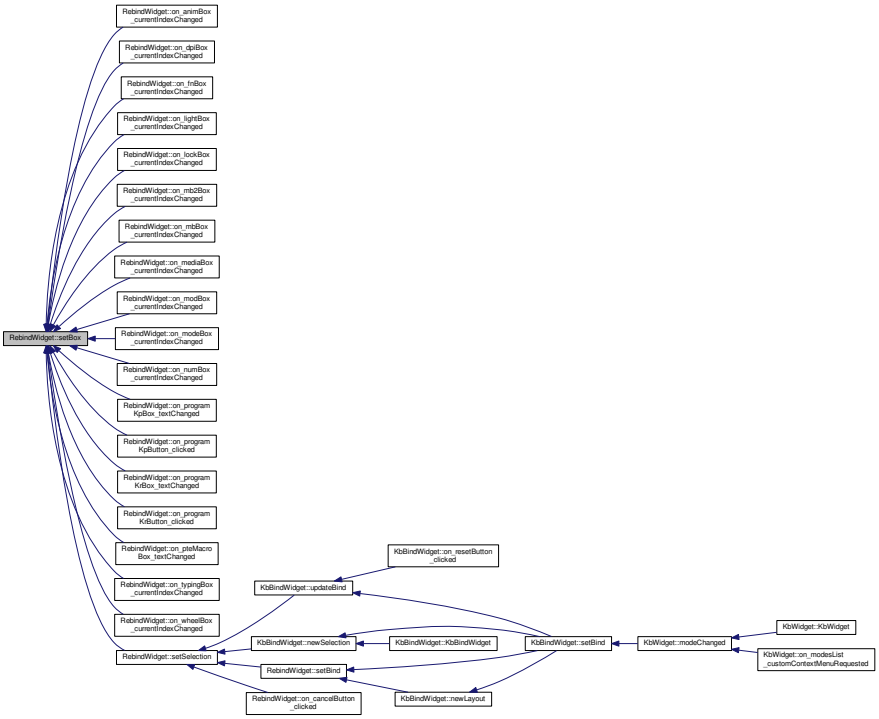
Referenced by on_animBox_currentIndexChanged(), on_dpiBox_currentIndexChanged(), on_fnBox_currentIndexChanged(), on_lightBox_currentIndexChanged(), on_lockBox_currentIndexChanged(), on_mb2Box_currentIndexChanged(), on_mbBox_currentIndexChanged(), on_mediaBox_currentIndexChanged(), on_modBox_currentIndexChanged(), on_modeBox_currentIndexChanged(), on_numBox_currentIndexChanged(), on_programKpBox_textChanged(), on_programKpButton_clicked(), on_programKrBox_textChanged(), on_programKrButton_clicked(), on_pteMacroBox_textChanged(), on_typingBox_currentIndexChanged(), on_wheelBox_currentIndexChanged(), and setSelection().

```
428                                     {
429     // Un-select every item except for the current one.
430     // on_*_currentIndexChanged will take care of deselecting the checkbox.
431     // Key
432     if(box != ui->typingBox) ui->typingBox->setCurrentIndex(0);
433     if(box != ui->modBox) ui->modBox->setCurrentIndex(0);
434     if(box != ui->fnBox) ui->fnBox->setCurrentIndex(0);
435     if(box != ui->numBox) ui->numBox->setCurrentIndex(0);
436     if(box != ui->mediaBox) ui->mediaBox->setCurrentIndex(0);
437     // Mouse
438     if(box != ui->mbBox) ui->mbBox->setCurrentIndex(0);
439     if(box != ui->mb2Box) ui->mb2Box->setCurrentIndex(0);
440     if(box != ui->wheelBox) ui->wheelBox->setCurrentIndex(0);
441     if(box != ui->dpiBox) ui->dpiBox->setCurrentIndex(0);
442     // Anim
443     if(box != ui->animBox) ui->animBox->setCurrentIndex(0);
444     // Special
445     if(box != ui->modeBox) ui->modeBox->setCurrentIndex(0);
446     if(box != ui->lightBox) ui->lightBox->setCurrentIndex(0);
447     if(box != ui->lockBox) ui->lockBox->setCurrentIndex(0);
448     // Program
449     if(box != ui->programKpBox && box != ui->programKrBox){
450         ui->programKpButton->setChecked(false);
451         ui->programKrButton->setChecked(false);
452     }
453     // Clear macro panel
454     if (box != ui->pteMacroBox) {
455         ui->pteMacroBox->setPlainText("");
456         ui->txtBuffer->setText("");
457         helpStatus(1);
458     }
459 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.80.3.49 void RebindWidget::setCorrectRadioButton (QString *macdef*) [private]

Parameters

macdef	Set the radiobutton for timing paramters according to the context. If no "=" followed by a number and comma can be found, it is the default button. If "=" can be found and numbers with more than one digit (means: $>$ 9), it is the "asTyped" button Otherwise it is the "no" button.

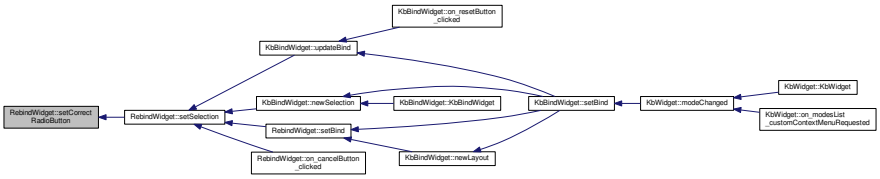
Definition at line 887 of file rebindwidget.cpp.

References Ui_RebindWidget::rb_delay_asTyped, Ui_RebindWidget::rb_delay_default, Ui_RebindWidget::rb_delay_no, and ui.

Referenced by setSelection().

```
887                                                     {
888     if (!macdef.contains(QRegExp("=\\d+,"))) {
889         ui->rb_delay_default->setChecked(true);
890         return;
891     }
892     if (macdef.contains(QRegExp("=\\d\\d+,"))) {
893         ui->rb_delay_asTyped->setChecked(true);
894         return;
895     }
896     ui->rb_delay_no->setChecked(true);
897 }
```

Here is the caller graph for this function:



12.80.3.50 void RebindWidget::setSelection (const QStringList & *newSelection,* bool *applyPrevious* = `false`)

Definition at line 128 of file rebindwidget.cpp.

References KbBind::action(), Ui_RebindWidget::animBox, KeyAction::animInfo(), Ui_RebindWidget::animKrBox, KbLight::animList(), Ui_RebindWidget::animOnceBox, applyChanges(), bind, KeyAction::DPI_CUSTOM, DPI_OFFSET, Ui_RebindWidget::dpiBox, Ui_RebindWidget::dpiCustXBox, Ui_RebindWidget::dpiCustYBox, KeyAction::dpiInfo(), KbLight::findAnimIdx(), Ui_RebindWidget::fnBox, fnKeys, KeyAction::isAnim(), KeyAction::isDPI(), KeyAction::isMouse(), KeyAction::isNormal(), KeyAction::isProgram(), KeyAction::isSpecial(), KeyAction::isValidMacro(), KbBind::light(), KeyAction::LIGHT_DOWN, KeyAction::LIGHT_DOWN_WRAP, KeyAction::LIGHT_UP, KeyAction::LIGHT_UP_WRAP, Ui_RebindWidget::lightBox, Ui_RebindWidget::lightWrapBox, Ui_RebindWidget::lockBox, KeyAction::macroContent(), KeyAction::macroDisplay(), KeyAction::macroLine(), KeyAction::macroTiming(), Ui_RebindWidget::mb2Box, Ui_RebindWidget::mbBox, Ui_RebindWidget::mediaBox, mediaKeys, Ui_RebindWidget::modBox, KeyAction::MODE_NEXT, KeyAction::MODE_NEXT_WRAP, KeyAction::MODE_PREV, KeyAction::MODE_PREV_WRAP, Ui_RebindWidget::modeBox, Ui_RebindWidget::modeWrapBox, modKeys, mouseExtKeys, mouseKeys, KbAnim::name(), Ui_RebindWidget::numBox, numKeys, KeyAction::PROGRAM_PR_MULTI, KeyAction::PROGRAM_RE_INDEF, KeyAction::PROGRAM_RE_KPSTOP, KeyAction::PROGRAM_RE_MULTI, KeyAction::programInfo(), Ui_RebindWidget::programKpBox, Ui_RebindWidget::programKpModeBox, Ui_RebindWidget::programKpSIBox, Ui_RebindWidget::programKrBox, Ui_RebindWidget::programKrModeBox, Ui_RebindWidget::programKrSIBox, Ui_RebindWidget::pteMacroBox, Ui_RebindWidget::pteMacroComment, Ui_RebindWidget::pteMacroText, selection, setBox(), setCorrectRadioButton(), KeyAction::specialInfo(), TAB_ANIM, TAB_KB, TAB_MACRO, TAB_MOUSE, TAB_PROGRAM, TAB_SPECIAL, Ui_RebindWidget::tabWidget, Ui_RebindWidget::txtBuffer, Ui_RebindWidget::typingBox, typingKeys, ui, Ui_RebindWidget::wheelBox, and wheelKeys.

Referenced by KbBindWidget::newSelection(), on_cancelButton_clicked(), setBind(), and KbBindWidget::updateBind().

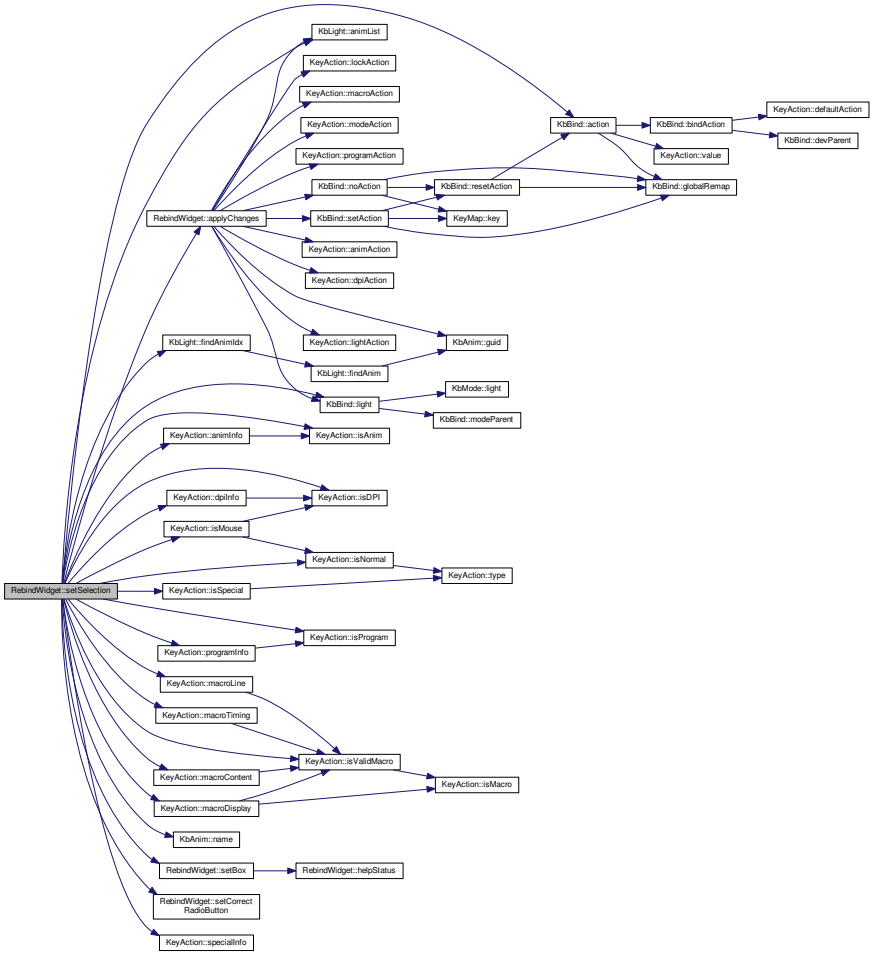
```
128                                                                                 {
129     if(bind){
130         // Populate animation list
131         // FIXME: There should be a trigger to update this whenever an animation is added/removed, even if
        the key selection hasn't changed
132         ui->animBox->clear();
133         ui->animBox->addItem(" ");
134         foreach(KbAnim* anim, bind->light()->animList())
135             ui->animBox->addItem(anim->name());
136     }
137
138     // Apply changes to previous selection (if any)
139     if(!selection.isEmpty() && applyPrevious){
140         QStringList previous = selection;
141         selection = QStringList();
142         applyChanges(previous, false);
143     }
144
145     selection = newSelection;
146     if(newSelection.isEmpty()){
147         hide();
148         return;
149     } else
150         show();
151
152     bool hasAction = false;
153     QString action;
154     if(selection.count() == 1){
```

```
155         // Select action from the UI
156         action = bind->action(selection.first());
157         if(!action.isEmpty())
158             hasAction = true;
159     } else {
160         // Select an action from the UI if all keys are the same.
161         // Select nothing if any are unbound or different
162         foreach(const QString& key, selection){
163             QString keyAction = bind->action(key);
164             if(keyAction.isEmpty() || (hasAction && keyAction != action)){
165                 hasAction = false;
166                 break;
167             } else if(action.isEmpty()){
168                 action = keyAction;
169                 hasAction = true;
170             }
171         }
172     }
173 
174     if(!hasAction)
175         action = "";
176     KeyAction act(action);
177     // Clear everything
178     setBox(0);
179     ui->dpiCustXBox->setValue(400);
180     ui->dpiCustYBox->setValue(400);
181     ui->animOnceBox->setChecked(false);
182     ui->animKrBox->setChecked(false);
183     ui->programKpBox->setText("");
184     ui->programKrBox->setText("");
185     ui->programKpSIBox->setChecked(false);
186     ui->programKrSIBox->setChecked(false);
187     ui->programKpModeBox->setCurrentIndex(0);
188     ui->programKrModeBox->setCurrentIndex(0);
189     ui->programKpModeBox->setEnabled(false);
190     ui->programKrModeBox->setEnabled(false);
191     // Clear neu UI elements in MacroTab
192     ui->pteMacroBox->setPlainText("");
193     ui->pteMacroText->setPlainText("");
194     ui->pteMacroComment->setPlainText("");
195     ui->txtBuffer->setText("");
196     // Fill in field and select tab according to action type
197     bool mouse = act.isMouse();
198     if(mouse){
199         // Mouse buttons
200         ui->tabWidget->setCurrentIndex(TAB_MOUSE);
201         // Set mouse buttons (indexOf returns -1 if not found, index zero is blank)
202         ui->mbBox->setCurrentIndex(mouseKeys.indexOf(action) + 1);
203         ui->mb2Box->setCurrentIndex(mouseExtKeys.indexOf(action) + 1);
204         ui->wheelBox->setCurrentIndex(wheelKeys.indexOf(action) + 1);
205         if(act.isDPI()){
206             QPoint custom;
207             int value = act.dpiInfo(custom);
208             if(value <= -DPI_OFFSET || value > KeyAction::DPI_CUSTOM)
209                 return;
210             ui->dpiBox->setCurrentIndex(value + DPI_OFFSET);
211             if(value == KeyAction::DPI_CUSTOM){
212                 ui->dpiCustXBox->setValue(custom.x());
213                 ui->dpiCustYBox->setValue(custom.y());
214             }
215         }
216     } else if(act.isNormal()){
217         // Standard key
218         ui->tabWidget->setCurrentIndex(TAB_KB);
219         ui->typingBox->setCurrentIndex(typingKeys.indexOf(action) + 1);
220         ui->modBox->setCurrentIndex(modKeys.indexOf(action) + 1);
221         ui->fnBox->setCurrentIndex(fnKeys.indexOf(action) + 1);
222         ui->numBox->setCurrentIndex(numKeys.indexOf(action) + 1);
223         ui->mediaBox->setCurrentIndex(mediaKeys.indexOf(action) + 1);
224     } else if(act.isProgram()){
225         // Program key
226         ui->tabWidget->setCurrentIndex(TAB_PROGRAM);
227         QString onPress, onRelease;
228         int stop = act.programInfo(onPress, onRelease);
229         ui->programKpBox->setText(onPress);
230         ui->programKrBox->setText(onRelease);
231         switch(stop & 0x0F){
232         case KeyAction::PROGRAM_PR_MULTI:
233             ui->programKpSIBox->setChecked(false);
234             ui->programKpModeBox->setCurrentIndex(0);
235             ui->programKpModeBox->setEnabled(false);
236             break;
237         default:
238             ui->programKpSIBox->setChecked(true);
239             ui->programKpModeBox->setCurrentIndex(stop & 0x0F);
240             ui->programKpModeBox->setEnabled(true);
241             break;
```

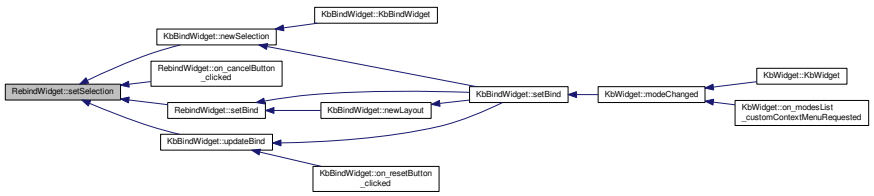
```
242         }
243         switch(stop & 0xF0){
244         case KeyAction::PROGRAM_RE_MULTI:
245             ui->programKrSIBox->setChecked(false);
246             ui->programKrModeBox->setCurrentIndex(0);
247             ui->programKrModeBox->setEnabled(false);
248             break;
249         case KeyAction::PROGRAM_RE_INDEF:
250             ui->programKrSIBox->setChecked(true);
251             ui->programKrModeBox->setCurrentIndex(0);
252             ui->programKrModeBox->setEnabled(true);
253             break;
254         case KeyAction::PROGRAM_RE_KPSTOP:
255             ui->programKrSIBox->setChecked(true);
256             ui->programKrModeBox->setCurrentIndex(1);
257             ui->programKrModeBox->setEnabled(true);
258             break;
259         }
260     } else if(act.isAnim()){
261         // Animation key
262         ui->tabWidget->setCurrentIndex(TAB_ANIM);
263         bool onlyOnce = false, stopOnRelease = false;
264         QUuid id = act.animInfo(onlyOnce, stopOnRelease);
265         // Find this animation in the list. If not found, -1 will be returned, resulting in the blank space
    being selected
266         int index = bind->light()->findAnimIdx(id);
267         ui->animBox->setCurrentIndex(index + 1);
268         ui->animOnceBox->setChecked(!onlyOnce);
269         ui->animKrBox->setChecked(stopOnRelease);
270     } else if(act.isSpecial()){
271         // Other special keys
272         ui->tabWidget->setCurrentIndex(TAB_SPECIAL);
273         int param;
274         QString sAction = act.specialInfo(param);
275         // Mode selection. Check wrap-around flag
276         ui->modeWrapBox->setChecked(true);
277         if(sAction == "mode"){
278             ui->modeWrapBox->setChecked(true);
279             if(param == KeyAction::MODE_PREV_WRAP)
280                 param = KeyAction::MODE_PREV;
281             else if(param == KeyAction::MODE_NEXT_WRAP)
282                 param = KeyAction::MODE_NEXT;
283             else if(param < 0)
284                 ui->modeWrapBox->setChecked(false);
285             // Set mode box to current selection, or to mode 1 if invalid
286             param += 3;
287             if(param >= 0 && param < ui->modeBox->count())
288                 ui->modeBox->setCurrentIndex(param);
289             else
290                 // 0 -> "", 1 -> Prev, 2 -> Next, 3 -> Mode 1
291                 ui->modeBox->setCurrentIndex(3);
292         } else if (sAction == "macro") {
293             ui->tabWidget->setCurrentIndex(TAB_MACRO);
294             if (act.isValidMacro()) {
295                 ui->pteMacroBox->setPlainText(act.macroContent());
296                 ui->pteMacroText->setPlainText(act.macroLine()[1].replace("
    &das_IST_31N_col0n;", ":"));
297                 ui->pteMacroComment->setPlainText(act.macroLine()[2].replace("
    &das_IST_31N_col0n;", ":"));
298                 // Set the invisible Buffer to the original timing information.
299                 // For convenience / Migration from older versions:
300                 // If the timing information is only "x", then ignore it by setting it to an empty QString.
301                 ui->txtBuffer->setText("");
302                 if (act.macroTiming() != "x") ui->txtBuffer->setText(act.macroTiming());
303                 setCorrectRadioButton(act.macroContent());
304             } else {
305                 qDebug("RebindWidget::setSelection found invalid macro definition.");
306                 act.macroDisplay();
307             }
308         } else
309             ui->modeBox->setCurrentIndex(0);
310         // Brightness control. Also check wrap
311         if(sAction == "light"){
312             ui->lightWrapBox->setChecked(true);
313             if(param == KeyAction::LIGHT_DOWN_WRAP)
314                 param = KeyAction::LIGHT_DOWN;
315             else if(param == KeyAction::LIGHT_UP_WRAP)
316                 param = KeyAction::LIGHT_UP;
317             else
318                 ui->lightWrapBox->setChecked(false);
319             if(param < 0 || param > 1)
320                 param = -1;
321             ui->lightBox->setCurrentIndex(param + 1);
322         } else
323             ui->lightBox->setCurrentIndex(0);
324         // Win lock
325         if(sAction == "lock"){
```

```
326             if(param < 0 || param > 2)
327                 param = -1;
328             ui->lockBox->setCurrentIndex(param + 1);
329         } else
330             ui->lockBox->setCurrentIndex(0);
331     }
332 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.80.4 Field Documentation

12.80.4.1 KbBind* RebindWidget::bind [private]

Definition at line 94 of file rebindwidget.h.

Referenced by applyChanges(), on_btnStartMacro_clicked(), on_btnStopMacro_clicked(), on_resetButton_clicked(), on_unbindButton_clicked(), setBind(), and setSelection().

12.80.4.2 QStringList RebindWidget::fnKeys [private]

Definition at line 100 of file rebindwidget.h.

Referenced by applyChanges(), RebindWidget(), setBind(), and setSelection().

12.80.4.3 MacroReader* RebindWidget::macReader [private]

Definition at line 106 of file rebindwidget.h.

Referenced by on_btnStartMacro_clicked(), and on_btnStopMacro_clicked().

12.80.4.4 QStringList RebindWidget::mediaKeys [private]

Definition at line 102 of file rebindwidget.h.

Referenced by applyChanges(), RebindWidget(), setBind(), and setSelection().

12.80.4.5 QStringList RebindWidget::modKeys [private]

Definition at line 99 of file rebindwidget.h.

Referenced by applyChanges(), RebindWidget(), setBind(), and setSelection().

12.80.4.6 QStringList RebindWidget::mouseExtKeys [private]

Definition at line 104 of file rebindwidget.h.

Referenced by applyChanges(), RebindWidget(), and setSelection().

12.80.4.7 QStringList RebindWidget::mouseKeys [private]

Definition at line 103 of file rebindwidget.h.

Referenced by applyChanges(), RebindWidget(), and setSelection().

12.80.4.8 QStringList RebindWidget::numKeys [private]

Definition at line 101 of file rebindwidget.h.

Referenced by applyChanges(), RebindWidget(), setBind(), and setSelection().

12.80.4.9 KbProfile* RebindWidget::profile [private]

Definition at line 95 of file rebindwidget.h.

Referenced by setBind().

12.80.4.10 QStringList RebindWidget::selection [private]

Definition at line 96 of file rebindwidget.h.

Referenced by on_applyButton_clicked(), on_cancelButton_clicked(), on_resetButton_clicked(), on_unbindButton_clicked(), and setSelection().

12.80.4.11 const int RebindWidget::TAB_ANIM = 2 [static], [private]

Definition at line 83 of file rebindwidget.h.

Referenced by setSelection().

12.80.4.12 const int RebindWidget::TAB_KB = 0 [static], [private]

Definition at line 83 of file rebindwidget.h.

Referenced by setSelection().

12.80.4.13 const int RebindWidget::TAB_MACRO = 5 [static], [private]

Definition at line 83 of file rebindwidget.h.

Referenced by setSelection().

12.80.4.14 const int RebindWidget::TAB_MOUSE = 1 [static], [private]

Definition at line 83 of file rebindwidget.h.

Referenced by setSelection().

12.80.4.15 const int RebindWidget::TAB_PROGRAM = 4 [static], [private]

Definition at line 83 of file rebindwidget.h.

Referenced by setSelection().

12.80.4.16 const int RebindWidget::TAB_SPECIAL = 3 [static], [private]

Definition at line 83 of file rebindwidget.h.

Referenced by setSelection().

12.80.4.17 QStringList RebindWidget::typingKeys [private]

Definition at line 98 of file rebindwidget.h.

Referenced by applyChanges(), setBind(), and setSelection().

12.80.4.18 Ui::RebindWidget* RebindWidget::ui [private]

Definition at line 80 of file rebindwidget.h.

Referenced by applyChanges(), convertMacroBox(), helpStatus(), on_animBox_currentIndexChanged(), on_animButton_clicked(), on_btnStartMacro_clicked(), on_btnStopMacro_clicked(), on_dpiBox_currentIndexChanged(),

on_dpiButton_clicked(), on_fnBox_currentIndexChanged(), on_fnButton_clicked(), on_lightBox_currentIndexChanged(), on_lightButton_clicked(), on_lockBox_currentIndexChanged(), on_lockButton_clicked(), on_mb2Box_currentIndexChanged(), on_mb2Button_clicked(), on_mbBox_currentIndexChanged(), on_mbButton_clicked(), on_mediaBox_currentIndexChanged(), on_mediaButton_clicked(), on_modBox_currentIndexChanged(), on_modButton_clicked(), on_modeBox_currentIndexChanged(), on_modeButton_clicked(), on_numBox_currentIndexChanged(), on_numButton_clicked(), on_programKpBox_textChanged(), on_programKpButton_clicked(), on_programKpSIBox_clicked(), on_programKrBox_textChanged(), on_programKrButton_clicked(), on_programKrSIBox_clicked(), on_pteMacroBox_textChanged(), on_typingBox_currentIndexChanged(), on_typingButton_clicked(), on_wheelBox_currentIndexChanged(), on_wheelButton_clicked(), RebindWidget(), setBind(), setBox(), setCorrectRadioButton(), setSelection(), and ~RebindWidget().

12.80.4.19 QStringList RebindWidget::wheelKeys [private]

Definition at line 105 of file rebindwidget.h.

Referenced by applyChanges(), RebindWidget(), and setSelection().

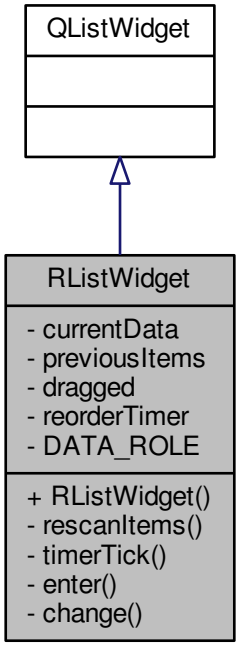
The documentation for this class was generated from the following files:

- src/ckb/rebindwidget.h
- src/ckb/rebindwidget.cpp

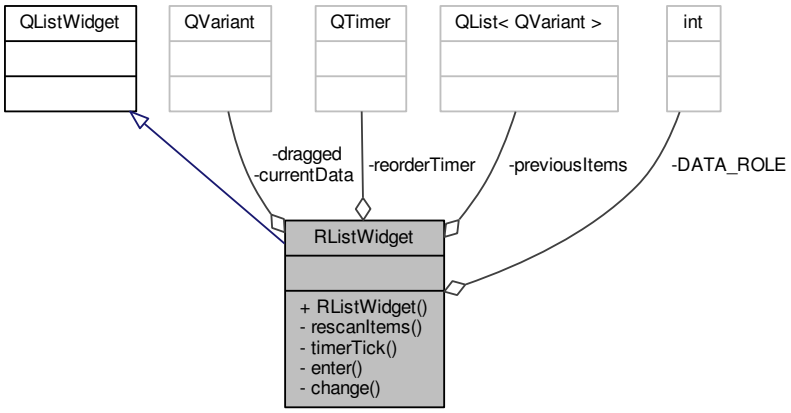
12.81 RListWidget Class Reference

`#include <src/ckb/rlistwidget.h>`

Inheritance diagram for RListWidget:



Collaboration diagram for RListWidget:



Signals

- void orderChanged ()

Public Member Functions

- RListWidget (QWidget ∗parent=0)

Private Slots

- void timerTick ()
- void enter (QListWidgetItem ∗item)
- void change (QListWidgetItem ∗item)

Private Member Functions

- void rescanItems ()

Private Attributes

- QVariant currentData
- QList< QVariant > previousItems
- QVariant dragged
- QTimer reorderTimer

Static Private Attributes

- static const int DATA_ROLE = Qt::UserRole + 100

12.81.1 Detailed Description

Definition at line 9 of file rlistwidget.h.

12.81.2 Constructor & Destructor Documentation

12.81.2.1 RListWidget::RListWidget (QWidget * *parent* = 0) [explicit]

Definition at line 4 of file rlistwidget.cpp.

References change(), enter(), reorderTimer, and timerTick().

```
4                                          :
5     QListWidget(parent)
6 {
7     setDragDropMode(QAbstractItemView::InternalMove);
8     setMovement(QListView::Snap);
9
10     reorderTimer.setSingleShot(true);
11     reorderTimer.setInterval(100);
12     connect(&reorderTimer, SIGNAL(timeout()), this, SLOT(timerTick()));
13
14     connect(this, SIGNAL(itemEntered(QListWidgetItem*)), this, SLOT(enter(QListWidgetItem*)));
15     connect(this, SIGNAL(itemChanged(QListWidgetItem*)), this, SLOT(change(QListWidgetItem*)));
16 }
```

Here is the call graph for this function:

12.81.3 Member Function Documentation

12.81.3.1 void RListWidget::change (QListWidgetItem * *item*) [private], [slot]

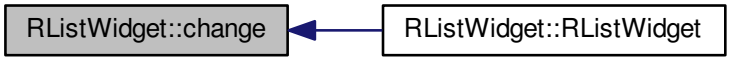
Definition at line 54 of file rlistwidget.cpp.

References reorderTimer.

Referenced by RListWidget().

```
54                                                {
55     reorderTimer.stop();
56     reorderTimer.start();
57 }
```

Here is the caller graph for this function:



12.81.3.2 void RListWidget::enter (QListWidgetItem * *item*) [private],[slot]

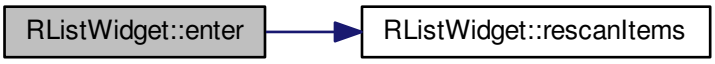
Definition at line 47 of file rlistwidget.cpp.

References DATA_ROLE, dragged, and rescanItems().

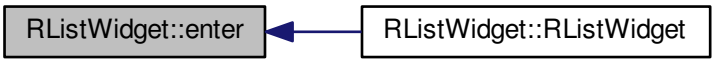
Referenced by RListWidget().

```
47                                              {
48     rescanItems();
49     // Check for drag+drop setup
50     if(item)
51         dragged = item->data(DATA_ROLE);
52 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



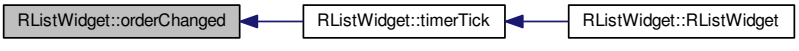
12.81.3.3 void RListWidget::orderChanged ( ) [signal]

Definition at line 141 of file moc_rlistwidget.cpp.

Referenced by timerTick().

```
142 {
143     QMetaObject::activate(this, &staticMetaObject, 0, Q_NULLPTR);
144 }
```

Here is the caller graph for this function:



12.81.3.4 void RListWidget::rescanItems ( ) [private]

Definition at line 59 of file rlistwidget.cpp.

References DATA_ROLE, and previousItems.

Referenced by enter().

```
59                              {
60      QList<QVariant> newItems;
61      int c = count();
62      for(int i = 0; i < c; i++){
63          QVariant data = this->item(i)->data(DATA_ROLE);
64          if(data.isNull()){
65              // Generate the ID for this item if it doesn't already exist
66              data = QUuid::createUuid();
67              this->item(i)->setData(DATA_ROLE, data);
68          }
69          newItems.append(data);
70      }
71      previousItems = newItems;
72  }
```

Here is the caller graph for this function:

12.81.3.5 void RListWidget::timerTick ( ) [private], [slot]

Definition at line 18 of file rlistwidget.cpp.

References DATA_ROLE, dragged, orderChanged(), and previousItems.

Referenced by RListWidget().

```
18                            {
19      bool reordered = false;
20      QListWidgetItem* reselect = 0;
21      QList<QVariant> newItems;
22      // Scan the item list to see if they changed
23      int c = count();
24      for(int i = 0; i < c; i++){
25          QListWidgetItem* itm = item(i);
26          QVariant data = itm->data(DATA_ROLE);
27          newItems.append(data);
28          if(i >= previousItems.count() || data != previousItems[i])
29              reordered = true;
30          // Re-select the dragged item (if any)
31          if(data == dragged)
32              reselect = itm;
33      }
34      if(previousItems.length() != newItems.length())
35          return;
36      if(reordered){
37          previousItems = newItems;
38          emit orderChanged();
39          if(reselect){
40              reselect->setSelected(true);
41              setCurrentItem(reselect);
42              dragged = QVariant();
43          }
44      }
45  }
```

Here is the caller graph for this function:

12.81.4 Field Documentation

12.81.4.1 QVariant RListWidget::currentData [private]

Definition at line 24 of file rlistwidget.h.

12.81.4.2 const int RListWidget::DATA_ROLE = Qt::UserRole + 100 [static], [private]

Definition at line 29 of file rlistwidget.h.

Referenced by enter(), rescanItems(), and timerTick().

12.81.4.3 QVariant RListWidget::dragged [private]

Definition at line 26 of file rlistwidget.h.

Referenced by enter(), and timerTick().

12.81.4.4 QList<QVariant> RListWidget::previousItems [private]

Definition at line 25 of file rlistwidget.h.

Referenced by rescanItems(), and timerTick().

12.81.4.5 QTimer RListWidget::reorderTimer [private]

Definition at line 27 of file rlistwidget.h.

Referenced by change(), and RListWidget().

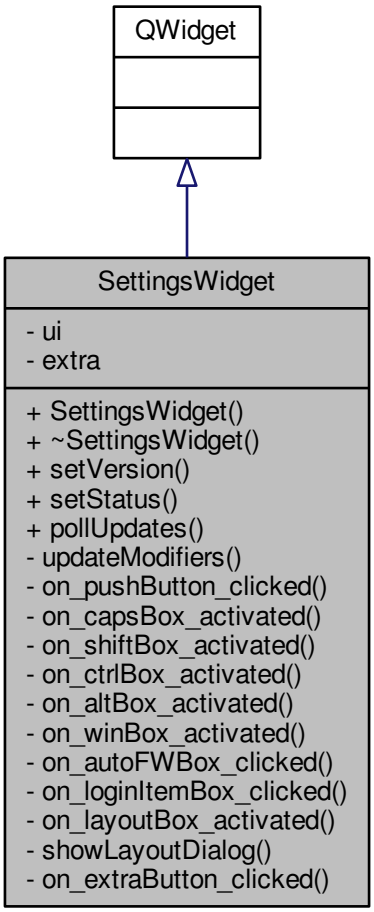
The documentation for this class was generated from the following files:

- src/ckb/rlistwidget.h
- src/ckb/release/moc_rlistwidget.cpp
- src/ckb/rlistwidget.cpp

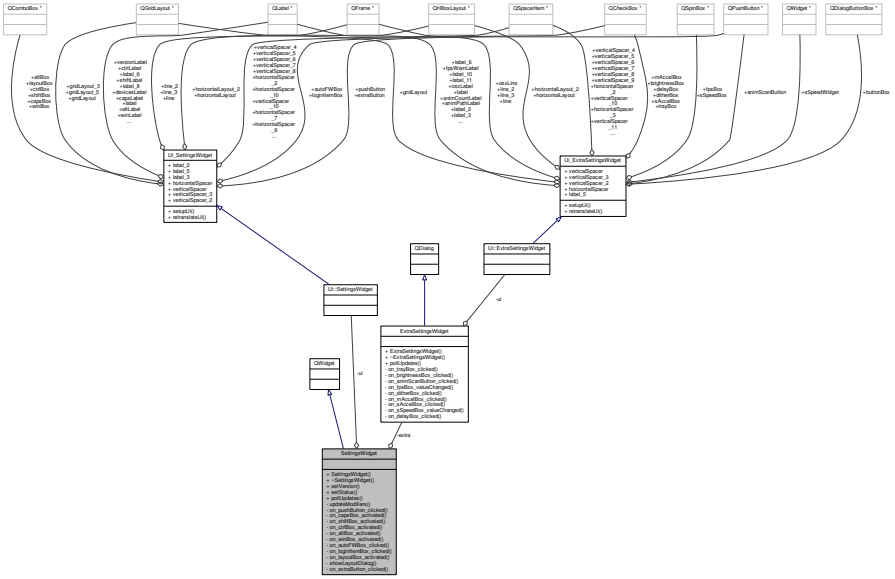
12.82 SettingsWidget Class Reference

`#include <src/ckb/settingswidget.h>`

Inheritance diagram for SettingsWidget:



Collaboration diagram for SettingsWidget:



Public Member Functions

- SettingsWidget (QWidget *parent=0)
- ~SettingsWidget ()
- void setVersion (const QString &version)
- void setStatus (const QString &text)
- void pollUpdates ()

Private Slots

- void on_pushButton_clicked ()
- void on_capsBox_activated (int index)
- void on_shiftBox_activated (int index)
- void on_ctrlBox_activated (int index)
- void on_altBox_activated (int index)
- void on_winBox_activated (int index)
- void on_autoFWBox_clicked (bool checked)
- void on_loginItemBox_clicked (bool checked)
- void on_layoutBox_activated (int index)
- void showLayoutDialog ()
- void on_extraButton_clicked ()

Private Member Functions

- void updateModifiers ()

Private Attributes

- Ui::SettingsWidget * ui
- ExtraSettingsWidget * extra

Friends

- class MainWindow

12.82.1 Detailed Description

Definition at line 11 of file settingswidget.h.

12.82.2 Constructor & Destructor Documentation

12.82.2.1 SettingsWidget::SettingsWidget (QWidget * *parent* = 0) [explicit]

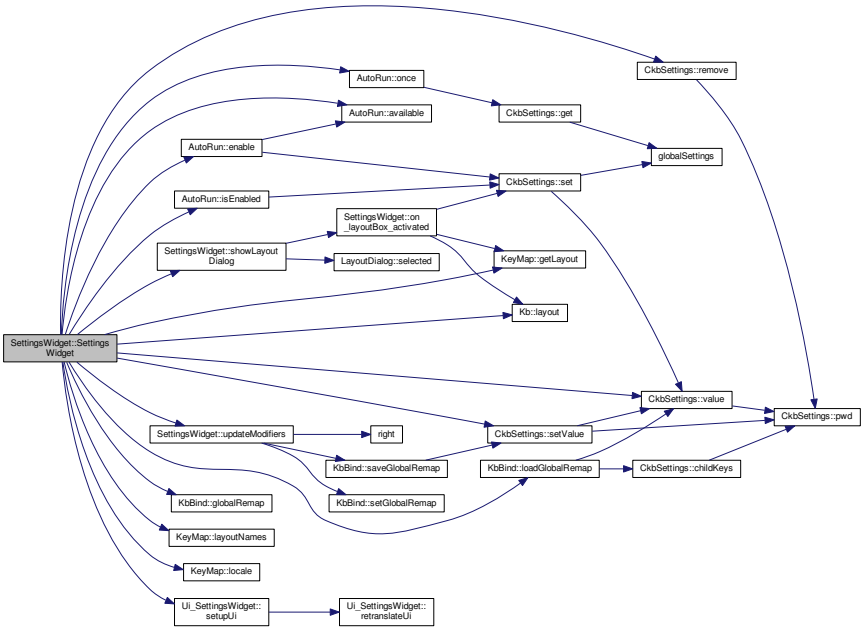
Definition at line 15 of file settingswidget.cpp.

References Ui_SettingsWidget::altBox, Ui_SettingsWidget::autoFWBox, AutoRun::available(), Ui_SettingsWidget::capsBox, Ui_SettingsWidget::ctrlBox, AutoRun::enable(), extra, KeyMap::getLayout(), KbBind::globalRemap(), AutoRun::isEnabled(), Kb::layout(), Ui_SettingsWidget::layoutBox, KeyMap::layoutNames(), KbBind::loadGlobalRemap(), KeyMap::locale(), Ui_SettingsWidget::loginItemBox, modKeys, modNames, KeyMap::NO_LAYOUT, AutoRun::once(), CkbSettings::remove(), Ui_SettingsWidget::setupUi(), CkbSettings::setValue(), Ui_SettingsWidget::shiftBox, showLayoutDialog(), ui, updateModifiers(), CkbSettings::value(), and Ui_SettingsWidget::winBox.

```
15                                                  :
16     QWidget(parent),
17     ui(new Ui::SettingsWidget)
18 {
19     ui->setupUi(this);
20     CkbSettings settings("Program");
21
22     // Read keyboard layout
23     ui->layoutBox->addItems(KeyMap::layoutNames());
24     KeyMap::Layout layout = KeyMap::getLayout(settings.value("KbdLayout").
       toString());
25     if(layout == KeyMap::NO_LAYOUT){
26         // If the layout hasn't been set yet, show a dialog to let the user choose it
27         layout = KeyMap::locale();
28         QTimer::singleShot(1000, this, SLOT(showLayoutDialog()));   // Run the function
       after a delay as the dialog may not appear correctly otherwise
29         settings.setValue("KbdLayout", KeyMap::getLayout(layout));
30     }
31     Kb::layout(layout);
32     ui->layoutBox->setCurrentIndex((int)layout);
33
34     // Load modifier remap
35     KbBind::loadGlobalRemap();
36     if(modKeys.isEmpty()){
37         modKeys << "caps" << "lshift" << "lctrl" << "lalt" << "lwin";
38 #ifdef Q_OS_MACX
39         modNames << "Caps Lock" << "Shift" << "Control ()" << "Option ()" << "Command ()";
40 #else
41         modNames << "Caps Lock" << "Shift" << "Control" << "Alt" << "Super";
42 #endif
43     }
44     foreach(const QString& name, modNames){
45         ui->capsBox->addItem(name);
46         ui->shiftBox->addItem(name);
47         ui->ctrlBox->addItem(name);
48         ui->altBox->addItem(name);
49         ui->winBox->addItem(name);
50     }
51
52     int idx = modKeys.indexOf(KbBind::globalRemap("caps"));
53     if(idx >= 0)
54         ui->capsBox->setCurrentIndex(idx);
55     idx = modKeys.indexOf(KbBind::globalRemap("lshift"));
56     if(idx >= 0)
57         ui->shiftBox->setCurrentIndex(idx);
58     idx = modKeys.indexOf(KbBind::globalRemap("lctrl"));
59     if(idx >= 0)
60         ui->ctrlBox->setCurrentIndex(idx);
61     idx = modKeys.indexOf(KbBind::globalRemap("lwin"));
62     if(idx >= 0)
63         ui->winBox->setCurrentIndex(idx);
64     idx = modKeys.indexOf(KbBind::globalRemap("lalt"));
65     if(idx >= 0)
66         ui->altBox->setCurrentIndex(idx);
```

```
67 #ifdef Q_OS_MACX
68     if(settings.value("osxCmdSwap").toBool()){
69         // ckb <0.0.41
70         settings.remove("osxCmdSwap");
71         ui->ctrlBox->setCurrentIndex(modKeys.indexOf("lwin"));
72         ui->winBox->setCurrentIndex(modKeys.indexOf("lctrl"));
73         updateModifiers();
74     }
75 #endif
76
77     // Read auto update settings
78     ui->autoFWBox->setChecked(!settings.value("DisableAutoFWCheck").toBool());
79
80     // Read auto run settings
81     if(!AutoRun::available())
82         ui->loginItemBox->hide();
83     else {
84         if(!AutoRun::once())
85             // If this is the first time running the app, enable auto run by default
86             AutoRun::enable();
87         ui->loginItemBox->setChecked(AutoRun::isEnabled());
88     }
89
90     // Prepare extra settings
91     extra = new ExtraSettingsWidget(this);
92 }
```

Here is the call graph for this function:



12.82.2.2 SettingsWidget::~SettingsWidget ( )

Definition at line 94 of file settingswidget.cpp.

References ui.

```
94                                {
95     delete ui;
96 }
```

12.82.3 Member Function Documentation

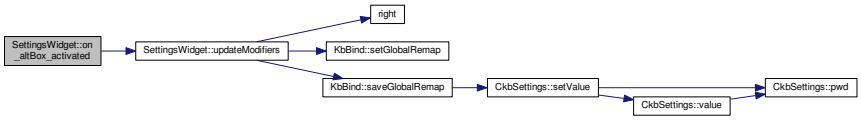
12.82.3.1 void SettingsWidget::on_altBox_activated (int *index*) [private], [slot]

Definition at line 155 of file settingswidget.cpp.

References updateModifiers().

```
155                                                 {
156     updateModifiers();
157 }
```

Here is the call graph for this function:



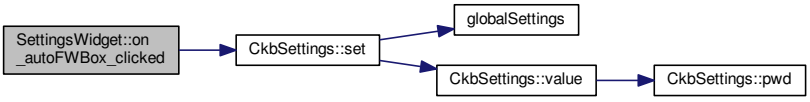
12.82.3.2 void SettingsWidget::on_autoFWBox_clicked (bool *checked*) [private], [slot]

Definition at line 163 of file settingswidget.cpp.

References CkbSettings::set().

```
163                                                    {
164     CkbSettings::set("Program/DisableAutoFWCheck", !checked);
165 }
```

Here is the call graph for this function:



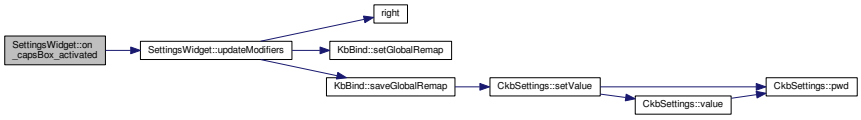
12.82.3.3 void SettingsWidget::on_capsBox_activated (int *index*) [private], [slot]

Definition at line 143 of file settingswidget.cpp.

References updateModifiers().

```
143                                                 {
144     updateModifiers();
145 }
```

Here is the call graph for this function:



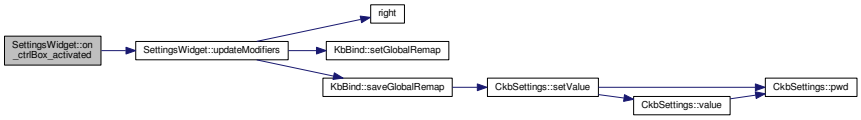
12.82.3.4 void SettingsWidget::on_ctrlBox_activated (int *index*) [private], [slot]

Definition at line 151 of file settingswidget.cpp.

References updateModifiers().

```
151                                                 {
152     updateModifiers();
153 }
```

Here is the call graph for this function:



12.82.3.5 void SettingsWidget::on_extraButton_clicked ( ) [private], [slot]

Definition at line 180 of file settingswidget.cpp.

References extra.

```
180                                             {
181     extra->exec();
182 }
```

12.82.3.6 void SettingsWidget::on_layoutBox_activated (int *index*) [private], [slot]

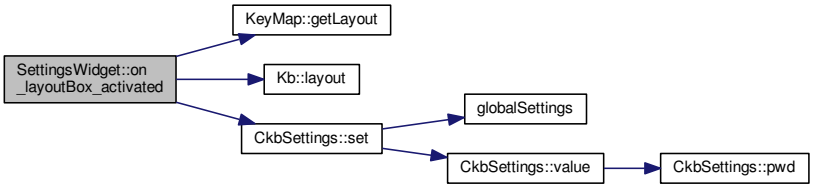
Definition at line 174 of file settingswidget.cpp.

References KeyMap::getLayout(), Kb::layout(), and CkbSettings::set().

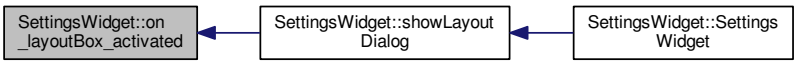
Referenced by showLayoutDialog().

```
174                                                    {
175     KeyMap::Layout layout = (KeyMap::Layout)index;
176     CkbSettings::set("Program/KbdLayout", KeyMap::getLayout(layout));
177     Kb::layout(layout);
178 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.82.3.7 void SettingsWidget::on_loginItemBox_clicked (bool *checked*) [private],[slot]

Definition at line 167 of file settingswidget.cpp.

References AutoRun::disable(), and AutoRun::enable().

```
167                                                        {
168     if(checked)
169         AutoRun::enable();
170     else
171         AutoRun::disable();
172 }
```

Here is the call graph for this function:

12.82.3.8 void SettingsWidget::on_pushButton_clicked ( ) [private],[slot]

Definition at line 139 of file settingswidget.cpp.

```
139                                          {
140     qApp->quit();
141 }
```

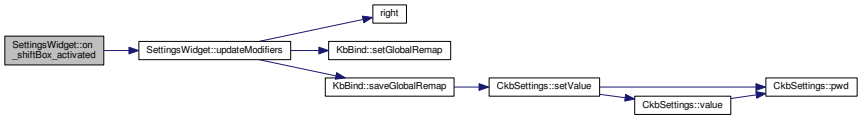
12.82.3.9 void SettingsWidget::on_shiftBox_activated (int *index*) [private],[slot]

Definition at line 147 of file settingswidget.cpp.

References updateModifiers().

```
147                                                  {
148     updateModifiers();
149 }
```

Here is the call graph for this function:



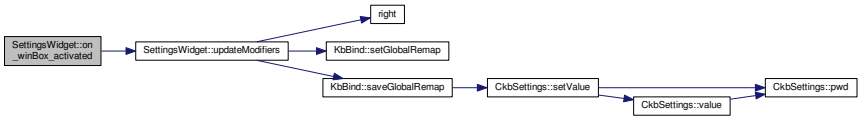
12.82.3.10 void SettingsWidget::on_winBox_activated (int *index*) [private], [slot]

Definition at line 159 of file settingswidget.cpp.

References updateModifiers().

```
159                                                 {
160     updateModifiers();
161 }
```

Here is the call graph for this function:



12.82.3.11 void SettingsWidget::pollUpdates ( )

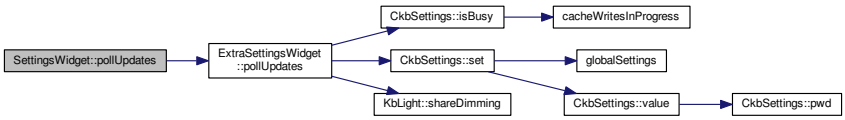
Definition at line 106 of file settingswidget.cpp.

References extra, and ExtraSettingsWidget::pollUpdates().

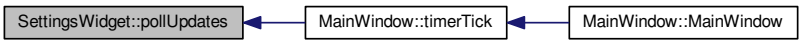
Referenced by MainWindow::timerTick().

```
106                                  {
107     extra->pollUpdates();
108 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.82.3.12 void SettingsWidget::setStatus (const QString & *text*)

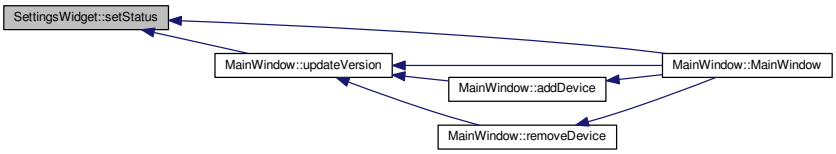
Definition at line 110 of file settingswidget.cpp.

References Ui_SettingsWidget::devicesLabel, and ui.

Referenced by MainWindow::MainWindow(), and MainWindow::updateVersion().

```
110                                                 {
111     ui->devicesLabel->setText(text);
112 }
```

Here is the caller graph for this function:



12.82.3.13 void SettingsWidget::setVersion (const QString & *version*)

Definition at line 114 of file settingswidget.cpp.

References ui, and Ui_SettingsWidget::versionLabel.

Referenced by MainWindow::MainWindow().

```
114                                                     {
115     ui->versionLabel->setText(version);
116 }
```

Here is the caller graph for this function:

12.82.3.14 void SettingsWidget::showLayoutDialog ( ) [private],[slot]

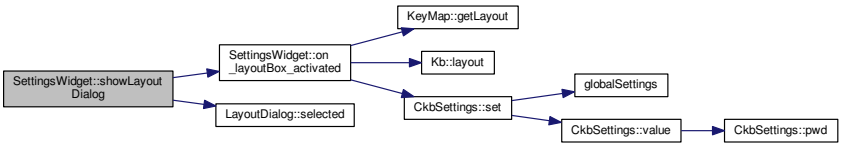
Definition at line 98 of file settingswidget.cpp.

References Ui_SettingsWidget::layoutBox, on_layoutBox_activated(), LayoutDialog::selected(), and ui.

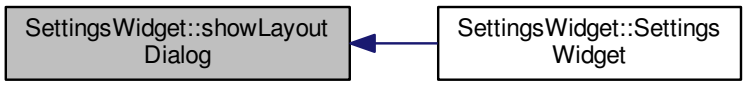
Referenced by SettingsWidget().

```
98                                      {
99     LayoutDialog dialog(this);
100     dialog.exec();
101     // Set selected layout
102     ui->layoutBox->setCurrentIndex((int)dialog.selected());
103     on_layoutBox_activated((int)dialog.selected());         // Call activated()
        signal manually to trigger save
104 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.82.3.15 void SettingsWidget::updateModifiers ( ) [private]

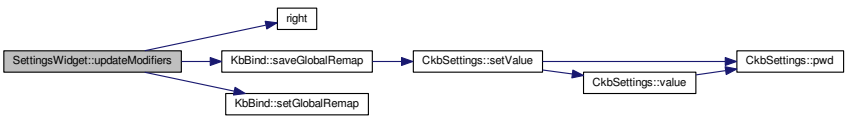
Definition at line 124 of file settingswidget.cpp.

References Ui_SettingsWidget::altBox, Ui_SettingsWidget::capsBox, Ui_SettingsWidget::ctrlBox, modKeys, right(), KbBind::saveGlobalRemap(), KbBind::setGlobalRemap(), Ui_SettingsWidget::shiftBox, ui, and Ui_SettingsWidget-::winBox.

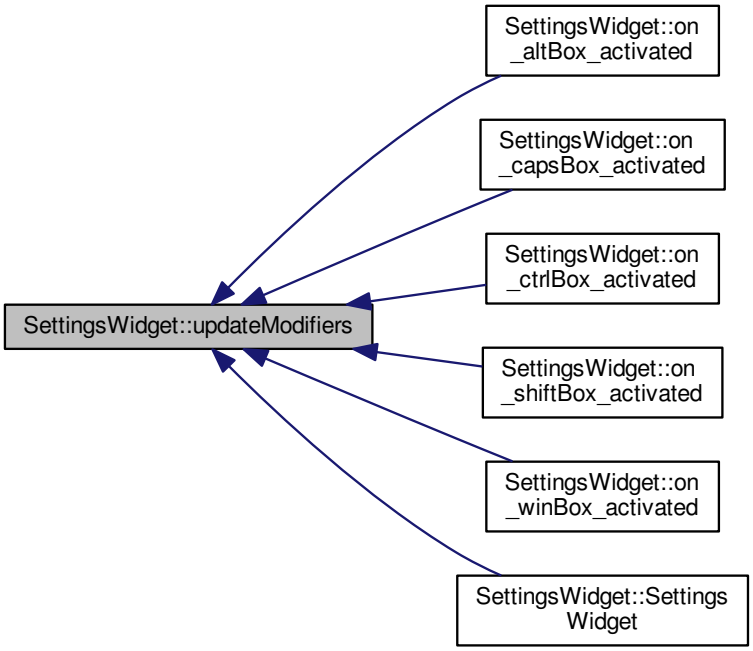
Referenced by on_altBox_activated(), on_capsBox_activated(), on_ctrlBox_activated(), on_shiftBox_activated(), on_winBox_activated(), and SettingsWidget().

```
124                                  {
125     QHash<QString, QString> newMods;
126     newMods["caps"] = modKeys[ui->capsBox->currentIndex()];
127     newMods["lshift"] = modKeys[ui->shiftBox->currentIndex()];
128     newMods["rshift"] = right(modKeys[ui->shiftBox->currentIndex()]);
129     newMods["lctrl"] = modKeys[ui->ctrlBox->currentIndex()];
130     newMods["rctrl"] = right(modKeys[ui->ctrlBox->currentIndex()]);
131     newMods["lalt"] = modKeys[ui->altBox->currentIndex()];
132     newMods["ralt"] = right(modKeys[ui->altBox->currentIndex()]);
133     newMods["lwin"] = modKeys[ui->winBox->currentIndex()];
134     newMods["rwin"] = right(modKeys[ui->winBox->currentIndex()]);
135     KbBind::setGlobalRemap(newMods);
136     KbBind::saveGlobalRemap();
137 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.82.4 Friends And Related Function Documentation

12.82.4.1 friend class MainWindow [friend]

Definition at line 44 of file settingswidget.h.

12.82.5 Field Documentation

12.82.5.1 ExtraSettingsWidget* SettingsWidget::extra [private]

Definition at line 46 of file settingswidget.h.

Referenced by on_extraButton_clicked(), pollUpdates(), and SettingsWidget().

12.82.5.2 Ui::SettingsWidget* SettingsWidget::ui [private]

Definition at line 43 of file settingswidget.h.

Referenced by setStatus(), SettingsWidget(), setVersion(), showLayoutDialog(), updateModifiers(), and ~SettingsWidget().

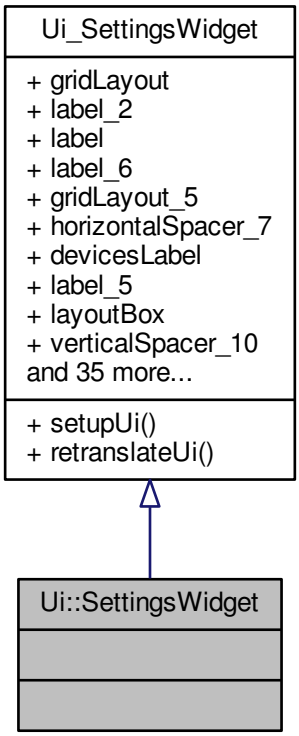
The documentation for this class was generated from the following files:

- src/ckb/settingswidget.h
- src/ckb/settingswidget.cpp

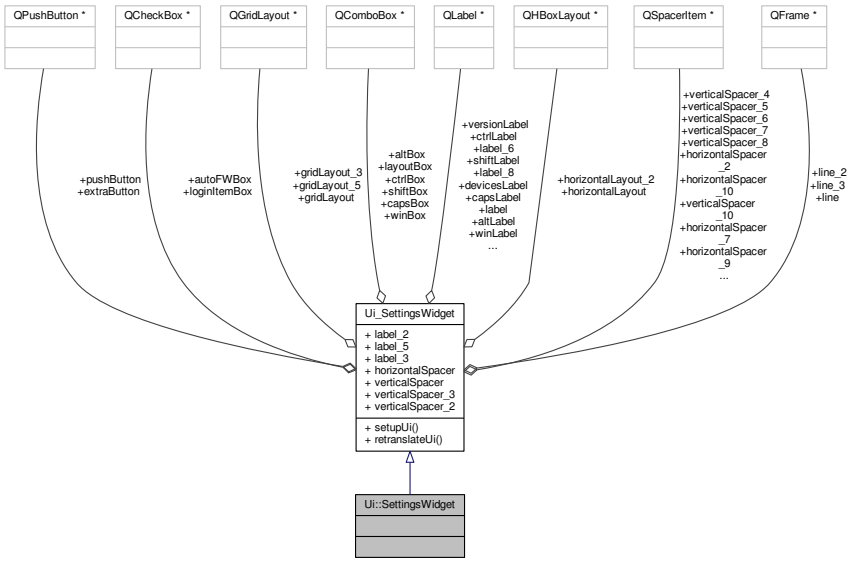
12.83 Ui::SettingsWidget Class Reference

`#include <src/ckb/ui_settingswidget.h>`

Inheritance diagram for Ui::SettingsWidget:



Collaboration diagram for Ui::SettingsWidget:



Public Member Functions

- void setupUi (QWidget *SettingsWidget)
- void retranslateUi (QWidget *SettingsWidget)

Data Fields

- QGridLayout * gridLayout
- QLabel * label_2
- QLabel * label
- QLabel * label_6
- QGridLayout * gridLayout_5
- QSpacerItem * horizontalSpacer_7
- QLabel * devicesLabel
- QLabel * label_5
- QComboBox * layoutBox
- QSpacerItem * verticalSpacer_10
- QHBoxLayout * horizontalLayout_2
- QLabel * label_3
- QSpacerItem * horizontalSpacer
- QPushButton * pushButton
- QSpacerItem * verticalSpacer
- QSpacerItem * verticalSpacer_8
- QCheckBox * loginItemBox
- QSpacerItem * verticalSpacer_6
- QCheckBox * autoFWBox
- QFrame * line_2
- QLabel * label_8
- QGridLayout * gridLayout_3
- QLabel * shiftLabel
- QLabel * capsLabel

- QSpacerItem * horizontalSpacer_10
- QComboBox * capsBox
- QLabel * ctrlLabel
- QSpacerItem * horizontalSpacer_9
- QComboBox * ctrlBox
- QComboBox * shiftBox
- QLabel * winLabel
- QComboBox * winBox
- QComboBox * altBox
- QLabel * altLabel
- QFrame * line
- QSpacerItem * verticalSpacer_3
- QSpacerItem * verticalSpacer_5
- QFrame * line_3
- QSpacerItem * verticalSpacer_4
- QSpacerItem * verticalSpacer_2
- QLabel * versionLabel
- QHBoxLayout * horizontalLayout
- QPushButton * extraButton
- QSpacerItem * horizontalSpacer_2
- QSpacerItem * verticalSpacer_7

12.83.1 Detailed Description

Definition at line 387 of file ui_settingswidget.h.

12.83.2 Member Function Documentation

12.83.2.1 void Ui_SettingsWidget::retranslateUi (QWidget * *SettingsWidget*) [inline],[inherited]

Definition at line 356 of file ui_settingswidget.h.

References Ui_SettingsWidget::altLabel, Ui_SettingsWidget::autoFWBox, Ui_SettingsWidget::capsLabel, Ui_SettingsWidget::ctrlLabel, Ui_SettingsWidget::devicesLabel, Ui_SettingsWidget::extraButton, Ui_SettingsWidget::label, Ui_SettingsWidget::label_2, Ui_SettingsWidget::label_3, Ui_SettingsWidget::label_5, Ui_SettingsWidget::label_6, Ui_SettingsWidget::label_8, Ui_SettingsWidget::loginItemBox, Ui_SettingsWidget::pushButton, Ui_SettingsWidget::shiftLabel, Ui_SettingsWidget::versionLabel, and Ui_SettingsWidget::winLabel.

Referenced by Ui_SettingsWidget::setupUi().

```
357     {
358         SettingsWidget->setWindowTitle(QApplication::translate("SettingsWidget", "Form", Q_NULLPTR));
359         label_2->setText(QApplication::translate("SettingsWidget", "\302\251 2014-2016 <a href=\"
      https://github.com/ccMSC/\" style=\"text-decoration:none;\">ccMSC</a>.<br/>\302\251 2017 <a href=\"
      https://github.com/mattanger/ckb-next/graphs/contributors\" style=\"text-decoration:none;\">The ckb-next development
       team</a>.", Q_NULLPTR));
360         label->setText(QApplication::translate("SettingsWidget", "Modifier keys", Q_NULLPTR));
361         label_6->setText(QApplication::translate("SettingsWidget", "Application", Q_NULLPTR));
362         devicesLabel->setText(QApplication::translate("SettingsWidget", "No devices connected",
      Q_NULLPTR));
363         label_5->setText(QApplication::translate("SettingsWidget", "Keyboard layout:", Q_NULLPTR));
364         label_3->setText(QApplication::translate("SettingsWidget", "<a href=\"
      https://github.com/mattanger/ckb-next\" style=\"text-decoration:none;\">https://github.com/mattanger/ckb-next</a>", Q_NU
365         pushButton->setText(QApplication::translate("SettingsWidget", "Quit", Q_NULLPTR));
366 #ifndef QT_NO_TOOLTIP
367         loginItemBox->setToolTip(QApplication::translate("SettingsWidget", "ckb-next will be
      started when you log in to your computer.", Q_NULLPTR));
368 #endif // QT_NO_TOOLTIP
369         loginItemBox->setText(QApplication::translate("SettingsWidget", "Start ckb-next at
      login", Q_NULLPTR));
370 #ifndef QT_NO_TOOLTIP
371         autoFWBox->setToolTip(QApplication::translate("SettingsWidget", "You will be notified when
      new firmware versions are available. You'll have the option to install them immediately or wait until
      later.", Q_NULLPTR));
```

```
372 #endif // QT_NO_TOOLTIP
373         autoFWBox->setText(QApplication::translate("SettingsWidget", "Check for new firmware
      automatically", Q_NULLPTR));
374         label_8->setText(QApplication::translate("SettingsWidget", "These will override the keyboard
      profile. See \"Binding\" tab for more settings.", Q_NULLPTR));
375         shiftLabel->setText(QApplication::translate("SettingsWidget", "Shift:", Q_NULLPTR));
376         capsLabel->setText(QApplication::translate("SettingsWidget", "Caps Lock:", Q_NULLPTR));
377         ctrlLabel->setText(QApplication::translate("SettingsWidget", "Ctrl:", Q_NULLPTR));
378         winLabel->setText(QApplication::translate("SettingsWidget", "Windows:", Q_NULLPTR));
379         altLabel->setText(QApplication::translate("SettingsWidget", "Alt:", Q_NULLPTR));
380         versionLabel->setText(QApplication::translate("SettingsWidget", "ckb-next", Q_NULLPTR))
    ;
381         extraButton->setText(QApplication::translate("SettingsWidget", "More settings",
     Q_NULLPTR));
382     } // retranslateUi
```

Here is the caller graph for this function:

12.83.2.2 void Ui_SettingsWidget::setupUi (QWidget * *SettingsWidget*) [inline],[inherited]

Definition at line 78 of file ui_settingswidget.h.

References Ui_SettingsWidget::altBox, Ui_SettingsWidget::altLabel, Ui_SettingsWidget::autoFWBox, Ui_SettingsWidget::capsBox, Ui_SettingsWidget::capsLabel, Ui_SettingsWidget::ctrlBox, Ui_SettingsWidget::ctrlLabel, Ui_SettingsWidget::devicesLabel, Ui_SettingsWidget::extraButton, Ui_SettingsWidget::gridLayout, Ui_SettingsWidget::gridLayout_3, Ui_SettingsWidget::gridLayout_5, Ui_SettingsWidget::horizontalLayout, Ui_SettingsWidget::horizontalLayout_2, Ui_SettingsWidget::horizontalSpacer, Ui_SettingsWidget::horizontalSpacer_10, Ui_SettingsWidget::horizontalSpacer_2, Ui_SettingsWidget::horizontalSpacer_7, Ui_SettingsWidget::horizontalSpacer_9, Ui_SettingsWidget::label, Ui_SettingsWidget::label_2, Ui_SettingsWidget::label_3, Ui_SettingsWidget::label_5, Ui_SettingsWidget::label_6, Ui_SettingsWidget::label_8, Ui_SettingsWidget::layoutBox, Ui_SettingsWidget::line, Ui_SettingsWidget::line_2, Ui_SettingsWidget::line_3, Ui_SettingsWidget::loginItemBox, Ui_SettingsWidget::pushButton, Ui_SettingsWidget::retranslateUi(), Ui_SettingsWidget::shiftBox, Ui_SettingsWidget::shiftLabel, Ui_SettingsWidget::versionLabel, Ui_SettingsWidget::verticalSpacer, Ui_SettingsWidget::verticalSpacer_10, Ui_SettingsWidget::verticalSpacer_2, Ui_SettingsWidget::verticalSpacer_3, Ui_SettingsWidget::verticalSpacer_4, Ui_SettingsWidget::verticalSpacer_5, Ui_SettingsWidget::verticalSpacer_6, Ui_SettingsWidget::verticalSpacer_7, Ui_SettingsWidget::verticalSpacer_8, Ui_SettingsWidget::winBox, and Ui_SettingsWidget::winLabel.

Referenced by SettingsWidget::SettingsWidget().

```
79     {
80         if (SettingsWidget->objectName().isEmpty())
81             SettingsWidget->setObjectName(QStringLiteral("SettingsWidget"));
82         SettingsWidget->resize(827, 654);
83         gridLayout = new QGridLayout(SettingsWidget);
84         gridLayout->setObjectName(QStringLiteral("gridLayout"));
85         label_2 = new QLabel(SettingsWidget);
86         label_2->setObjectName(QStringLiteral("label_2"));
87         label_2->setAlignment(Qt::AlignLeading|Qt::AlignLeft|Qt::AlignVCenter);
88         label_2->setOpenExternalLinks(true);
89
90         gridLayout->addWidget(label_2, 20, 0, 1, 2);
91
92         label = new QLabel(SettingsWidget);
93         label->setObjectName(QStringLiteral("label"));
94         QFont font;
95         font.setBold(true);
96         font.setWeight(75);
97         label->setFont(font);
98
99         gridLayout->addWidget(label, 6, 0, 1, 2);
100
```

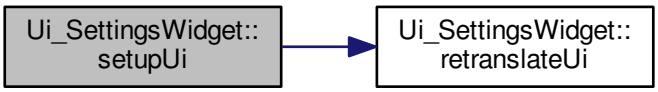
```
101         label_6 = new QLabel(SettingsWidget);
102         label_6->setObjectName(QStringLiteral("label_6"));
103         label_6->setFont(font);
104 
105         gridLayout->addWidget(label_6, 12, 0, 1, 2);
106 
107         gridLayout_5 = new QGridLayout();
108         gridLayout_5->setObjectName(QStringLiteral("gridLayout_5"));
109         horizontalSpacer_7 = new QSpacerItem(40, 20, QSizePolicy::Expanding,
    QSizePolicy::Minimum);
110 
111         gridLayout_5->addItem(horizontalSpacer_7, 2, 2, 1, 1);
112 
113         devicesLabel = new QLabel(SettingsWidget);
114         devicesLabel->setObjectName(QStringLiteral("devicesLabel"));
115 
116         gridLayout_5->addWidget(devicesLabel, 0, 0, 1, 3);
117 
118         label_5 = new QLabel(SettingsWidget);
119         label_5->setObjectName(QStringLiteral("label_5"));
120 
121         gridLayout_5->addWidget(label_5, 2, 0, 1, 1);
122 
123         layoutBox = new QComboBox(SettingsWidget);
124         layoutBox->setObjectName(QStringLiteral("layoutBox"));
125 
126         gridLayout_5->addWidget(layoutBox, 2, 1, 1, 1);
127 
128         verticalSpacer_10 = new QSpacerItem(20, 6, QSizePolicy::Minimum,
    QSizePolicy::Fixed);
129 
130         gridLayout_5->addItem(verticalSpacer_10, 1, 0, 1, 1);
131 
132 
133         gridLayout->addLayout(gridLayout_5, 4, 1, 1, 1);
134 
135         horizontalLayout_2 = new QHBoxLayout();
136         horizontalLayout_2->setObjectName(QStringLiteral("horizontalLayout_2"));
137         label_3 = new QLabel(SettingsWidget);
138         label_3->setObjectName(QStringLiteral("label_3"));
139         label_3->setOpenExternalLinks(true);
140 
141         horizontalLayout_2->addWidget(label_3);
142 
143         horizontalSpacer = new QSpacerItem(40, 20, QSizePolicy::Expanding,
    QSizePolicy::Minimum);
144 
145         horizontalLayout_2->addItem(horizontalSpacer);
146 
147         pushButton = new QPushButton(SettingsWidget);
148         pushButton->setObjectName(QStringLiteral("pushButton"));
149         pushButton->setFlat(false);
150 
151         horizontalLayout_2->addWidget(pushButton);
152 
153 
154         gridLayout->addLayout(horizontalLayout_2, 21, 0, 1, 2);
155 
156         verticalSpacer = new QSpacerItem(20, 40, QSizePolicy::Minimum, QSizePolicy::Expanding
    );
157 
158         gridLayout->addItem(verticalSpacer, 19, 0, 1, 2);
159 
160         verticalSpacer_8 = new QSpacerItem(0, 28, QSizePolicy::Minimum, QSizePolicy::Fixed)
    ;
161 
162         gridLayout->addItem(verticalSpacer_8, 16, 0, 1, 1);
163 
164         loginItemBox = new QCheckBox(SettingsWidget);
165         loginItemBox->setObjectName(QStringLiteral("loginItemBox"));
166 
167         gridLayout->addWidget(loginItemBox, 16, 1, 1, 1);
168 
169         verticalSpacer_6 = new QSpacerItem(0, 28, QSizePolicy::Minimum, QSizePolicy::Fixed)
    ;
170 
171         gridLayout->addItem(verticalSpacer_6, 15, 0, 1, 1);
172 
173         autoFWBox = new QCheckBox(SettingsWidget);
174         autoFWBox->setObjectName(QStringLiteral("autoFWBox"));
175         autoFWBox->setChecked(true);
176 
177         gridLayout->addWidget(autoFWBox, 15, 1, 1, 1);
178 
179         line_2 = new QFrame(SettingsWidget);
180         line_2->setObjectName(QStringLiteral("line_2"));
181         line_2->setFrameShape(QFrame::HLine);
```

```
182         line_2->setFrameShadow(QFrame::Sunken);
183 
184         gridLayout->addWidget(line_2, 13, 0, 1, 2);
185 
186         label_8 = new QLabel(SettingsWidget);
187         label_8->setObjectName(QStringLiteral("label_8"));
188         QFont font1;
189         font1.setItalic(true);
190         label_8->setFont(font1);
191 
192         gridLayout->addWidget(label_8, 8, 1, 1, 1);
193 
194         gridLayout_3 = new QGridLayout();
195         gridLayout_3->setObjectName(QStringLiteral("gridLayout_3"));
196         shiftLabel = new QLabel(SettingsWidget);
197         shiftLabel->setObjectName(QStringLiteral("shiftLabel"));
198         QSizePolicy sizePolicy(QSizePolicy::Expanding, QSizePolicy::Preferred);
199         sizePolicy.setHorizontalStretch(0);
200         sizePolicy.setVerticalStretch(0);
201         sizePolicy.setHeightForWidth(shiftLabel->sizePolicy().hasHeightForWidth());
202         shiftLabel->setSizePolicy(sizePolicy);
203         shiftLabel->setAlignment(Qt::AlignRight|Qt::AlignTrailing|Qt::AlignVCenter);
204 
205         gridLayout_3->addWidget(shiftLabel, 0, 3, 1, 1);
206 
207         capsLabel = new QLabel(SettingsWidget);
208         capsLabel->setObjectName(QStringLiteral("capsLabel"));
209         capsLabel->setAlignment(Qt::AlignRight|Qt::AlignTrailing|Qt::AlignVCenter);
210 
211         gridLayout_3->addWidget(capsLabel, 0, 1, 1, 1);
212 
213         horizontalSpacer_10 = new QSpacerItem(0, 0, QSizePolicy::Fixed,
    QSizePolicy::Minimum);
214 
215         gridLayout_3->addItem(horizontalSpacer_10, 0, 0, 1, 1);
216 
217         capsBox = new QComboBox(SettingsWidget);
218         capsBox->setObjectName(QStringLiteral("capsBox"));
219         QSizePolicy sizePolicy1(QSizePolicy::Expanding, QSizePolicy::Fixed);
220         sizePolicy1.setHorizontalStretch(0);
221         sizePolicy1.setVerticalStretch(0);
222         sizePolicy1.setHeightForWidth(capsBox->sizePolicy().hasHeightForWidth());
223         capsBox->setSizePolicy(sizePolicy1);
224 
225         gridLayout_3->addWidget(capsBox, 0, 2, 1, 1);
226 
227         ctrlLabel = new QLabel(SettingsWidget);
228         ctrlLabel->setObjectName(QStringLiteral("ctrlLabel"));
229         ctrlLabel->setAlignment(Qt::AlignRight|Qt::AlignTrailing|Qt::AlignVCenter);
230 
231         gridLayout_3->addWidget(ctrlLabel, 1, 1, 1, 1);
232 
233         horizontalSpacer_9 = new QSpacerItem(40, 20, QSizePolicy::Expanding,
    QSizePolicy::Minimum);
234 
235         gridLayout_3->addItem(horizontalSpacer_9, 1, 7, 1, 1);
236 
237         ctrlBox = new QComboBox(SettingsWidget);
238         ctrlBox->setObjectName(QStringLiteral("ctrlBox"));
239 
240         gridLayout_3->addWidget(ctrlBox, 1, 2, 1, 1);
241 
242         shiftBox = new QComboBox(SettingsWidget);
243         shiftBox->setObjectName(QStringLiteral("shiftBox"));
244         sizePolicy1.setHeightForWidth(shiftBox->sizePolicy().hasHeightForWidth());
245         shiftBox->setSizePolicy(sizePolicy1);
246 
247         gridLayout_3->addWidget(shiftBox, 0, 4, 1, 1);
248 
249         winLabel = new QLabel(SettingsWidget);
250         winLabel->setObjectName(QStringLiteral("winLabel"));
251         sizePolicy.setHeightForWidth(winLabel->sizePolicy().hasHeightForWidth());
252         winLabel->setSizePolicy(sizePolicy);
253         winLabel->setAlignment(Qt::AlignRight|Qt::AlignTrailing|Qt::AlignVCenter);
254 
255         gridLayout_3->addWidget(winLabel, 1, 3, 1, 1);
256 
257         winBox = new QComboBox(SettingsWidget);
258         winBox->setObjectName(QStringLiteral("winBox"));
259         sizePolicy1.setHeightForWidth(winBox->sizePolicy().hasHeightForWidth());
260         winBox->setSizePolicy(sizePolicy1);
261 
262         gridLayout_3->addWidget(winBox, 1, 4, 1, 1);
263 
264         altBox = new QComboBox(SettingsWidget);
265         altBox->setObjectName(QStringLiteral("altBox"));
266         sizePolicy1.setHeightForWidth(altBox->sizePolicy().hasHeightForWidth());
```

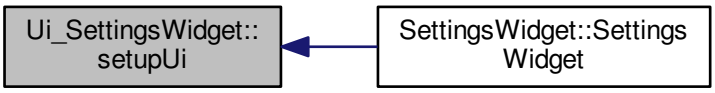
```
267         altBox->setSizePolicy(sizePolicy1);
268
269         gridLayout_3->addWidget(altBox, 1, 6, 1, 1);
270
271         altLabel = new QLabel(SettingsWidget);
272         altLabel->setObjectName(QStringLiteral("altLabel"));
273         sizePolicy.setHeightForWidth(altLabel->sizePolicy().hasHeightForWidth());
274         altLabel->setSizePolicy(sizePolicy);
275         altLabel->setAlignment(Qt::AlignRight|Qt::AlignTrailing|Qt::AlignVCenter);
276
277         gridLayout_3->addWidget(altLabel, 1, 5, 1, 1);
278
279
280         gridLayout->addLayout(gridLayout_3, 10, 1, 1, 1);
281
282         line = new QFrame(SettingsWidget);
283         line->setObjectName(QStringLiteral("line"));
284         line->setFrameShape(QFrame::HLine);
285         line->setFrameShadow(QFrame::Sunken);
286
287         gridLayout->addWidget(line, 1, 0, 1, 2);
288
289         verticalSpacer_3 = new QSpacerItem(0, 5, QSizePolicy::Minimum, QSizePolicy::Fixed);
290
291         gridLayout->addItem(verticalSpacer_3, 2, 1, 1, 1);
292
293         verticalSpacer_5 = new QSpacerItem(20, 12, QSizePolicy::Minimum, QSizePolicy::Fixed
    );
294
295         gridLayout->addItem(verticalSpacer_5, 11, 1, 1, 1);
296
297         line_3 = new QFrame(SettingsWidget);
298         line_3->setObjectName(QStringLiteral("line_3"));
299         QSizePolicy sizePolicy2(QSizePolicy::Minimum, QSizePolicy::Fixed);
300         sizePolicy2.setHorizontalStretch(0);
301         sizePolicy2.setVerticalStretch(0);
302         sizePolicy2.setHeightForWidth(line_3->sizePolicy().hasHeightForWidth());
303         line_3->setSizePolicy(sizePolicy2);
304         line_3->setFrameShape(QFrame::HLine);
305         line_3->setFrameShadow(QFrame::Sunken);
306
307         gridLayout->addWidget(line_3, 7, 0, 1, 2);
308
309         verticalSpacer_4 = new QSpacerItem(20, 6, QSizePolicy::Minimum, QSizePolicy::Fixed)
    ;
310
311         gridLayout->addItem(verticalSpacer_4, 9, 1, 1, 1);
312
313         verticalSpacer_2 = new QSpacerItem(20, 12, QSizePolicy::Minimum, QSizePolicy::Fixed
    );
314
315         gridLayout->addItem(verticalSpacer_2, 5, 1, 1, 1);
316
317         versionLabel = new QLabel(SettingsWidget);
318         versionLabel->setObjectName(QStringLiteral("versionLabel"));
319         versionLabel->setFont(font);
320
321         gridLayout->addWidget(versionLabel, 0, 0, 1, 2);
322
323         horizontalLayout = new QHBoxLayout();
324         horizontalLayout->setObjectName(QStringLiteral("horizontalLayout"));
325         extraButton = new QPushButton(SettingsWidget);
326         extraButton->setObjectName(QStringLiteral("extraButton"));
327
328         horizontalLayout->addWidget(extraButton);
329
330         horizontalSpacer_2 = new QSpacerItem(40, 20, QSizePolicy::Expanding,
    QSizePolicy::Minimum);
331
332         horizontalLayout->addItem(horizontalSpacer_2);
333
334
335         gridLayout->addLayout(horizontalLayout, 17, 1, 1, 1);
336
337         verticalSpacer_7 = new QSpacerItem(0, 28, QSizePolicy::Minimum, QSizePolicy::Fixed)
    ;
338
339         gridLayout->addItem(verticalSpacer_7, 17, 0, 1, 1);
340
341         QWidget::setTabOrder(layoutBox, capsBox);
342         QWidget::setTabOrder(capsBox, shiftBox);
343         QWidget::setTabOrder(shiftBox, ctrlBox);
344         QWidget::setTabOrder(ctrlBox, winBox);
345         QWidget::setTabOrder(winBox, altBox);
346         QWidget::setTabOrder(altBox, autoFWBox);
347         QWidget::setTabOrder(autoFWBox, loginItemBox);
348         QWidget::setTabOrder(loginItemBox, extraButton);
```

```
349         QWidget::setTabOrder(extraButton, pushButton);
350
351         retranslateUi(SettingsWidget);
352
353         QMetaObject::connectSlotsByName(SettingsWidget);
354     } // setupUi
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.83.3 Field Documentation

12.83.3.1 QComboBox* Ui_SettingsWidget::altBox [inherited]

Definition at line 64 of file ui_settingswidget.h.

Referenced by SettingsWidget::SettingsWidget(), Ui_SettingsWidget::setupUi(), and SettingsWidget::updateModifiers().

12.83.3.2 QLabel* Ui_SettingsWidget::altLabel [inherited]

Definition at line 65 of file ui_settingswidget.h.

Referenced by Ui_SettingsWidget::retranslateUi(), and Ui_SettingsWidget::setupUi().

12.83.3.3 QCheckBox* Ui_SettingsWidget::autoFWBox [inherited]

Definition at line 50 of file ui_settingswidget.h.

Referenced by Ui_SettingsWidget::retranslateUi(), SettingsWidget::SettingsWidget(), and Ui_SettingsWidget::setupUi().

12.83.3.4 QComboBox* Ui_SettingsWidget::capsBox [inherited]

Definition at line 57 of file ui_settingswidget.h.

Referenced by SettingsWidget::SettingsWidget(), Ui_SettingsWidget::setupUi(), and SettingsWidget::updateModifiers().

12.83.3.5 QLabel* Ui_SettingsWidget::capsLabel [inherited]

Definition at line 55 of file ui_settingswidget.h.

Referenced by Ui_SettingsWidget::retranslateUi(), and Ui_SettingsWidget::setupUi().

12.83.3.6 QComboBox* Ui_SettingsWidget::ctrlBox [inherited]

Definition at line 60 of file ui_settingswidget.h.

Referenced by SettingsWidget::SettingsWidget(), Ui_SettingsWidget::setupUi(), and SettingsWidget::updateModifiers().

12.83.3.7 QLabel* Ui_SettingsWidget::ctrlLabel [inherited]

Definition at line 58 of file ui_settingswidget.h.

Referenced by Ui_SettingsWidget::retranslateUi(), and Ui_SettingsWidget::setupUi().

12.83.3.8 QLabel* Ui_SettingsWidget::devicesLabel [inherited]

Definition at line 38 of file ui_settingswidget.h.

Referenced by Ui_SettingsWidget::retranslateUi(), SettingsWidget::setStatus(), and Ui_SettingsWidget::setupUi().

12.83.3.9 QPushButton* Ui_SettingsWidget::extraButton [inherited]

Definition at line 74 of file ui_settingswidget.h.

Referenced by Ui_SettingsWidget::retranslateUi(), and Ui_SettingsWidget::setupUi().

12.83.3.10 QGridLayout* Ui_SettingsWidget::gridLayout [inherited]

Definition at line 32 of file ui_settingswidget.h.

Referenced by Ui_SettingsWidget::setupUi().

12.83.3.11 QGridLayout* Ui_SettingsWidget::gridLayout_3 [inherited]

Definition at line 53 of file ui_settingswidget.h.

Referenced by Ui_SettingsWidget::setupUi().

12.83.3.12 QGridLayout* Ui_SettingsWidget::gridLayout_5 [inherited]

Definition at line 36 of file ui_settingswidget.h.

Referenced by Ui_SettingsWidget::setupUi().

12.83.3.13 QHBoxLayout* Ui_SettingsWidget::horizontalLayout [inherited]

Definition at line 73 of file ui_settingswidget.h.

Referenced by Ui_SettingsWidget::setupUi().

12.83.3.14 QHBoxLayout* Ui_SettingsWidget::horizontalLayout_2 [inherited]

Definition at line 42 of file ui_settingswidget.h.

Referenced by Ui_SettingsWidget::setupUi().

12.83.3.15 QSpacerItem* Ui_SettingsWidget::horizontalSpacer [inherited]

Definition at line 44 of file ui_settingswidget.h.

Referenced by Ui_SettingsWidget::setupUi().

12.83.3.16 QSpacerItem* Ui_SettingsWidget::horizontalSpacer_10 [inherited]

Definition at line 56 of file ui_settingswidget.h.

Referenced by Ui_SettingsWidget::setupUi().

12.83.3.17 QSpacerItem* Ui_SettingsWidget::horizontalSpacer_2 [inherited]

Definition at line 75 of file ui_settingswidget.h.

Referenced by Ui_SettingsWidget::setupUi().

12.83.3.18 QSpacerItem* Ui_SettingsWidget::horizontalSpacer_7 [inherited]

Definition at line 37 of file ui_settingswidget.h.

Referenced by Ui_SettingsWidget::setupUi().

12.83.3.19 QSpacerItem* Ui_SettingsWidget::horizontalSpacer_9 [inherited]

Definition at line 59 of file ui_settingswidget.h.

Referenced by Ui_SettingsWidget::setupUi().

12.83.3.20 QLabel* Ui_SettingsWidget::label [inherited]

Definition at line 34 of file ui_settingswidget.h.

Referenced by Ui_SettingsWidget::retranslateUi(), and Ui_SettingsWidget::setupUi().

12.83.3.21 QLabel* Ui_SettingsWidget::label_2 [inherited]

Definition at line 33 of file ui_settingswidget.h.

Referenced by Ui_SettingsWidget::retranslateUi(), and Ui_SettingsWidget::setupUi().

12.83.3.22 QLabel* Ui_SettingsWidget::label_3 [inherited]

Definition at line 43 of file ui_settingswidget.h.

Referenced by Ui_SettingsWidget::retranslateUi(), and Ui_SettingsWidget::setupUi().

12.83.3.23 QLabel* Ui_SettingsWidget::label_5 [inherited]

Definition at line 39 of file ui_settingswidget.h.

Referenced by Ui_SettingsWidget::retranslateUi(), and Ui_SettingsWidget::setupUi().

12.83.3.24 QLabel* Ui_SettingsWidget::label_6 [inherited]

Definition at line 35 of file ui_settingswidget.h.

Referenced by Ui_SettingsWidget::retranslateUi(), and Ui_SettingsWidget::setupUi().

12.83.3.25 QLabel* Ui_SettingsWidget::label_8 [inherited]

Definition at line 52 of file ui_settingswidget.h.

Referenced by Ui_SettingsWidget::retranslateUi(), and Ui_SettingsWidget::setupUi().

12.83.3.26 QComboBox* Ui_SettingsWidget::layoutBox [inherited]

Definition at line 40 of file ui_settingswidget.h.

Referenced by SettingsWidget::SettingsWidget(), Ui_SettingsWidget::setupUi(), and SettingsWidget::showLayout-Dialog().

12.83.3.27 QFrame* Ui_SettingsWidget::line [inherited]

Definition at line 66 of file ui_settingswidget.h.

Referenced by Ui_SettingsWidget::setupUi().

12.83.3.28 QFrame* Ui_SettingsWidget::line_2 [inherited]

Definition at line 51 of file ui_settingswidget.h.

Referenced by Ui_SettingsWidget::setupUi().

12.83.3.29 QFrame* Ui_SettingsWidget::line_3 [inherited]

Definition at line 69 of file ui_settingswidget.h.

Referenced by Ui_SettingsWidget::setupUi().

12.83.3.30 QCheckBox* Ui_SettingsWidget::loginItemBox [inherited]

Definition at line 48 of file ui_settingswidget.h.

Referenced by Ui_SettingsWidget::retranslateUi(), SettingsWidget::SettingsWidget(), and Ui_SettingsWidget-::setupUi().

12.83.3.31 QPushButton* Ui_SettingsWidget::pushButton [inherited]

Definition at line 45 of file ui_settingswidget.h.

Referenced by Ui_SettingsWidget::retranslateUi(), and Ui_SettingsWidget::setupUi().

12.83.3.32 QComboBox* Ui_SettingsWidget::shiftBox [inherited]

Definition at line 61 of file ui_settingswidget.h.

Referenced by SettingsWidget::SettingsWidget(), Ui_SettingsWidget::setupUi(), and SettingsWidget::updateModifiers().

12.83.3.33 QLabel* Ui_SettingsWidget::shiftLabel [inherited]

Definition at line 54 of file ui_settingswidget.h.

Referenced by Ui_SettingsWidget::retranslateUi(), and Ui_SettingsWidget::setupUi().

12.83.3.34 QLabel* Ui_SettingsWidget::versionLabel [inherited]

Definition at line 72 of file ui_settingswidget.h.

Referenced by Ui_SettingsWidget::retranslateUi(), Ui_SettingsWidget::setupUi(), and SettingsWidget::setVersion().

12.83.3.35 QSpacerItem* Ui_SettingsWidget::verticalSpacer [inherited]

Definition at line 46 of file ui_settingswidget.h.

Referenced by Ui_SettingsWidget::setupUi().

12.83.3.36 QSpacerItem* Ui_SettingsWidget::verticalSpacer_10 [inherited]

Definition at line 41 of file ui_settingswidget.h.

Referenced by Ui_SettingsWidget::setupUi().

12.83.3.37 QSpacerItem* Ui_SettingsWidget::verticalSpacer_2 [inherited]

Definition at line 71 of file ui_settingswidget.h.

Referenced by Ui_SettingsWidget::setupUi().

12.83.3.38 QSpacerItem* Ui_SettingsWidget::verticalSpacer_3 [inherited]

Definition at line 67 of file ui_settingswidget.h.

Referenced by Ui_SettingsWidget::setupUi().

12.83.3.39 QSpacerItem* Ui_SettingsWidget::verticalSpacer_4 [inherited]

Definition at line 70 of file ui_settingswidget.h.

Referenced by Ui_SettingsWidget::setupUi().

12.83.3.40 QSpacerItem* Ui_SettingsWidget::verticalSpacer_5 [inherited]

Definition at line 68 of file ui_settingswidget.h.

Referenced by Ui_SettingsWidget::setupUi().

12.83.3.41 QSpacerItem* Ui_SettingsWidget::verticalSpacer_6 [inherited]

Definition at line 49 of file ui_settingswidget.h.

Referenced by Ui_SettingsWidget::setupUi().

12.83.3.42 QSpacerItem* Ui_SettingsWidget::verticalSpacer_7 [inherited]

Definition at line 76 of file ui_settingswidget.h.

Referenced by Ui_SettingsWidget::setupUi().

12.83.3.43 QSpacerItem* Ui_SettingsWidget::verticalSpacer_8 [inherited]

Definition at line 47 of file ui_settingswidget.h.

Referenced by Ui_SettingsWidget::setupUi().

12.83.3.44 QComboBox* Ui_SettingsWidget::winBox [inherited]

Definition at line 63 of file ui_settingswidget.h.

Referenced by SettingsWidget::SettingsWidget(), Ui_SettingsWidget::setupUi(), and SettingsWidget::updateModifiers().

12.83.3.45 QLabel* Ui_SettingsWidget::winLabel [inherited]

Definition at line 62 of file ui_settingswidget.h.

Referenced by Ui_SettingsWidget::retranslateUi(), and Ui_SettingsWidget::setupUi().

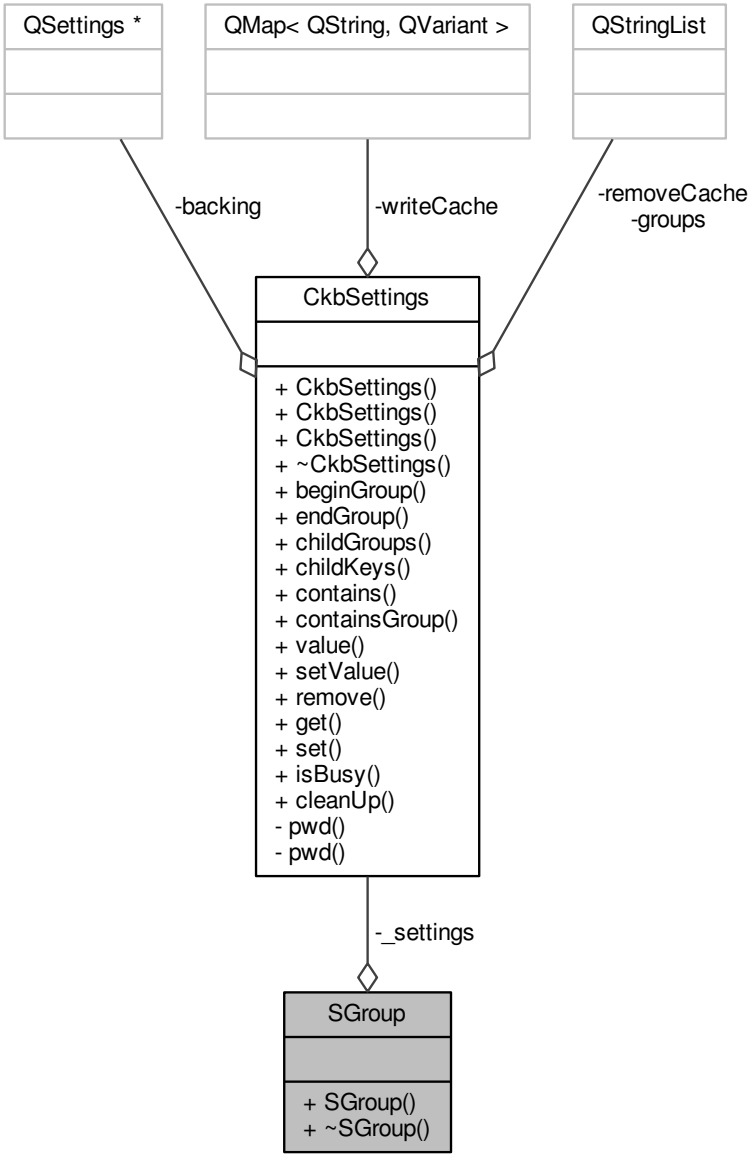
The documentation for this class was generated from the following file:

- src/ckb/ui_settingswidget.h

12.84 SGroup Class Reference

`#include <src/ckb/ckbsettings.h>`

Collaboration diagram for SGroup:



Public Member Functions

- SGroup (CkbSettings &settings, const QString &prefix)
- ~SGroup ()

Private Attributes

- CkbSettings & _settings

12.84.1 Detailed Description

Definition at line 57 of file ckbsettings.h.

12.84.2 Constructor & Destructor Documentation

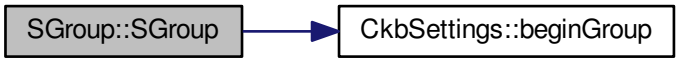
12.84.2.1 SGroup::SGroup (CkbSettings & *settings*, const QString & *prefix*) [inline]

Definition at line 60 of file ckbsettings.h.

References CkbSettings::beginGroup().

```
60 : _settings(settings) { settings.beginGroup(prefix); }
```

Here is the call graph for this function:



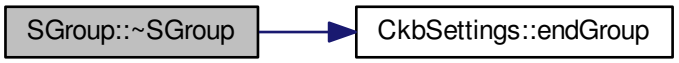
12.84.2.2 SGroup::~SGroup ( ) [inline]

Definition at line 61 of file ckbsettings.h.

References _settings, and CkbSettings::endGroup().

```
61 { _settings.endGroup(); }
```

Here is the call graph for this function:



12.84.3 Field Documentation

12.84.3.1 CkbSettings& SGroup::_settings [private]

Definition at line 64 of file ckbsettings.h.

Referenced by ~SGroup().

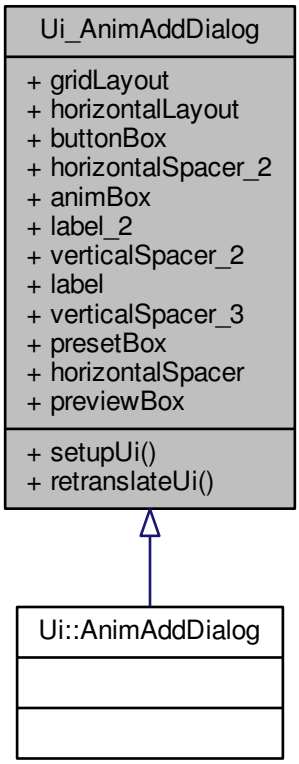
The documentation for this class was generated from the following file:

- src/ckb/ckbsettings.h

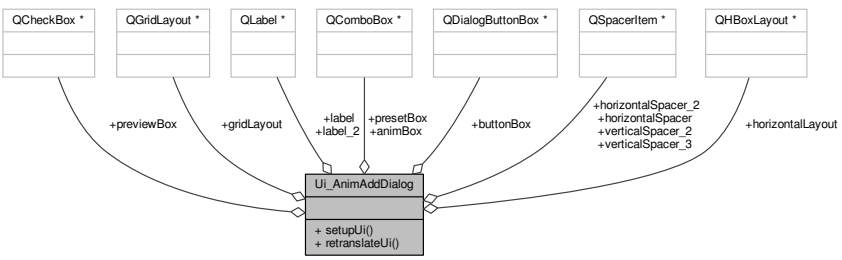
12.85 Ui_AnimAddDialog Class Reference

`#include <src/ckb/ui_animadddialog.h>`

Inheritance diagram for Ui_AnimAddDialog:



Collaboration diagram for Ui_AnimAddDialog:



Public Member Functions

- void setupUi (QDialog *AnimAddDialog)
- void retranslateUi (QDialog *AnimAddDialog)

Data Fields

- QGridLayout * gridLayout
- QHBoxLayout * horizontalLayout
- QDialogButtonBox * buttonBox
- QSpacerItem * horizontalSpacer_2
- QComboBox * animBox
- QLabel * label_2
- QSpacerItem * verticalSpacer_2
- QLabel * label
- QSpacerItem * verticalSpacer_3
- QComboBox * presetBox
- QSpacerItem * horizontalSpacer
- QCheckBox * previewBox

12.85.1 Detailed Description

Definition at line 28 of file ui_animadddialog.h.

12.85.2 Member Function Documentation

12.85.2.1 void Ui_AnimAddDialog::retranslateUi (QDialog * *AnimAddDialog*) [inline]

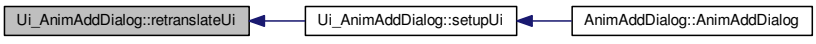
Definition at line 135 of file ui_animadddialog.h.

References label, label_2, and previewBox.

Referenced by setupUi().

```
136     {
137         AnimAddDialog->setWindowTitle(QApplication::translate("AnimAddDialog", "Add Animation", Q_NULLPTR))
    ;
138         label_2->setText(QApplication::translate("AnimAddDialog", "Preset:", Q_NULLPTR));
139         label->setText(QApplication::translate("AnimAddDialog", "Animation:", Q_NULLPTR));
140         previewBox->setText(QApplication::translate("AnimAddDialog", "Show preview", Q_NULLPTR));
141     } // retranslateUi
```

Here is the caller graph for this function:



12.85.2.2 void Ui_AnimAddDialog::setupUi (QDialog * *AnimAddDialog*) [inline]

Definition at line 44 of file ui_animadddialog.h.

References animBox, buttonBox, gridLayout, horizontalLayout, horizontalSpacer, horizontalSpacer_2, label, label_2, presetBox, previewBox, retranslateUi(), verticalSpacer_2, and verticalSpacer_3.

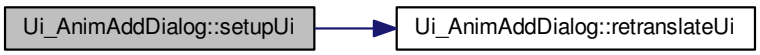
Referenced by AnimAddDialog::AnimAddDialog().

```
45     {
46         if (AnimAddDialog->objectName().isEmpty())
47             AnimAddDialog->setObjectName(QStringLiteral("AnimAddDialog"));
48         AnimAddDialog->resize(350, 205);
```

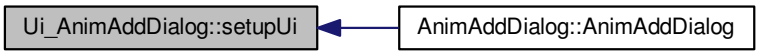
```
49          QSizePolicy sizePolicy(QSizePolicy::Fixed, QSizePolicy::Fixed);
50          sizePolicy.setHorizontalStretch(0);
51          sizePolicy.setVerticalStretch(0);
52          sizePolicy.setHeightForWidth(AnimAddDialog->sizePolicy().hasHeightForWidth());
53          AnimAddDialog->setSizePolicy(sizePolicy);
54          gridLayout = new QGridLayout(AnimAddDialog);
55          gridLayout->setObjectName(QStringLiteral("gridLayout"));
56          horizontalLayout = new QHBoxLayout();
57          horizontalLayout->setObjectName(QStringLiteral("horizontalLayout"));
58          buttonBox = new QDialogButtonBox(AnimAddDialog);
59          buttonBox->setObjectName(QStringLiteral("buttonBox"));
60          buttonBox->setOrientation(Qt::Horizontal);
61          buttonBox->setStandardButtons(QDialogButtonBox::Cancel|QDialogButtonBox::Ok);
62          buttonBox->setCenterButtons(false);
63
64          horizontalLayout->addWidget(buttonBox);
65
66
67          gridLayout->addLayout(horizontalLayout, 5, 0, 1, 4);
68
69          horizontalSpacer_2 = new QSpacerItem(50, 20, QSizePolicy::Fixed,
    QSizePolicy::Minimum);
70
71          gridLayout->addItem(horizontalSpacer_2, 1, 3, 1, 1);
72
73          animBox = new QComboBox(AnimAddDialog);
74          animBox->setObjectName(QStringLiteral("animBox"));
75          QSizePolicy sizePolicy1(QSizePolicy::Expanding, QSizePolicy::Fixed);
76          sizePolicy1.setHorizontalStretch(0);
77          sizePolicy1.setVerticalStretch(0);
78          sizePolicy1.setHeightForWidth(animBox->sizePolicy().hasHeightForWidth());
79          animBox->setSizePolicy(sizePolicy1);
80
81          gridLayout->addWidget(animBox, 1, 2, 1, 1);
82
83          label_2 = new QLabel(AnimAddDialog);
84          label_2->setObjectName(QStringLiteral("label_2"));
85          QSizePolicy sizePolicy2(QSizePolicy::Preferred, QSizePolicy::Preferred);
86          sizePolicy2.setHorizontalStretch(0);
87          sizePolicy2.setVerticalStretch(0);
88          sizePolicy2.setHeightForWidth(label_2->sizePolicy().hasHeightForWidth());
89          label_2->setSizePolicy(sizePolicy2);
90          label_2->setAlignment(Qt::AlignRight|Qt::AlignTrailing|Qt::AlignVCenter);
91
92          gridLayout->addWidget(label_2, 2, 1, 1, 1);
93
94          verticalSpacer_2 = new QSpacerItem(20, 0, QSizePolicy::Minimum,
    QSizePolicy::Expanding);
95
96          gridLayout->addItem(verticalSpacer_2, 4, 2, 1, 1);
97
98          label = new QLabel(AnimAddDialog);
99          label->setObjectName(QStringLiteral("label"));
100          sizePolicy2.setHeightForWidth(label->sizePolicy().hasHeightForWidth());
101          label->setSizePolicy(sizePolicy2);
102          label->setAlignment(Qt::AlignRight|Qt::AlignTrailing|Qt::AlignVCenter);
103
104          gridLayout->addWidget(label, 1, 1, 1, 1);
105
106          verticalSpacer_3 = new QSpacerItem(20, 0, QSizePolicy::Minimum,
    QSizePolicy::Expanding);
107
108          gridLayout->addItem(verticalSpacer_3, 0, 2, 1, 1);
109
110          presetBox = new QComboBox(AnimAddDialog);
111          presetBox->setObjectName(QStringLiteral("presetBox"));
112          sizePolicy1.setHeightForWidth(presetBox->sizePolicy().hasHeightForWidth());
113          presetBox->setSizePolicy(sizePolicy1);
114
115          gridLayout->addWidget(presetBox, 2, 2, 1, 1);
116
117          horizontalSpacer = new QSpacerItem(50, 20, QSizePolicy::Fixed, QSizePolicy::Minimum
    );
118
119          gridLayout->addItem(horizontalSpacer, 1, 0, 1, 1);
120
121          previewBox = new QCheckBox(AnimAddDialog);
122          previewBox->setObjectName(QStringLiteral("previewBox"));
123          previewBox->setChecked(true);
124
125          gridLayout->addWidget(previewBox, 3, 2, 1, 1);
126
127
128          retranslateUi(AnimAddDialog);
129          QObject::connect(buttonBox, SIGNAL(accepted()), AnimAddDialog, SLOT(accept()));
130          QObject::connect(buttonBox, SIGNAL(rejected()), AnimAddDialog, SLOT(reject()));
131
```

```
132         QMetaObject::connectSlotsByName(AnimAddDialog);
133     } // setupUi
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.85.3 Field Documentation

12.85.3.1 QComboBox* Ui_AnimAddDialog::animBox

Definition at line 35 of file ui_animadddialog.h.

Referenced by AnimAddDialog::AnimAddDialog(), AnimAddDialog::chosenScript(), and setupUi().

12.85.3.2 QDialogButtonBox* Ui_AnimAddDialog::buttonBox

Definition at line 33 of file ui_animadddialog.h.

Referenced by setupUi().

12.85.3.3 QGridLayout* Ui_AnimAddDialog::gridLayout

Definition at line 31 of file ui_animadddialog.h.

Referenced by setupUi().

12.85.3.4 QHBoxLayout* Ui_AnimAddDialog::horizontalLayout

Definition at line 32 of file ui_animadddialog.h.

Referenced by setupUi().

12.85.3.5 QSpacerItem* Ui_AnimAddDialog::horizontalSpacer

Definition at line 41 of file ui_animadddialog.h.

Referenced by setupUi().

12.85.3.6 QSpacerItem* Ui_AnimAddDialog::horizontalSpacer_2

Definition at line 34 of file ui_animadddialog.h.

Referenced by setupUi().

12.85.3.7 QLabel* Ui_AnimAddDialog::label

Definition at line 38 of file ui_animadddialog.h.

Referenced by retranslateUi(), and setupUi().

12.85.3.8 QLabel* Ui_AnimAddDialog::label_2

Definition at line 36 of file ui_animadddialog.h.

Referenced by retranslateUi(), and setupUi().

12.85.3.9 QComboBox* Ui_AnimAddDialog::presetBox

Definition at line 40 of file ui_animadddialog.h.

Referenced by AnimAddDialog::chosenPreset(), AnimAddDialog::on_animBox_activated(), AnimAddDialog::on_previewBox_clicked(), and setupUi().

12.85.3.10 QCheckBox* Ui_AnimAddDialog::previewBox

Definition at line 42 of file ui_animadddialog.h.

Referenced by retranslateUi(), and setupUi().

12.85.3.11 QSpacerItem* Ui_AnimAddDialog::verticalSpacer_2

Definition at line 37 of file ui_animadddialog.h.

Referenced by setupUi().

12.85.3.12 QSpacerItem* Ui_AnimAddDialog::verticalSpacer_3

Definition at line 39 of file ui_animadddialog.h.

Referenced by setupUi().

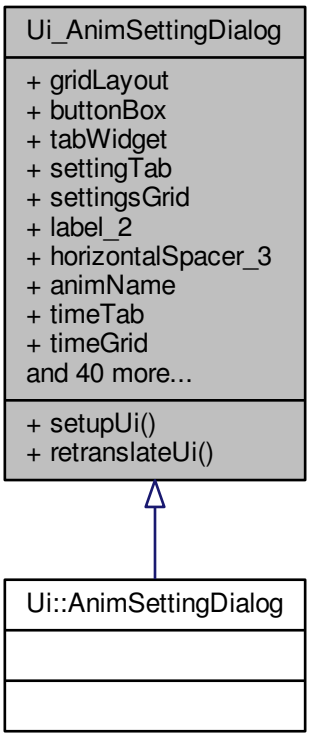
The documentation for this class was generated from the following file:

- src/ckb/ui_animadddialog.h

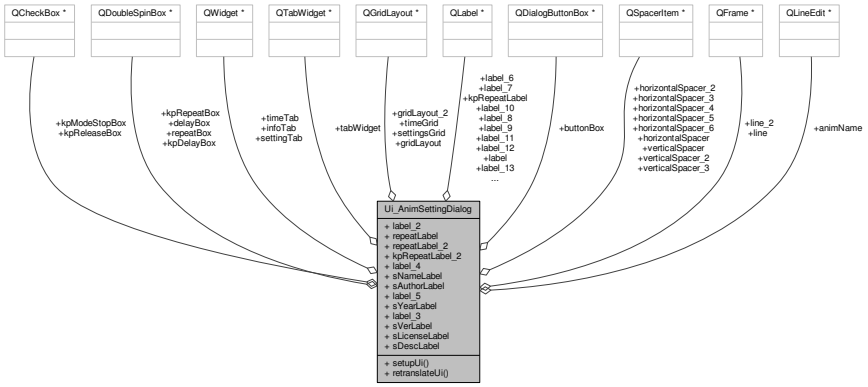
12.86 Ui_AnimSettingDialog Class Reference

`#include <src/ckb/ui_animsettingdialog.h>`

Inheritance diagram for Ui_AnimSettingDialog:



Collaboration diagram for Ui_AnimSettingDialog:



Public Member Functions

- void setupUi (QDialog *AnimSettingDialog)
- void retranslateUi (QDialog *AnimSettingDialog)

Data Fields

- QGridLayout * gridLayout
- QDialogButtonBox * buttonBox
- QTabWidget * tabWidget
- QWidget * settingTab
- QGridLayout * settingsGrid
- QLabel * label_2
- QSpacerItem * horizontalSpacer_3
- QLineEdit * animName
- QWidget * timeTab
- QGridLayout * timeGrid
- QSpacerItem * horizontalSpacer_5
- QLabel * kpRepeatLabel
- QLabel * repeatLabel
- QDoubleSpinBox * repeatBox
- QLabel * repeatLabel_2
- QFrame * line
- QLabel * kpRepeatLabel_2
- QLabel * label_4
- QLabel * label_8
- QDoubleSpinBox * kpDelayBox
- QLabel * label_13
- QLabel * label_12
- QDoubleSpinBox * delayBox
- QCheckBox * kpModeStopBox
- QLabel * label_10
- QSpacerItem * verticalSpacer_3
- QSpacerItem * horizontalSpacer_4
- QCheckBox * kpReleaseBox
- QLabel * label_6
- QDoubleSpinBox * kpRepeatBox
- QSpacerItem * verticalSpacer_2
- QFrame * line_2
- QSpacerItem * horizontalSpacer_6
- QWidget * infoTab
- QGridLayout * gridLayout_2
- QLabel * sNameLabel
- QLabel * label_7
- QLabel * label_11
- QLabel * sAuthorLabel
- QLabel * label_9
- QLabel * label_5
- QLabel * sYearLabel
- QLabel * label_3
- QSpacerItem * horizontalSpacer
- QSpacerItem * horizontalSpacer_2
- QLabel * sVerLabel
- QLabel * sLicenseLabel
- QLabel * label
- QSpacerItem * verticalSpacer
- QLabel * sDescLabel

12.86.1 Detailed Description

Definition at line 31 of file ui_animsettingdialog.h.

12.86.2 Member Function Documentation

12.86.2.1 void Ui_AnimSettingDialog::retranslateUi (QDialog * *AnimSettingDialog*) [inline]

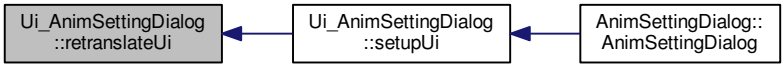
Definition at line 356 of file ui_animsettingdialog.h.

References infoTab, kpModeStopBox, kpReleaseBox, kpRepeatLabel, kpRepeatLabel_2, label, label_10, label_11, label_12, label_13, label_2, label_3, label_4, label_5, label_6, label_7, label_8, label_9, repeatLabel, repeatLabel_2, sAuthorLabel, sDescLabel, settingTab, sLicenseLabel, sNameLabel, sVerLabel, sYearLabel, tabWidget, and timeTab.

Referenced by setupUi().

```
357     {
358         AnimSettingDialog->setWindowTitle(QApplication::translate("AnimSettingDialog", "Dialog", Q_NULLPTR)
    );
359         label_2->setText(QApplication::translate("AnimSettingDialog", "Name:", Q_NULLPTR));
360         tabWidget->setTabText(tabWidget->indexOf(settingTab),
    QApplication::translate("AnimSettingDialog", "Settings", Q_NULLPTR));
361         kpRepeatLabel->setText(QApplication::translate("AnimSettingDialog", "Repeat every:",
    Q_NULLPTR));
362         repeatLabel->setText(QApplication::translate("AnimSettingDialog", "Repeat every:",
    Q_NULLPTR));
363         repeatLabel_2->setText(QApplication::translate("AnimSettingDialog", "seconds",
    Q_NULLPTR));
364         kpRepeatLabel_2->setText(QApplication::translate("AnimSettingDialog", "seconds",
    Q_NULLPTR));
365         label_4->setText(QApplication::translate("AnimSettingDialog", "<b>On mode start</b>",
    Q_NULLPTR));
366         label_8->setText(QApplication::translate("AnimSettingDialog", "seconds", Q_NULLPTR));
367         label_13->setText(QApplication::translate("AnimSettingDialog", "seconds", Q_NULLPTR));
368         label_12->setText(QApplication::translate("AnimSettingDialog", "Start delay:", Q_NULLPTR));
369         kpModeStopBox->setText(QApplication::translate("AnimSettingDialog", "Stop on key press
    ", Q_NULLPTR));
370         label_10->setText(QApplication::translate("AnimSettingDialog", "Start delay:", Q_NULLPTR));
371         kpReleaseBox->setText(QApplication::translate("AnimSettingDialog", "Stop on key release
    ", Q_NULLPTR));
372         label_6->setText(QApplication::translate("AnimSettingDialog", "<b>On key press</b>",
    Q_NULLPTR));
373         tabWidget->setTabText(tabWidget->indexOf(timeTab), QApplication::translate
    ("AnimSettingDialog", "Timing", Q_NULLPTR));
374         sNameLabel->setText(QApplication::translate("AnimSettingDialog", "Name", Q_NULLPTR));
375         label_7->setText(QApplication::translate("AnimSettingDialog", "Year:", Q_NULLPTR));
376         label_11->setText(QApplication::translate("AnimSettingDialog", "Description:", Q_NULLPTR));
377         sAuthorLabel->setText(QApplication::translate("AnimSettingDialog", "Author", Q_NULLPTR)
    );
378         label_9->setText(QApplication::translate("AnimSettingDialog", "License:", Q_NULLPTR));
379         label_5->setText(QApplication::translate("AnimSettingDialog", "Version:", Q_NULLPTR));
380         sYearLabel->setText(QApplication::translate("AnimSettingDialog", "Year", Q_NULLPTR));
381         label_3->setText(QApplication::translate("AnimSettingDialog", "Author:", Q_NULLPTR));
382         sVerLabel->setText(QApplication::translate("AnimSettingDialog", "Version", Q_NULLPTR));
383         sLicenseLabel->setText(QApplication::translate("AnimSettingDialog", "License",
    Q_NULLPTR));
384         label->setText(QApplication::translate("AnimSettingDialog", "Script name:", Q_NULLPTR));
385         sDescLabel->setText(QApplication::translate("AnimSettingDialog", "Description", Q_NULLPTR
    ));
386         tabWidget->setTabText(tabWidget->indexOf(infoTab), QApplication::translate
    ("AnimSettingDialog", "About", Q_NULLPTR));
387     } // retranslateUi
```

Here is the caller graph for this function:



12.86.2.2 void Ui_AnimSettingDialog::setupUi (QDialog * *AnimSettingDialog*) [inline]

Definition at line 85 of file ui_animsettingdialog.h.

References animName, buttonBox, delayBox, gridLayout, gridLayout_2, horizontalSpacer, horizontalSpacer_2, horizontalSpacer_3, horizontalSpacer_4, horizontalSpacer_5, horizontalSpacer_6, infoTab, kpDelayBox, kpModeStopBox, kpReleaseBox, kpRepeatBox, kpRepeatLabel, kpRepeatLabel_2, label, label_10, label_11, label_12, label_13, label_2, label_3, label_4, label_5, label_6, label_7, label_8, label_9, line, line_2, repeatBox, repeatLabel, repeatLabel_2, retranslateUi(), sAuthorLabel, sDescLabel, settingsGrid, settingTab, sLicenseLabel, sNameLabel, sVerLabel, sYearLabel, tabWidget, timeGrid, timeTab, verticalSpacer, verticalSpacer_2, and verticalSpacer_3.

Referenced by AnimSettingDialog::AnimSettingDialog().

```
86     {
87         if (AnimSettingDialog->objectName().isEmpty())
88             AnimSettingDialog->setObjectName(QStringLiteral("AnimSettingDialog"));
89         AnimSettingDialog->resize(480, 457);
90         AnimSettingDialog->setMinimumSize(QSize(480, 0));
91         AnimSettingDialog->setFocusPolicy(Qt::ClickFocus);
92         gridLayout = new QGridLayout(AnimSettingDialog);
93         gridLayout->setObjectName(QStringLiteral("gridLayout"));
94         buttonBox = new QDialogButtonBox(AnimSettingDialog);
95         buttonBox->setObjectName(QStringLiteral("buttonBox"));
96         buttonBox->setOrientation(Qt::Horizontal);
97         buttonBox->setStandardButtons(QDialogButtonBox::Cancel|QDialogButtonBox::Ok);
98 
99         gridLayout->addWidget(buttonBox, 2, 0, 1, 1);
100 
101         tabWidget = new QTabWidget(AnimSettingDialog);
102         tabWidget->setObjectName(QStringLiteral("tabWidget"));
103         settingTab = new QWidget();
104         settingTab->setObjectName(QStringLiteral("settingTab"));
105         settingsGrid = new QGridLayout(settingTab);
106         settingsGrid->setSpacing(12);
107         settingsGrid->setObjectName(QStringLiteral("settingsGrid"));
108         settingsGrid->setContentsMargins(12, 12, 12, 12);
109         label_2 = new QLabel(settingTab);
110         label_2->setObjectName(QStringLiteral("label_2"));
111 
112         settingsGrid->addWidget(label_2, 0, 1, 1, 1);
113 
114         horizontalSpacer_3 = new QSpacerItem(5, 0, QSizePolicy::Fixed,
    QSizePolicy::Minimum);
115 
116         settingsGrid->addItem(horizontalSpacer_3, 0, 2, 1, 1);
117 
118         animName = new QLineEdit(settingTab);
119         animName->setObjectName(QStringLiteral("animName"));
120 
121         settingsGrid->addWidget(animName, 0, 3, 1, 3);
122 
123         tabWidget->addTab(settingTab, QString());
124         timeTab = new QWidget();
125         timeTab->setObjectName(QStringLiteral("timeTab"));
126         timeGrid = new QGridLayout(timeTab);
127         timeGrid->setSpacing(12);
128         timeGrid->setObjectName(QStringLiteral("timeGrid"));
129         timeGrid->setContentsMargins(12, 12, 12, 12);
130         horizontalSpacer_5 = new QSpacerItem(15, 0, QSizePolicy::Fixed,
    QSizePolicy::Minimum);
131 
132         timeGrid->addItem(horizontalSpacer_5, 3, 2, 1, 1);
133 
134         kpRepeatLabel = new QLabel(timeTab);
135         kpRepeatLabel->setObjectName(QStringLiteral("kpRepeatLabel"));
136 
137         timeGrid->addWidget(kpRepeatLabel, 11, 1, 1, 1);
138 
139         repeatLabel = new QLabel(timeTab);
140         repeatLabel->setObjectName(QStringLiteral("repeatLabel"));
141 
142         timeGrid->addWidget(repeatLabel, 3, 1, 1, 1);
143 
144         repeatBox = new QDoubleSpinBox(timeTab);
145         repeatBox->setObjectName(QStringLiteral("repeatBox"));
146         repeatBox->setDecimals(1);
147         repeatBox->setMinimum(0.1);
148         repeatBox->setMaximum(86400);
149 
150         timeGrid->addWidget(repeatBox, 3, 3, 1, 1);
151 
152         repeatLabel_2 = new QLabel(timeTab);
153         repeatLabel_2->setObjectName(QStringLiteral("repeatLabel_2"));
```

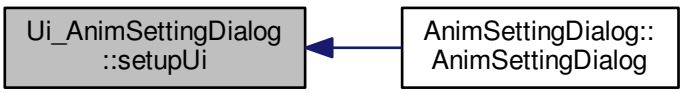
```
154
155         timeGrid->addWidget(repeatLabel_2, 3, 4, 1, 1);
156
157         line = new QFrame(timeTab);
158         line->setObjectName(QStringLiteral("line"));
159         line->setFrameShape(QFrame::HLine);
160         line->setFrameShadow(QFrame::Sunken);
161
162         timeGrid->addWidget(line, 1, 0, 1, 6);
163
164         kpRepeatLabel_2 = new QLabel(timeTab);
165         kpRepeatLabel_2->setObjectName(QStringLiteral("kpRepeatLabel_2"));
166
167         timeGrid->addWidget(kpRepeatLabel_2, 11, 4, 1, 1);
168
169         label_4 = new QLabel(timeTab);
170         label_4->setObjectName(QStringLiteral("label_4"));
171
172         timeGrid->addWidget(label_4, 0, 0, 1, 6);
173
174         label_8 = new QLabel(timeTab);
175         label_8->setObjectName(QStringLiteral("label_8"));
176
177         timeGrid->addWidget(label_8, 2, 4, 1, 1);
178
179         kpDelayBox = new QDoubleSpinBox(timeTab);
180         kpDelayBox->setObjectName(QStringLiteral("kpDelayBox"));
181         kpDelayBox->setDecimals(1);
182         kpDelayBox->setMaximum(86400);
183
184         timeGrid->addWidget(kpDelayBox, 10, 3, 1, 1);
185
186         label_13 = new QLabel(timeTab);
187         label_13->setObjectName(QStringLiteral("label_13"));
188
189         timeGrid->addWidget(label_13, 10, 4, 1, 1);
190
191         label_12 = new QLabel(timeTab);
192         label_12->setObjectName(QStringLiteral("label_12"));
193
194         timeGrid->addWidget(label_12, 10, 1, 1, 1);
195
196         delayBox = new QDoubleSpinBox(timeTab);
197         delayBox->setObjectName(QStringLiteral("delayBox"));
198         delayBox->setDecimals(1);
199         delayBox->setMaximum(86400);
200
201         timeGrid->addWidget(delayBox, 2, 3, 1, 1);
202
203         kpModeStopBox = new QCheckBox(timeTab);
204         kpModeStopBox->setObjectName(QStringLiteral("kpModeStopBox"));
205
206         timeGrid->addWidget(kpModeStopBox, 5, 1, 1, 3);
207
208         label_10 = new QLabel(timeTab);
209         label_10->setObjectName(QStringLiteral("label_10"));
210
211         timeGrid->addWidget(label_10, 2, 1, 1, 1);
212
213         verticalSpacer_3 = new QSpacerItem(0, 10, QSizePolicy::Minimum, QSizePolicy::Fixed)
    ;
214
215         timeGrid->addItem(verticalSpacer_3, 7, 1, 1, 1);
216
217         horizontalSpacer_4 = new QSpacerItem(40, 20, QSizePolicy::Expanding,
    QSizePolicy::Minimum);
218
219         timeGrid->addItem(horizontalSpacer_4, 2, 5, 1, 1);
220
221         kpReleaseBox = new QCheckBox(timeTab);
222         kpReleaseBox->setObjectName(QStringLiteral("kpReleaseBox"));
223
224         timeGrid->addWidget(kpReleaseBox, 14, 1, 1, 3);
225
226         label_6 = new QLabel(timeTab);
227         label_6->setObjectName(QStringLiteral("label_6"));
228
229         timeGrid->addWidget(label_6, 8, 0, 1, 6);
230
231         kpRepeatBox = new QDoubleSpinBox(timeTab);
232         kpRepeatBox->setObjectName(QStringLiteral("kpRepeatBox"));
233         kpRepeatBox->setDecimals(1);
234         kpRepeatBox->setMinimum(0.1);
235         kpRepeatBox->setMaximum(86400);
236
237         timeGrid->addWidget(kpRepeatBox, 11, 3, 1, 1);
238
```

```
239         verticalSpacer_2 = new QSpacerItem(20, 40, QSizePolicy::Minimum,
    QSizePolicy::Expanding);
240
241         timeGrid->addItem(verticalSpacer_2, 15, 1, 1, 1);
242
243         line_2 = new QFrame(timeTab);
244         line_2->setObjectName(QStringLiteral("line_2"));
245         line_2->setFrameShape(QFrame::HLine);
246         line_2->setFrameShadow(QFrame::Sunken);
247
248         timeGrid->addWidget(line_2, 9, 0, 1, 6);
249
250         horizontalSpacer_6 = new QSpacerItem(0, 0, QSizePolicy::Fixed,
    QSizePolicy::Minimum);
251
252         timeGrid->addItem(horizontalSpacer_6, 12, 2, 1, 1);
253
254         tabWidget->addTab(timeTab, QString());
255         infoTab = new QWidget();
256         infoTab->setObjectName(QStringLiteral("infoTab"));
257         gridLayout_2 = new QGridLayout(infoTab);
258         gridLayout_2->setSpacing(12);
259         gridLayout_2->setObjectName(QStringLiteral("gridLayout_2"));
260         gridLayout_2->setContentsMargins(12, 12, 12, 12);
261         sNameLabel = new QLabel(infoTab);
262         sNameLabel->setObjectName(QStringLiteral("sNameLabel"));
263
264         gridLayout_2->addWidget(sNameLabel, 0, 2, 1, 1);
265
266         label_7 = new QLabel(infoTab);
267         label_7->setObjectName(QStringLiteral("label_7"));
268
269         gridLayout_2->addWidget(label_7, 3, 0, 1, 1);
270
271         label_11 = new QLabel(infoTab);
272         label_11->setObjectName(QStringLiteral("label_11"));
273
274         gridLayout_2->addWidget(label_11, 5, 0, 1, 1);
275
276         sAuthorLabel = new QLabel(infoTab);
277         sAuthorLabel->setObjectName(QStringLiteral("sAuthorLabel"));
278
279         gridLayout_2->addWidget(sAuthorLabel, 2, 2, 1, 1);
280
281         label_9 = new QLabel(infoTab);
282         label_9->setObjectName(QStringLiteral("label_9"));
283
284         gridLayout_2->addWidget(label_9, 4, 0, 1, 1);
285
286         label_5 = new QLabel(infoTab);
287         label_5->setObjectName(QStringLiteral("label_5"));
288
289         gridLayout_2->addWidget(label_5, 1, 0, 1, 1);
290
291         sYearLabel = new QLabel(infoTab);
292         sYearLabel->setObjectName(QStringLiteral("sYearLabel"));
293
294         gridLayout_2->addWidget(sYearLabel, 3, 2, 1, 1);
295
296         label_3 = new QLabel(infoTab);
297         label_3->setObjectName(QStringLiteral("label_3"));
298
299         gridLayout_2->addWidget(label_3, 2, 0, 1, 1);
300
301         horizontalSpacer = new QSpacerItem(40, 20, QSizePolicy::Expanding,
    QSizePolicy::Minimum);
302
303         gridLayout_2->addItem(horizontalSpacer, 0, 3, 1, 1);
304
305         horizontalSpacer_2 = new QSpacerItem(20, 20, QSizePolicy::Fixed,
    QSizePolicy::Minimum);
306
307         gridLayout_2->addItem(horizontalSpacer_2, 0, 1, 1, 1);
308
309         sVerLabel = new QLabel(infoTab);
310         sVerLabel->setObjectName(QStringLiteral("sVerLabel"));
311
312         gridLayout_2->addWidget(sVerLabel, 1, 2, 1, 1);
313
314         sLicenseLabel = new QLabel(infoTab);
315         sLicenseLabel->setObjectName(QStringLiteral("sLicenseLabel"));
316
317         gridLayout_2->addWidget(sLicenseLabel, 4, 2, 1, 1);
318
319         label = new QLabel(infoTab);
320         label->setObjectName(QStringLiteral("label"));
321
```

```
322         gridLayout_2->addWidget(label, 0, 0, 1, 1);
323
324         verticalSpacer = new QSpacerItem(20, 40, QSizePolicy::Minimum, QSizePolicy::Expanding
    );
325
326         gridLayout_2->addItem(verticalSpacer, 6, 0, 1, 1);
327
328         sDescLabel = new QLabel(infoTab);
329         sDescLabel->setObjectName(QStringLiteral("sDescLabel"));
330         sDescLabel->setAlignment(Qt::AlignLeading|Qt::AlignLeft|Qt::AlignTop);
331
332         gridLayout_2->addWidget(sDescLabel, 5, 2, 2, 2);
333
334         tabWidget->addTab(infoTab, QString());
335
336         gridLayout->addWidget(tabWidget, 0, 0, 1, 1);
337
338         QWidget::setTabOrder(tabWidget, animName);
339         QWidget::setTabOrder(animName, delayBox);
340         QWidget::setTabOrder(delayBox, repeatBox);
341         QWidget::setTabOrder(repeatBox, kpModeStopBox);
342         QWidget::setTabOrder(kpModeStopBox, kpDelayBox);
343         QWidget::setTabOrder(kpDelayBox, kpRepeatBox);
344         QWidget::setTabOrder(kpRepeatBox, kpReleaseBox);
345
346         retranslateUi(AnimSettingDialog);
347         QObject::connect(buttonBox, SIGNAL(accepted()), AnimSettingDialog, SLOT(accept()));
348         QObject::connect(buttonBox, SIGNAL(rejected()), AnimSettingDialog, SLOT(reject()));
349
350         tabWidget->setCurrentIndex(0);
351
352
353         QMetaObject::connectSlotsByName(AnimSettingDialog);
354     } // setupUi
```

Here is the call graph for this function:

Here is the caller graph for this function:



12.86.3 Field Documentation

12.86.3.1 QLineEdit* Ui_AnimSettingDialog::animName

Definition at line 41 of file ui_animsettingdialog.h.

Referenced by AnimSettingDialog::AnimSettingDialog(), AnimSettingDialog::name(), and setupUi().

12.86.3.2 QDialogButtonBox* Ui_AnimSettingDialog::buttonBox

Definition at line 35 of file ui_animsettingdialog.h.

Referenced by setupUi().

12.86.3.3 QDoubleSpinBox* Ui_AnimSettingDialog::delayBox

Definition at line 56 of file ui_animsettingdialog.h.

Referenced by AnimSettingDialog::AnimSettingDialog(), and setupUi().

12.86.3.4 QGridLayout* Ui_AnimSettingDialog::gridLayout

Definition at line 34 of file ui_animsettingdialog.h.

Referenced by setupUi().

12.86.3.5 QGridLayout* Ui_AnimSettingDialog::gridLayout_2

Definition at line 68 of file ui_animsettingdialog.h.

Referenced by setupUi().

12.86.3.6 QSpacerItem* Ui_AnimSettingDialog::horizontalSpacer

Definition at line 77 of file ui_animsettingdialog.h.

Referenced by setupUi().

12.86.3.7 QSpacerItem* Ui_AnimSettingDialog::horizontalSpacer_2

Definition at line 78 of file ui_animsettingdialog.h.

Referenced by setupUi().

12.86.3.8 QSpacerItem* Ui_AnimSettingDialog::horizontalSpacer_3

Definition at line 40 of file ui_animsettingdialog.h.

Referenced by setupUi().

12.86.3.9 QSpacerItem* Ui_AnimSettingDialog::horizontalSpacer_4

Definition at line 60 of file ui_animsettingdialog.h.

Referenced by setupUi().

12.86.3.10 QSpacerItem* Ui_AnimSettingDialog::horizontalSpacer_5

Definition at line 44 of file ui_animsettingdialog.h.

Referenced by setupUi().

12.86.3.11 QSpacerItem* Ui_AnimSettingDialog::horizontalSpacer_6

Definition at line 66 of file ui_animsettingdialog.h.

Referenced by setupUi().

12.86.3.12 QWidget* Ui_AnimSettingDialog::infoTab

Definition at line 67 of file ui_animsettingdialog.h.

Referenced by retranslateUi(), and setupUi().

12.86.3.13 QDoubleSpinBox* Ui_AnimSettingDialog::kpDelayBox

Definition at line 53 of file ui_animsettingdialog.h.

Referenced by AnimSettingDialog::AnimSettingDialog(), and setupUi().

12.86.3.14 QCheckBox* Ui_AnimSettingDialog::kpModeStopBox

Definition at line 57 of file ui_animsettingdialog.h.

Referenced by AnimSettingDialog::AnimSettingDialog(), retranslateUi(), and setupUi().

12.86.3.15 QCheckBox* Ui_AnimSettingDialog::kpReleaseBox

Definition at line 61 of file ui_animsettingdialog.h.

Referenced by AnimSettingDialog::AnimSettingDialog(), retranslateUi(), and setupUi().

12.86.3.16 QDoubleSpinBox* Ui_AnimSettingDialog::kpRepeatBox

Definition at line 63 of file ui_animsettingdialog.h.

Referenced by AnimSettingDialog::AnimSettingDialog(), and setupUi().

12.86.3.17 QLabel* Ui_AnimSettingDialog::kpRepeatLabel

Definition at line 45 of file ui_animsettingdialog.h.

Referenced by AnimSettingDialog::AnimSettingDialog(), retranslateUi(), and setupUi().

12.86.3.18 QLabel* Ui_AnimSettingDialog::kpRepeatLabel_2

Definition at line 50 of file ui_animsettingdialog.h.

Referenced by AnimSettingDialog::AnimSettingDialog(), retranslateUi(), and setupUi().

12.86.3.19 QLabel* Ui_AnimSettingDialog::label

Definition at line 81 of file ui_animsettingdialog.h.

Referenced by retranslateUi(), and setupUi().

12.86.3.20 QLabel* Ui_AnimSettingDialog::label_10

Definition at line 58 of file ui_animsettingdialog.h.

Referenced by retranslateUi(), and setupUi().

12.86.3.21 QLabel* Ui_AnimSettingDialog::label_11

Definition at line 71 of file ui_animsettingdialog.h.

Referenced by retranslateUi(), and setupUi().

12.86.3.22 QLabel* Ui_AnimSettingDialog::label_12

Definition at line 55 of file ui_animsettingdialog.h.

Referenced by retranslateUi(), and setupUi().

12.86.3.23 QLabel* Ui_AnimSettingDialog::label_13

Definition at line 54 of file ui_animsettingdialog.h.

Referenced by retranslateUi(), and setupUi().

12.86.3.24 QLabel* Ui_AnimSettingDialog::label_2

Definition at line 39 of file ui_animsettingdialog.h.

Referenced by retranslateUi(), and setupUi().

12.86.3.25 QLabel* Ui_AnimSettingDialog::label_3

Definition at line 76 of file ui_animsettingdialog.h.

Referenced by retranslateUi(), and setupUi().

12.86.3.26 QLabel* Ui_AnimSettingDialog::label_4

Definition at line 51 of file ui_animsettingdialog.h.

Referenced by retranslateUi(), and setupUi().

12.86.3.27 QLabel* Ui_AnimSettingDialog::label_5

Definition at line 74 of file ui_animsettingdialog.h.

Referenced by retranslateUi(), and setupUi().

12.86.3.28 QLabel* Ui_AnimSettingDialog::label_6

Definition at line 62 of file ui_animsettingdialog.h.

Referenced by retranslateUi(), and setupUi().

12.86.3.29 QLabel* Ui_AnimSettingDialog::label_7

Definition at line 70 of file ui_animsettingdialog.h.

Referenced by retranslateUi(), and setupUi().

12.86.3.30 QLabel* Ui_AnimSettingDialog::label_8

Definition at line 52 of file ui_animsettingdialog.h.

Referenced by retranslateUi(), and setupUi().

12.86.3.31 QLabel* Ui_AnimSettingDialog::label_9

Definition at line 73 of file ui_animsettingdialog.h.

Referenced by retranslateUi(), and setupUi().

12.86.3.32 QFrame* Ui_AnimSettingDialog::line

Definition at line 49 of file ui_animsettingdialog.h.

Referenced by setupUi().

12.86.3.33 QFrame* Ui_AnimSettingDialog::line_2

Definition at line 65 of file ui_animsettingdialog.h.

Referenced by setupUi().

12.86.3.34 QDoubleSpinBox* Ui_AnimSettingDialog::repeatBox

Definition at line 47 of file ui_animsettingdialog.h.

Referenced by AnimSettingDialog::AnimSettingDialog(), and setupUi().

12.86.3.35 QLabel* Ui_AnimSettingDialog::repeatLabel

Definition at line 46 of file ui_animsettingdialog.h.

Referenced by AnimSettingDialog::AnimSettingDialog(), retranslateUi(), and setupUi().

12.86.3.36 QLabel* Ui_AnimSettingDialog::repeatLabel_2

Definition at line 48 of file ui_animsettingdialog.h.

Referenced by AnimSettingDialog::AnimSettingDialog(), retranslateUi(), and setupUi().

12.86.3.37 QLabel* Ui_AnimSettingDialog::sAuthorLabel

Definition at line 72 of file ui_animsettingdialog.h.

Referenced by AnimSettingDialog::AnimSettingDialog(), retranslateUi(), and setupUi().

12.86.3.38 QLabel* Ui_AnimSettingDialog::sDescLabel

Definition at line 83 of file ui_animsettingdialog.h.

Referenced by AnimSettingDialog::AnimSettingDialog(), retranslateUi(), and setupUi().

12.86.3.39 QGridLayout* Ui_AnimSettingDialog::settingsGrid

Definition at line 38 of file ui_animsettingdialog.h.

Referenced by AnimSettingDialog::AnimSettingDialog(), and setupUi().

12.86.3.40 QWidget* Ui_AnimSettingDialog::settingTab

Definition at line 37 of file ui_animsettingdialog.h.

Referenced by retranslateUi(), and setupUi().

12.86.3.41 QLabel* Ui_AnimSettingDialog::sLicenseLabel

Definition at line 80 of file ui_animsettingdialog.h.

Referenced by AnimSettingDialog::AnimSettingDialog(), retranslateUi(), and setupUi().

12.86.3.42 QLabel* Ui_AnimSettingDialog::sNameLabel

Definition at line 69 of file ui_animsettingdialog.h.

Referenced by AnimSettingDialog::AnimSettingDialog(), retranslateUi(), and setupUi().

12.86.3.43 QLabel* Ui_AnimSettingDialog::sVerLabel

Definition at line 79 of file ui_animsettingdialog.h.

Referenced by AnimSettingDialog::AnimSettingDialog(), retranslateUi(), and setupUi().

12.86.3.44 QLabel* Ui_AnimSettingDialog::sYearLabel

Definition at line 75 of file ui_animsettingdialog.h.

Referenced by AnimSettingDialog::AnimSettingDialog(), retranslateUi(), and setupUi().

12.86.3.45 QTabWidget* Ui_AnimSettingDialog::tabWidget

Definition at line 36 of file ui_animsettingdialog.h.

Referenced by retranslateUi(), and setupUi().

12.86.3.46 QGridLayout* Ui_AnimSettingDialog::timeGrid

Definition at line 43 of file ui_animsettingdialog.h.

Referenced by AnimSettingDialog::AnimSettingDialog(), and setupUi().

12.86.3.47 QWidget* Ui_AnimSettingDialog::timeTab

Definition at line 42 of file ui_animsettingdialog.h.

Referenced by retranslateUi(), and setupUi().

12.86.3.48 QSpacerItem* Ui_AnimSettingDialog::verticalSpacer

Definition at line 82 of file ui_animsettingdialog.h.

Referenced by setupUi().

12.86.3.49 QSpacerItem* Ui_AnimSettingDialog::verticalSpacer_2

Definition at line 64 of file ui_animsettingdialog.h.

Referenced by setupUi().

12.86.3.50 QSpacerItem* Ui_AnimSettingDialog::verticalSpacer_3

Definition at line 59 of file ui_animsettingdialog.h.

Referenced by setupUi().

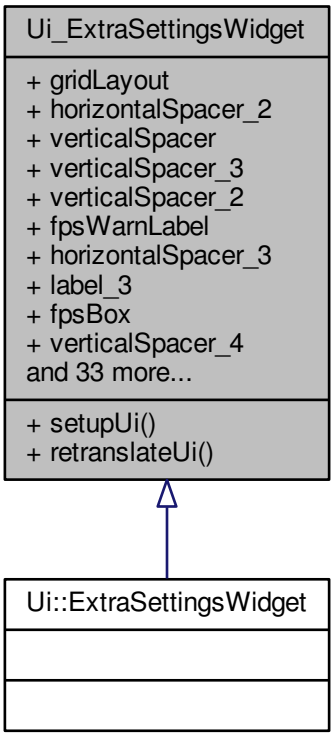
The documentation for this class was generated from the following file:

- src/ckb/ui_animsettingdialog.h

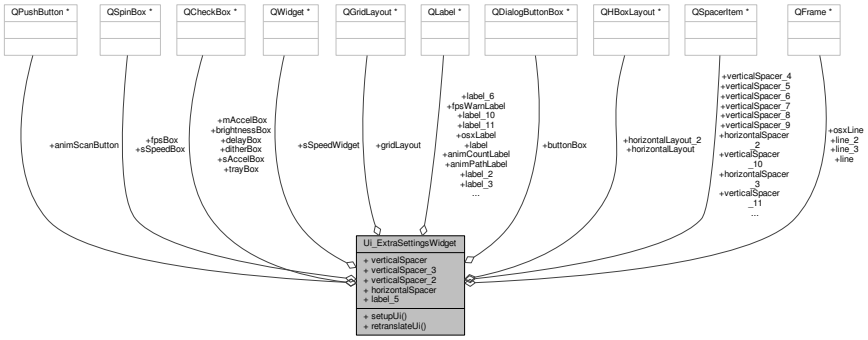
12.87 Ui_ExtraSettingsWidget Class Reference

`#include <src/ckb/ui_extrasettingswidget.h>`

Inheritance diagram for Ui_ExtraSettingsWidget:



Collaboration diagram for Ui_ExtraSettingsWidget:



Public Member Functions

- void setupUi (QDialog *ExtraSettingsWidget)
- void retranslateUi (QDialog *ExtraSettingsWidget)

Data Fields

- QGridLayout * gridLayout
- QSpacerItem * horizontalSpacer_2
- QSpacerItem * verticalSpacer
- QSpacerItem * verticalSpacer_3
- QSpacerItem * verticalSpacer_2
- QLabel * fpsWarnLabel
- QSpacerItem * horizontalSpacer_3
- QLabel * label_3
- QSpinBox * fpsBox
- QSpacerItem * verticalSpacer_4
- QSpacerItem * verticalSpacer_9
- QLabel * label_6
- QSpacerItem * verticalSpacer_6
- QSpacerItem * verticalSpacer_5
- QCheckBox * sAccelBox
- QSpacerItem * verticalSpacer_7
- QSpacerItem * verticalSpacer_8
- QSpacerItem * verticalSpacer_10
- QSpacerItem * verticalSpacer_11
- QSpacerItem * horizontalSpacer
- QFrame * line_2
- QHBoxLayout * horizontalLayout
- QLabel * animPathLabel
- QLabel * animCountLabel
- QPushButton * animScanButton
- QCheckBox * brightnessBox
- QCheckBox * trayBox
- QLabel * label
- QFrame * line
- QLabel * label_5
- QFrame * line_3
- QLabel * label_2
- QCheckBox * ditherBox
- QCheckBox * delayBox
- QLabel * osxLabel
- QFrame * osxLine
- QCheckBox * mAccelBox
- QWidget * sSpeedWidget
- QHBoxLayout * horizontalLayout_2
- QLabel * label_10
- QSpinBox * sSpeedBox
- QLabel * label_11
- QDialogButtonBox * buttonBox

12.87.1 Detailed Description

Definition at line 31 of file ui_extrasettingswidget.h.

12.87.2 Member Function Documentation

12.87.2.1 void Ui_ExtraSettingsWidget::retranslateUi (QDialog * *ExtraSettingsWidget*) [inline]

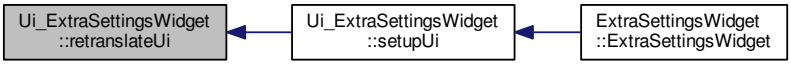
Definition at line 330 of file ui_extrasettingswidget.h.

References animCountLabel, animPathLabel, animScanButton, brightnessBox, delayBox, ditherBox, fpsWarnLabel, label, label_10, label_11, label_2, label_3, label_5, label_6, mAccelBox, osxLabel, sAccelBox, and trayBox.

Referenced by setupUi().

```
331     {
332         ExtraSettingsWidget->setWindowTitle(QApplication::translate("ExtraSettingsWidget", "Application
      Settings", Q_NULLPTR));
333         fpsWarnLabel->setText(QApplication::translate("ExtraSettingsWidget", "Warning: high
      frame rates may cause stability issues", Q_NULLPTR));
334         label_3->setText(QApplication::translate("ExtraSettingsWidget", "Frame rate (FPS):",
     Q_NULLPTR));
335         label_6->setText(QApplication::translate("ExtraSettingsWidget", "Location:", Q_NULLPTR));
336 #ifndef QT_NO_TOOLTIP
337         sAccelBox->setToolTip(QApplication::translate("ExtraSettingsWidget", "Try this if you're
      having problems with the scroll wheel.", Q_NULLPTR));
338 #endif // QT_NO_TOOLTIP
339         sAccelBox->setText(QApplication::translate("ExtraSettingsWidget", "Disable scroll
     acceleration", Q_NULLPTR));
340         animPathLabel->setText(QApplication::translate("ExtraSettingsWidget", "
     /path/to/animations", Q_NULLPTR));
341         animCountLabel->setText(QApplication::translate("ExtraSettingsWidget", "0 animations
     found", Q_NULLPTR));
342         animScanButton->setText(QApplication::translate("ExtraSettingsWidget", "Re-scan",
     Q_NULLPTR));
343 #ifndef QT_NO_TOOLTIP
344         brightnessBox->setToolTip(QApplication::translate("ExtraSettingsWidget", "By default,
      the same brightness level will be applied to all profiles and all devices. Enable this to store it with the
      lighting mode instead.", Q_NULLPTR));
345 #endif // QT_NO_TOOLTIP
346         brightnessBox->setText(QApplication::translate("ExtraSettingsWidget", "Set brightness
     per-mode", Q_NULLPTR));
347 #ifndef QT_NO_TOOLTIP
348         trayBox->setToolTip(QApplication::translate("ExtraSettingsWidget", "The tray icon will not
     be displayed. The application will still run in the background; re-launch the app to see the GUI again.",
     Q_NULLPTR));
349 #endif // QT_NO_TOOLTIP
350         trayBox->setText(QApplication::translate("ExtraSettingsWidget", "Hide tray icon", Q_NULLPTR)
    );
351         label->setText(QApplication::translate("ExtraSettingsWidget", "Behavior", Q_NULLPTR));
352         label_5->setText(QApplication::translate("ExtraSettingsWidget", "Animation scripts",
     Q_NULLPTR));
353         label_2->setText(QApplication::translate("ExtraSettingsWidget", "Hardware", Q_NULLPTR));
354 #ifndef QT_NO_TOOLTIP
355         ditherBox->setToolTip(QApplication::translate("ExtraSettingsWidget", "May improve
     appearance on some keyboards.", Q_NULLPTR));
356 #endif // QT_NO_TOOLTIP
357         ditherBox->setText(QApplication::translate("ExtraSettingsWidget", "Use spatial dithering
     to simulate extra color resolution", Q_NULLPTR));
358 #ifndef QT_NO_TOOLTIP
359         delayBox->setToolTip(QApplication::translate("ExtraSettingsWidget", "When using macros with
      strings longer than 25 chars, some OS may lose characters (e.g. Mint 17.2). Select to prevent that bug.",
     Q_NULLPTR));
360 #endif // QT_NO_TOOLTIP
361         delayBox->setText(QApplication::translate("ExtraSettingsWidget", "Use delay for very long
     macros", Q_NULLPTR));
362         osxLabel->setText(QApplication::translate("ExtraSettingsWidget", "OS X tweaks", Q_NULLPTR))
    ;
363 #ifndef QT_NO_TOOLTIP
364         mAccelBox->setToolTip(QApplication::translate("ExtraSettingsWidget", "Try this if you're
     having problems with mouse movement.", Q_NULLPTR));
365 #endif // QT_NO_TOOLTIP
366         mAccelBox->setText(QApplication::translate("ExtraSettingsWidget", "Disable mouse
     acceleration", Q_NULLPTR));
367         label_10->setText(QApplication::translate("ExtraSettingsWidget", "Scroll", Q_NULLPTR));
368         label_11->setText(QApplication::translate("ExtraSettingsWidget", "line(s) at a time",
     Q_NULLPTR));
369     } // retranslateUi
```

Here is the caller graph for this function:



12.87.2.2 void Ui_ExtraSettingsWidget::setupUi (QDialog * *ExtraSettingsWidget*) [inline]

Definition at line 78 of file ui_extrasettingswidget.h.

References animCountLabel, animPathLabel, animScanButton, brightnessBox, buttonBox, delayBox, ditherBox, fpsBox, fpsWarnLabel, gridLayout, horizontalLayout, horizontalLayout_2, horizontalSpacer, horizontalSpacer_2, horizontalSpacer_3, label, label_10, label_11, label_2, label_3, label_5, label_6, line, line_2, line_3, mAccelBox, osxLabel, osxLine, retranslateUi(), sAccelBox, sSpeedBox, sSpeedWidget, trayBox, verticalSpacer, verticalSpacer_10, verticalSpacer_11, verticalSpacer_2, verticalSpacer_3, verticalSpacer_4, verticalSpacer_5, verticalSpacer_6, verticalSpacer_7, verticalSpacer_8, and verticalSpacer_9.

Referenced by ExtraSettingsWidget::ExtraSettingsWidget().

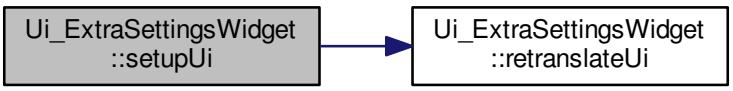
```
79     {
80         if (ExtraSettingsWidget->objectName().isEmpty())
81             ExtraSettingsWidget->setObjectName(QStringLiteral("ExtraSettingsWidget"));
82         ExtraSettingsWidget->resize(650, 515);
83         gridLayout = new QGridLayout(ExtraSettingsWidget);
84         gridLayout->setObjectName(QStringLiteral("gridLayout"));
85         horizontalSpacer_2 = new QSpacerItem(30, 20, QSizePolicy::Fixed,
    QSizePolicy::Minimum);
86
87         gridLayout->addItem(horizontalSpacer_2, 12, 2, 1, 1);
88
89         verticalSpacer = new QSpacerItem(0, 28, QSizePolicy::Minimum, QSizePolicy::Fixed);
90
91         gridLayout->addItem(verticalSpacer, 2, 0, 1, 1);
92
93         verticalSpacer_3 = new QSpacerItem(0, 28, QSizePolicy::Minimum, QSizePolicy::Fixed)
    ;
94
95         gridLayout->addItem(verticalSpacer_3, 3, 0, 1, 1);
96
97         verticalSpacer_2 = new QSpacerItem(20, 40, QSizePolicy::Minimum,
    QSizePolicy::Expanding);
98
99         gridLayout->addItem(verticalSpacer_2, 19, 5, 1, 1);
100
101         fpsWarnLabel = new QLabel(ExtraSettingsWidget);
102         fpsWarnLabel->setObjectName(QStringLiteral("fpsWarnLabel"));
103         QSizePolicy sizePolicy(QSizePolicy::Expanding, QSizePolicy::Preferred);
104         sizePolicy.setHorizontalStretch(0);
105         sizePolicy.setVerticalStretch(0);
106         sizePolicy.setHeightForWidth(fpsWarnLabel->sizePolicy().hasHeightForWidth());
107         fpsWarnLabel->setSizePolicy(sizePolicy);
108
109         gridLayout->addWidget(fpsWarnLabel, 12, 5, 1, 1);
110
111         horizontalSpacer_3 = new QSpacerItem(10, 20, QSizePolicy::Fixed,
    QSizePolicy::Minimum);
112
113         gridLayout->addItem(horizontalSpacer_3, 12, 4, 1, 1);
114
115         label_3 = new QLabel(ExtraSettingsWidget);
116         label_3->setObjectName(QStringLiteral("label_3"));
117
118         gridLayout->addWidget(label_3, 12, 1, 1, 1);
119
120         fpsBox = new QSpinBox(ExtraSettingsWidget);
121         fpsBox->setObjectName(QStringLiteral("fpsBox"));
122         fpsBox->setMinimum(5);
123         fpsBox->setMaximum(60);
124         fpsBox->setValue(30);
125
```

```
126         gridLayout->addWidget(fpsBox, 12, 3, 1, 1);
127 
128         verticalSpacer_4 = new QSpacerItem(0, 28, QSizePolicy::Minimum, QSizePolicy::Fixed)
    ;
129 
130         gridLayout->addItem(verticalSpacer_4, 7, 0, 1, 1);
131 
132         verticalSpacer_9 = new QSpacerItem(0, 28, QSizePolicy::Minimum, QSizePolicy::Fixed)
    ;
133 
134         gridLayout->addItem(verticalSpacer_9, 12, 0, 1, 1);
135 
136         label_6 = new QLabel(ExtraSettingsWidget);
137         label_6->setObjectName(QStringLiteral("label_6"));
138 
139         gridLayout->addWidget(label_6, 7, 1, 1, 1);
140 
141         verticalSpacer_6 = new QSpacerItem(20, 10, QSizePolicy::Minimum, QSizePolicy::Fixed
    );
142 
143         gridLayout->addItem(verticalSpacer_6, 8, 1, 1, 1);
144 
145         verticalSpacer_5 = new QSpacerItem(20, 10, QSizePolicy::Minimum, QSizePolicy::Fixed
    );
146 
147         gridLayout->addItem(verticalSpacer_5, 4, 1, 1, 1);
148 
149         sAccelBox = new QCheckBox(ExtraSettingsWidget);
150         sAccelBox->setObjectName(QStringLiteral("sAccelBox"));
151 
152         gridLayout->addWidget(sAccelBox, 18, 1, 1, 4);
153 
154         verticalSpacer_7 = new QSpacerItem(20, 10, QSizePolicy::Minimum, QSizePolicy::Fixed
    );
155 
156         gridLayout->addItem(verticalSpacer_7, 14, 1, 1, 1);
157 
158         verticalSpacer_8 = new QSpacerItem(0, 28, QSizePolicy::Minimum, QSizePolicy::Fixed)
    ;
159 
160         gridLayout->addItem(verticalSpacer_8, 13, 0, 1, 1);
161 
162         verticalSpacer_10 = new QSpacerItem(0, 28, QSizePolicy::Minimum,
    QSizePolicy::Fixed);
163 
164         gridLayout->addItem(verticalSpacer_10, 17, 0, 1, 1);
165 
166         verticalSpacer_11 = new QSpacerItem(0, 28, QSizePolicy::Minimum,
    QSizePolicy::Fixed);
167 
168         gridLayout->addItem(verticalSpacer_11, 18, 0, 1, 1);
169 
170         horizontalSpacer = new QSpacerItem(40, 20, QSizePolicy::Expanding,
    QSizePolicy::Minimum);
171 
172         gridLayout->addItem(horizontalSpacer, 12, 6, 1, 1);
173 
174         line_2 = new QFrame(ExtraSettingsWidget);
175         line_2->setObjectName(QStringLiteral("line_2"));
176         line_2->setFrameShape(QFrame::HLine);
177         line_2->setFrameShadow(QFrame::Sunken);
178 
179         gridLayout->addWidget(line_2, 10, 0, 1, 7);
180 
181         horizontalLayout = new QHBoxLayout();
182         horizontalLayout->setObjectName(QStringLiteral("horizontalLayout"));
183         animPathLabel = new QLabel(ExtraSettingsWidget);
184         animPathLabel->setObjectName(QStringLiteral("animPathLabel"));
185 
186         horizontalLayout->addWidget(animPathLabel);
187 
188         animCountLabel = new QLabel(ExtraSettingsWidget);
189         animCountLabel->setObjectName(QStringLiteral("animCountLabel"));
190         sizePolicy.setHeightForWidth(animCountLabel->sizePolicy().hasHeightForWidth());
191         animCountLabel->setSizePolicy(sizePolicy);
192         animCountLabel->setAlignment(Qt::AlignRight|Qt::AlignTrailing|Qt::AlignVCenter);
193 
194         horizontalLayout->addWidget(animCountLabel);
195 
196         animScanButton = new QPushButton(ExtraSettingsWidget);
197         animScanButton->setObjectName(QStringLiteral("animScanButton"));
198 
199         horizontalLayout->addWidget(animScanButton);
200 
201 
202         gridLayout->addLayout(horizontalLayout, 7, 3, 1, 4);
203 
```

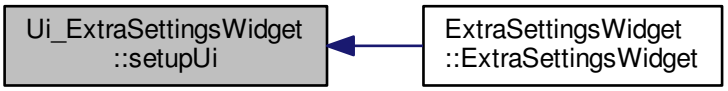
```
204         brightnessBox = new QCheckBox(ExtraSettingsWidget);
205         brightnessBox->setObjectName(QStringLiteral("brightnessBox"));
206 
207         gridLayout->addWidget(brightnessBox, 3, 1, 1, 6);
208 
209         trayBox = new QCheckBox(ExtraSettingsWidget);
210         trayBox->setObjectName(QStringLiteral("trayBox"));
211 
212         gridLayout->addWidget(trayBox, 2, 1, 1, 6);
213 
214         label = new QLabel(ExtraSettingsWidget);
215         label->setObjectName(QStringLiteral("label"));
216         QFont font;
217         font.setBold(true);
218         font.setWeight(75);
219         label->setFont(font);
220 
221         gridLayout->addWidget(label, 0, 0, 1, 7);
222 
223         line = new QFrame(ExtraSettingsWidget);
224         line->setObjectName(QStringLiteral("line"));
225         line->setFrameShape(QFrame::HLine);
226         line->setFrameShadow(QFrame::Sunken);
227 
228         gridLayout->addWidget(line, 1, 0, 1, 7);
229 
230         label_5 = new QLabel(ExtraSettingsWidget);
231         label_5->setObjectName(QStringLiteral("label_5"));
232         label_5->setFont(font);
233 
234         gridLayout->addWidget(label_5, 5, 0, 1, 7);
235 
236         line_3 = new QFrame(ExtraSettingsWidget);
237         line_3->setObjectName(QStringLiteral("line_3"));
238         line_3->setFrameShape(QFrame::HLine);
239         line_3->setFrameShadow(QFrame::Sunken);
240 
241         gridLayout->addWidget(line_3, 6, 0, 1, 7);
242 
243         label_2 = new QLabel(ExtraSettingsWidget);
244         label_2->setObjectName(QStringLiteral("label_2"));
245         label_2->setFont(font);
246 
247         gridLayout->addWidget(label_2, 9, 0, 1, 7);
248 
249         ditherBox = new QCheckBox(ExtraSettingsWidget);
250         ditherBox->setObjectName(QStringLiteral("ditherBox"));
251 
252         gridLayout->addWidget(ditherBox, 13, 1, 1, 6);
253 
254         delayBox = new QCheckBox(ExtraSettingsWidget);
255         delayBox->setObjectName(QStringLiteral("delayBox"));
256 
257         gridLayout->addWidget(delayBox, 14, 1, 1, 6);
258 
259         osxLabel = new QLabel(ExtraSettingsWidget);
260         osxLabel->setObjectName(QStringLiteral("osxLabel"));
261         osxLabel->setFont(font);
262 
263         gridLayout->addWidget(osxLabel, 15, 0, 1, 7);
264 
265         osxLine = new QFrame(ExtraSettingsWidget);
266         osxLine->setObjectName(QStringLiteral("osxLine"));
267         osxLine->setFrameShape(QFrame::HLine);
268         osxLine->setFrameShadow(QFrame::Sunken);
269 
270         gridLayout->addWidget(osxLine, 16, 0, 1, 7);
271 
272         mAccelBox = new QCheckBox(ExtraSettingsWidget);
273         mAccelBox->setObjectName(QStringLiteral("mAccelBox"));
274 
275         gridLayout->addWidget(mAccelBox, 17, 1, 1, 6);
276 
277         sSpeedWidget = new QWidget(ExtraSettingsWidget);
278         sSpeedWidget->setObjectName(QStringLiteral("sSpeedWidget"));
279         sizePolicy.setHeightForWidth(sSpeedWidget->sizePolicy().hasHeightForWidth());
280         sSpeedWidget->setSizePolicy(sizePolicy);
281         horizontalLayout_2 = new QHBoxLayout(sSpeedWidget);
282         horizontalLayout_2->setObjectName(QStringLiteral("horizontalLayout_2"));
283         horizontalLayout_2->setContentsMargins(0, 0, 0, 0);
284         label_10 = new QLabel(sSpeedWidget);
285         label_10->setObjectName(QStringLiteral("label_10"));
286 
287         horizontalLayout_2->addWidget(label_10);
288 
289         sSpeedBox = new QSpinBox(sSpeedWidget);
290         sSpeedBox->setObjectName(QStringLiteral("sSpeedBox"));
```

```
291         sSpeedBox->setMinimum(1);
292         sSpeedBox->setMaximum(10);
293         sSpeedBox->setValue(3);
294 
295         horizontalLayout_2->addWidget(sSpeedBox);
296 
297         label_11 = new QLabel(sSpeedWidget);
298         label_11->setObjectName(QStringLiteral("label_11"));
299         sizePolicy.setHeightForWidth(label_11->sizePolicy().hasHeightForWidth());
300         label_11->setSizePolicy(sizePolicy);
301 
302         horizontalLayout_2->addWidget(label_11);
303 
304 
305         gridLayout->addWidget(sSpeedWidget, 18, 5, 1, 2);
306 
307         buttonBox = new QDialogButtonBox(ExtraSettingsWidget);
308         buttonBox->setObjectName(QStringLiteral("buttonBox"));
309         buttonBox->setOrientation(Qt::Horizontal);
310         buttonBox->setStandardButtons(QDialogButtonBox::Ok);
311 
312         gridLayout->addWidget(buttonBox, 20, 0, 1, 7);
313 
314         QWidget::setTabOrder(trayBox, brightnessBox);
315         QWidget::setTabOrder(brightnessBox, animScanButton);
316         QWidget::setTabOrder(animScanButton, fpsBox);
317         QWidget::setTabOrder(fpsBox, ditherBox);
318         QWidget::setTabOrder(ditherBox, delayBox);
319         QWidget::setTabOrder(delayBox, mAccelBox);
320         QWidget::setTabOrder(mAccelBox, sAccelBox);
321         QWidget::setTabOrder(sAccelBox, sSpeedBox);
322 
323         retranslateUi(ExtraSettingsWidget);
324         QObject::connect(buttonBox, SIGNAL(accepted()), ExtraSettingsWidget, SLOT(accept()));
325         QObject::connect(buttonBox, SIGNAL(rejected()), ExtraSettingsWidget, SLOT(reject()));
326 
327         QMetaObject::connectSlotsByName(ExtraSettingsWidget);
328     } // setupUi
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.87.3 Field Documentation

12.87.3.1 QLabel* Ui_ExtraSettingsWidget::animCountLabel

Definition at line 57 of file ui_extrasettingswidget.h.

Referenced by ExtraSettingsWidget::on_animScanButton_clicked(), retranslateUi(), and setupUi().

12.87.3.2 QLabel* Ui_ExtraSettingsWidget::animPathLabel

Definition at line 56 of file ui_extrasettingswidget.h.

Referenced by ExtraSettingsWidget::ExtraSettingsWidget(), retranslateUi(), and setupUi().

12.87.3.3 QPushButton* Ui_ExtraSettingsWidget::animScanButton

Definition at line 58 of file ui_extrasettingswidget.h.

Referenced by retranslateUi(), and setupUi().

12.87.3.4 QCheckBox* Ui_ExtraSettingsWidget::brightnessBox

Definition at line 59 of file ui_extrasettingswidget.h.

Referenced by ExtraSettingsWidget::ExtraSettingsWidget(), retranslateUi(), and setupUi().

12.87.3.5 QDialogButtonBox* Ui_ExtraSettingsWidget::buttonBox

Definition at line 76 of file ui_extrasettingswidget.h.

Referenced by setupUi().

12.87.3.6 QCheckBox* Ui_ExtraSettingsWidget::delayBox

Definition at line 67 of file ui_extrasettingswidget.h.

Referenced by ExtraSettingsWidget::ExtraSettingsWidget(), retranslateUi(), and setupUi().

12.87.3.7 QCheckBox* Ui_ExtraSettingsWidget::ditherBox

Definition at line 66 of file ui_extrasettingswidget.h.

Referenced by ExtraSettingsWidget::ExtraSettingsWidget(), retranslateUi(), and setupUi().

12.87.3.8 QSpinBox* Ui_ExtraSettingsWidget::fpsBox

Definition at line 42 of file ui_extrasettingswidget.h.

Referenced by ExtraSettingsWidget::ExtraSettingsWidget(), and setupUi().

12.87.3.9 QLabel* Ui_ExtraSettingsWidget::fpsWarnLabel

Definition at line 39 of file ui_extrasettingswidget.h.

Referenced by ExtraSettingsWidget::ExtraSettingsWidget(), ExtraSettingsWidget::on_fpsBox_valueChanged(), retranslateUi(), and setupUi().

12.87.3.10 QGridLayout* Ui_ExtraSettingsWidget::gridLayout

Definition at line 34 of file ui_extrasettingswidget.h.

Referenced by setupUi().

12.87.3.11 QHBoxLayout* Ui_ExtraSettingsWidget::horizontalLayout

Definition at line 55 of file ui_extrasettingswidget.h.

Referenced by setupUi().

12.87.3.12 QHBoxLayout* Ui_ExtraSettingsWidget::horizontalLayout_2

Definition at line 72 of file ui_extrasettingswidget.h.

Referenced by setupUi().

12.87.3.13 QSpacerItem* Ui_ExtraSettingsWidget::horizontalSpacer

Definition at line 53 of file ui_extrasettingswidget.h.

Referenced by setupUi().

12.87.3.14 QSpacerItem* Ui_ExtraSettingsWidget::horizontalSpacer_2

Definition at line 35 of file ui_extrasettingswidget.h.

Referenced by setupUi().

12.87.3.15 QSpacerItem* Ui_ExtraSettingsWidget::horizontalSpacer_3

Definition at line 40 of file ui_extrasettingswidget.h.

Referenced by setupUi().

12.87.3.16 QLabel* Ui_ExtraSettingsWidget::label

Definition at line 61 of file ui_extrasettingswidget.h.

Referenced by retranslateUi(), and setupUi().

12.87.3.17 QLabel* Ui_ExtraSettingsWidget::label_10

Definition at line 73 of file ui_extrasettingswidget.h.

Referenced by retranslateUi(), and setupUi().

12.87.3.18 QLabel* Ui_ExtraSettingsWidget::label_11

Definition at line 75 of file ui_extrasettingswidget.h.

Referenced by retranslateUi(), and setupUi().

12.87.3.19 QLabel* Ui_ExtraSettingsWidget::label_2

Definition at line 65 of file ui_extrasettingswidget.h.

Referenced by retranslateUi(), and setupUi().

12.87.3.20 QLabel* Ui_ExtraSettingsWidget::label_3

Definition at line 41 of file ui_extrasettingswidget.h.

Referenced by retranslateUi(), and setupUi().

12.87.3.21 QLabel* Ui_ExtraSettingsWidget::label_5

Definition at line 63 of file ui_extrasettingswidget.h.

Referenced by retranslateUi(), and setupUi().

12.87.3.22 QLabel* Ui_ExtraSettingsWidget::label_6

Definition at line 45 of file ui_extrasettingswidget.h.

Referenced by retranslateUi(), and setupUi().

12.87.3.23 QFrame* Ui_ExtraSettingsWidget::line

Definition at line 62 of file ui_extrasettingswidget.h.

Referenced by setupUi().

12.87.3.24 QFrame* Ui_ExtraSettingsWidget::line_2

Definition at line 54 of file ui_extrasettingswidget.h.

Referenced by setupUi().

12.87.3.25 QFrame* Ui_ExtraSettingsWidget::line_3

Definition at line 64 of file ui_extrasettingswidget.h.

Referenced by setupUi().

12.87.3.26 QCheckBox* Ui_ExtraSettingsWidget::mAccelBox

Definition at line 70 of file ui_extrasettingswidget.h.

Referenced by ExtraSettingsWidget::ExtraSettingsWidget(), retranslateUi(), and setupUi().

12.87.3.27 QLabel* Ui_ExtraSettingsWidget::osxLabel

Definition at line 68 of file ui_extrasettingswidget.h.

Referenced by ExtraSettingsWidget::ExtraSettingsWidget(), retranslateUi(), and setupUi().

12.87.3.28 QFrame* Ui_ExtraSettingsWidget::osxLine

Definition at line 69 of file ui_extrasettingswidget.h.

Referenced by ExtraSettingsWidget::ExtraSettingsWidget(), and setupUi().

12.87.3.29 QCheckBox* Ui_ExtraSettingsWidget::sAccelBox

Definition at line 48 of file ui_extrasettingswidget.h.

Referenced by ExtraSettingsWidget::ExtraSettingsWidget(), ExtraSettingsWidget::on_sSpeedBox_valueChanged(), retranslateUi(), and setupUi().

12.87.3.30 QSpinBox* Ui_ExtraSettingsWidget::sSpeedBox

Definition at line 74 of file ui_extrasettingswidget.h.

Referenced by ExtraSettingsWidget::ExtraSettingsWidget(), ExtraSettingsWidget::on_sAccelBox_clicked(), and setupUi().

12.87.3.31 QWidget* Ui_ExtraSettingsWidget::sSpeedWidget

Definition at line 71 of file ui_extrasettingswidget.h.

Referenced by ExtraSettingsWidget::ExtraSettingsWidget(), ExtraSettingsWidget::on_sAccelBox_clicked(), and setupUi().

12.87.3.32 QCheckBox* Ui_ExtraSettingsWidget::trayBox

Definition at line 60 of file ui_extrasettingswidget.h.

Referenced by ExtraSettingsWidget::ExtraSettingsWidget(), retranslateUi(), and setupUi().

12.87.3.33 QSpacerItem* Ui_ExtraSettingsWidget::verticalSpacer

Definition at line 36 of file ui_extrasettingswidget.h.

Referenced by setupUi().

12.87.3.34 QSpacerItem* Ui_ExtraSettingsWidget::verticalSpacer_10

Definition at line 51 of file ui_extrasettingswidget.h.

Referenced by setupUi().

12.87.3.35 QSpacerItem* Ui_ExtraSettingsWidget::verticalSpacer_11

Definition at line 52 of file ui_extrasettingswidget.h.

Referenced by setupUi().

12.87.3.36 QSpacerItem* Ui_ExtraSettingsWidget::verticalSpacer_2

Definition at line 38 of file ui_extrasettingswidget.h.

Referenced by setupUi().

12.87.3.37 QSpacerItem* Ui_ExtraSettingsWidget::verticalSpacer_3

Definition at line 37 of file ui_extrasettingswidget.h.

Referenced by setupUi().

12.87.3.38 QSpacerItem* Ui_ExtraSettingsWidget::verticalSpacer_4

Definition at line 43 of file ui_extrasettingswidget.h.

Referenced by setupUi().

12.87.3.39 QSpacerItem* Ui_ExtraSettingsWidget::verticalSpacer_5

Definition at line 47 of file ui_extrasettingswidget.h.

Referenced by setupUi().

12.87.3.40 QSpacerItem* Ui_ExtraSettingsWidget::verticalSpacer_6

Definition at line 46 of file ui_extrasettingswidget.h.

Referenced by setupUi().

12.87.3.41 QSpacerItem* Ui_ExtraSettingsWidget::verticalSpacer_7

Definition at line 49 of file ui_extrasettingswidget.h.

Referenced by setupUi().

12.87.3.42 QSpacerItem* Ui_ExtraSettingsWidget::verticalSpacer_8

Definition at line 50 of file ui_extrasettingswidget.h.

Referenced by setupUi().

12.87.3.43 QSpacerItem* Ui_ExtraSettingsWidget::verticalSpacer_9

Definition at line 44 of file ui_extrasettingswidget.h.

Referenced by setupUi().

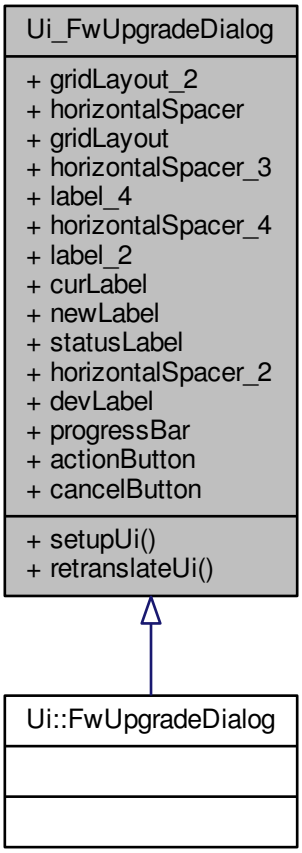
The documentation for this class was generated from the following file:

- src/ckb/ui_extrasettingswidget.h

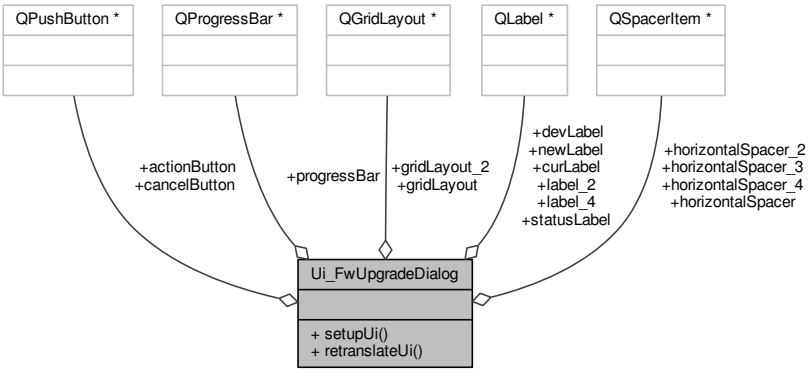
12.88 Ui_FwUpgradeDialog Class Reference

`#include <src/ckb/ui_fwupgradedialog.h>`

Inheritance diagram for Ui_FwUpgradeDialog:



Collaboration diagram for Ui_FwUpgradeDialog:



Public Member Functions

- void setupUi (QDialog *FwUpgradeDialog)
- void retranslateUi (QDialog *FwUpgradeDialog)

Data Fields

- QGridLayout * gridLayout_2
- QSpacerItem * horizontalSpacer
- QGridLayout * gridLayout
- QSpacerItem * horizontalSpacer_3
- QLabel * label_4
- QSpacerItem * horizontalSpacer_4
- QLabel * label_2
- QLabel * curLabel
- QLabel * newLabel
- QLabel * statusLabel
- QSpacerItem * horizontalSpacer_2
- QLabel * devLabel
- QProgressBar * progressBar
- QPushButton * actionButton
- QPushButton * cancelButton

12.88.1 Detailed Description

Definition at line 26 of file ui_fwupgradedialog.h.

12.88.2 Member Function Documentation

12.88.2.1 void Ui_FwUpgradeDialog::retranslateUi (QDialog * *FwUpgradeDialog*) [inline]

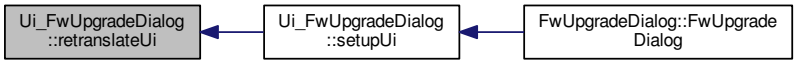
Definition at line 156 of file ui_fwupgradedialog.h.

References actionButton, cancelButton, curLabel, devLabel, label_2, label_4, newLabel, and statusLabel.

Referenced by setupUi().

```
157     {
158         FwUpgradeDialog->setWindowTitle(QApplication::translate("FwUpgradeDialog", "Firmware update",
    Q_NULLPTR));
159         label_4->setText(QApplication::translate("FwUpgradeDialog", "New:", Q_NULLPTR));
160         label_2->setText(QApplication::translate("FwUpgradeDialog", "Current:", Q_NULLPTR));
161         curLabel->setText(QApplication::translate("FwUpgradeDialog", "0.00", Q_NULLPTR));
162         newLabel->setText(QApplication::translate("FwUpgradeDialog", "0.00", Q_NULLPTR));
163         statusLabel->setText(QApplication::translate("FwUpgradeDialog", "Downloading new
     firmware, please wait...", Q_NULLPTR));
164         devLabel->setText(QApplication::translate("FwUpgradeDialog", "Device", Q_NULLPTR));
165         actionButton->setText(QApplication::translate("FwUpgradeDialog", "Start", Q_NULLPTR));
166         cancelButton->setText(QApplication::translate("FwUpgradeDialog", "Cancel", Q_NULLPTR));
167     } // retranslateUi
```

Here is the caller graph for this function:



12.88.2.2 void Ui_FwUpgradeDialog::setupUi (QDialog * *FwUpgradeDialog*) [inline]

Definition at line 45 of file ui_fwupgradedialog.h.

References actionButton, cancelButton, curLabel, devLabel, gridLayout, gridLayout_2, horizontalSpacer, horizontalSpacer_2, horizontalSpacer_3, horizontalSpacer_4, label_2, label_4, newLabel, progressBar, retranslateUi(), and statusLabel.

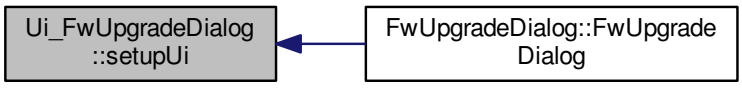
Referenced by FwUpgradeDialog::FwUpgradeDialog().

```
46     {
47         if (FwUpgradeDialog->objectName().isEmpty())
48             FwUpgradeDialog->setObjectName(QStringLiteral("FwUpgradeDialog"));
49         FwUpgradeDialog->resize(420, 280);
50         QSizePolicy sizePolicy(QSizePolicy::Fixed, QSizePolicy::Fixed);
51         sizePolicy.setHorizontalStretch(0);
52         sizePolicy.setVerticalStretch(0);
53         sizePolicy.setHeightForWidth(FwUpgradeDialog->sizePolicy().hasHeightForWidth());
54         FwUpgradeDialog->setSizePolicy(sizePolicy);
55         gridLayout_2 = new QGridLayout(FwUpgradeDialog);
56         gridLayout_2->setObjectName(QStringLiteral("gridLayout_2"));
57         horizontalSpacer = new QSpacerItem(127, 20, QSizePolicy::Expanding,
    QSizePolicy::Minimum);
58
59         gridLayout_2->addItem(horizontalSpacer, 1, 0, 1, 1);
60
61         gridLayout = new QGridLayout();
62         gridLayout->setObjectName(QStringLiteral("gridLayout"));
63         horizontalSpacer_3 = new QSpacerItem(40, 20, QSizePolicy::Expanding,
    QSizePolicy::Minimum);
64
65         gridLayout->addItem(horizontalSpacer_3, 1, 0, 1, 1);
66
67         label_4 = new QLabel(FwUpgradeDialog);
68         label_4->setObjectName(QStringLiteral("label_4"));
69         QFont font;
70         font.setBold(true);
71         font.setWeight(75);
72         label_4->setFont(font);
73
74         gridLayout->addWidget(label_4, 1, 4, 1, 1);
75
76         horizontalSpacer_4 = new QSpacerItem(40, 20, QSizePolicy::Expanding,
    QSizePolicy::Minimum);
77
78         gridLayout->addItem(horizontalSpacer_4, 1, 6, 1, 1);
79
80         label_2 = new QLabel(FwUpgradeDialog);
81         label_2->setObjectName(QStringLiteral("label_2"));
82         label_2->setFont(font);
83
84         gridLayout->addWidget(label_2, 1, 1, 1, 1);
85
86         curLabel = new QLabel(FwUpgradeDialog);
87         curLabel->setObjectName(QStringLiteral("curLabel"));
88
89         gridLayout->addWidget(curLabel, 1, 2, 1, 1);
90
91         newLabel = new QLabel(FwUpgradeDialog);
92         newLabel->setObjectName(QStringLiteral("newLabel"));
93
94         gridLayout->addWidget(newLabel, 1, 5, 1, 1);
95
96         statusLabel = new QLabel(FwUpgradeDialog);
97         statusLabel->setObjectName(QStringLiteral("statusLabel"));
98         QSizePolicy sizePolicy1(QSizePolicy::Preferred, QSizePolicy::Expanding);
99         sizePolicy1.setHorizontalStretch(0);
100         sizePolicy1.setVerticalStretch(0);
101         sizePolicy1.setHeightForWidth(statusLabel->sizePolicy().hasHeightForWidth());
102         statusLabel->setSizePolicy(sizePolicy1);
103         statusLabel->setAlignment(Qt::AlignCenter);
104         statusLabel->setWordWrap(true);
105
106         gridLayout->addWidget(statusLabel, 2, 0, 1, 7);
107
108         horizontalSpacer_2 = new QSpacerItem(40, 20, QSizePolicy::Expanding,
    QSizePolicy::Minimum);
109
110         gridLayout->addItem(horizontalSpacer_2, 1, 3, 1, 1);
111
112         devLabel = new QLabel(FwUpgradeDialog);
113         devLabel->setObjectName(QStringLiteral("devLabel"));
114         QSizePolicy sizePolicy2(QSizePolicy::Preferred, QSizePolicy::Minimum);
115         sizePolicy2.setHorizontalStretch(0);
```

```
116         sizePolicy2.setVerticalStretch(0);
117         sizePolicy2.setHeightForWidth(devLabel->sizePolicy().hasHeightForWidth());
118         devLabel->setSizePolicy(sizePolicy2);
119         devLabel->setFont(font);
120         devLabel->setAlignment(Qt::AlignCenter);
121
122         gridLayout->addWidget(devLabel, 0, 0, 1, 7);
123
124         progressBar = new QProgressBar(FwUpgradeDialog);
125         progressBar->setObjectName(QStringLiteral("progressBar"));
126         progressBar->setMaximum(0);
127         progressBar->setValue(0);
128
129         gridLayout->addWidget(progressBar, 3, 0, 1, 7);
130
131
132         gridLayout_2->addLayout(gridLayout, 0, 0, 1, 3);
133
134         actionButton = new QPushButton(FwUpgradeDialog);
135         actionButton->setObjectName(QStringLiteral("actionButton"));
136         actionButton->setEnabled(false);
137
138         gridLayout_2->addWidget(actionButton, 1, 2, 1, 1);
139
140         cancelButton = new QPushButton(FwUpgradeDialog);
141         cancelButton->setObjectName(QStringLiteral("cancelButton"));
142         cancelButton->setEnabled(false);
143
144         gridLayout_2->addWidget(cancelButton, 1, 1, 1, 1);
145
146         QWidget::setTabOrder(actionButton, cancelButton);
147
148         retranslateUi(FwUpgradeDialog);
149
150         actionButton->setDefault(true);
151
152
153         QMetaObject::connectSlotsByName(FwUpgradeDialog);
154     } // setupUi
```

Here is the call graph for this function:

Here is the caller graph for this function:



12.88.3 Field Documentation

12.88.3.1 QPushButton* Ui_FwUpgradeDialog::actionButton

Definition at line 42 of file ui_fwupgradedialog.h.

Referenced by FwUpgradeDialog::exec(), FwUpgradeDialog::fwUpdateFinished(), FwUpgradeDialog::on_actionButton_clicked(), retranslateUi(), and setupUi().

12.88.3.2 QPushButton* Ui_FwUpgradeDialog::cancelButton

Definition at line 43 of file ui_fwupgradedialog.h.

Referenced by FwUpgradeDialog::exec(), FwUpgradeDialog::on_actionButton_clicked(), retranslateUi(), and setupUi().

12.88.3.3 QLabel* Ui_FwUpgradeDialog::curLabel

Definition at line 36 of file ui_fwupgradedialog.h.

Referenced by FwUpgradeDialog::FwUpgradeDialog(), retranslateUi(), and setupUi().

12.88.3.4 QLabel* Ui_FwUpgradeDialog::devLabel

Definition at line 40 of file ui_fwupgradedialog.h.

Referenced by FwUpgradeDialog::FwUpgradeDialog(), retranslateUi(), and setupUi().

12.88.3.5 QGridLayout* Ui_FwUpgradeDialog::gridLayout

Definition at line 31 of file ui_fwupgradedialog.h.

Referenced by setupUi().

12.88.3.6 QGridLayout* Ui_FwUpgradeDialog::gridLayout_2

Definition at line 29 of file ui_fwupgradedialog.h.

Referenced by setupUi().

12.88.3.7 QSpacerItem* Ui_FwUpgradeDialog::horizontalSpacer

Definition at line 30 of file ui_fwupgradedialog.h.

Referenced by setupUi().

12.88.3.8 QSpacerItem* Ui_FwUpgradeDialog::horizontalSpacer_2

Definition at line 39 of file ui_fwupgradedialog.h.

Referenced by setupUi().

12.88.3.9 QSpacerItem* Ui_FwUpgradeDialog::horizontalSpacer_3

Definition at line 32 of file ui_fwupgradedialog.h.

Referenced by setupUi().

12.88.3.10 QSpacerItem* Ui_FwUpgradeDialog::horizontalSpacer_4

Definition at line 34 of file ui_fwupgradedialog.h.

Referenced by setupUi().

12.88.3.11 QLabel* Ui_FwUpgradeDialog::label_2

Definition at line 35 of file ui_fwupgradedialog.h.

Referenced by retranslateUi(), and setupUi().

12.88.3.12 QLabel* Ui_FwUpgradeDialog::label_4

Definition at line 33 of file ui_fwupgradedialog.h.

Referenced by retranslateUi(), and setupUi().

12.88.3.13 QLabel* Ui_FwUpgradeDialog::newLabel

Definition at line 37 of file ui_fwupgradedialog.h.

Referenced by FwUpgradeDialog::exec(), FwUpgradeDialog::FwUpgradeDialog(), retranslateUi(), and setupUi().

12.88.3.14 QProgressBar* Ui_FwUpgradeDialog::progressBar

Definition at line 41 of file ui_fwupgradedialog.h.

Referenced by FwUpgradeDialog::exec(), FwUpgradeDialog::fwUpdateFinished(), FwUpgradeDialog::fwUpdateProgress(), FwUpgradeDialog::on_actionButton_clicked(), and setupUi().

12.88.3.15 QLabel* Ui_FwUpgradeDialog::statusLabel

Definition at line 38 of file ui_fwupgradedialog.h.

Referenced by FwUpgradeDialog::exec(), FwUpgradeDialog::fwUpdateFinished(), FwUpgradeDialog::on_actionButton_clicked(), retranslateUi(), and setupUi().

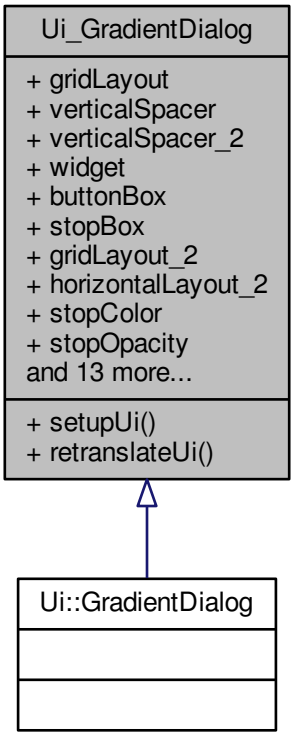
The documentation for this class was generated from the following file:

- src/ckb/ui_fwupgradedialog.h

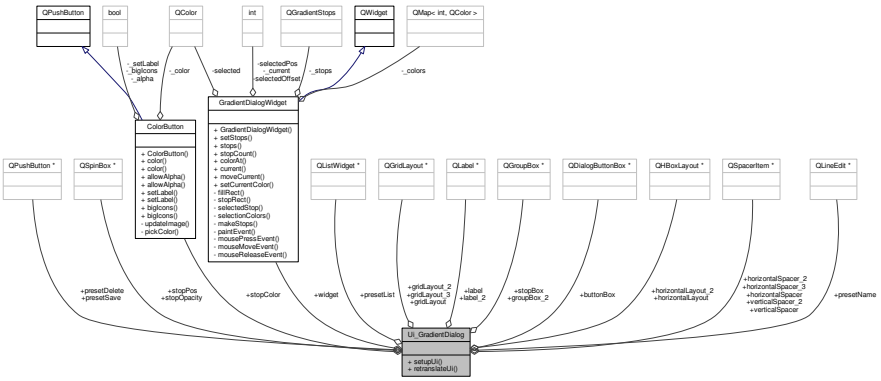
12.89 Ui_GradientDialog Class Reference

`#include <src/ckb/ui_gradientdialog.h>`

Inheritance diagram for Ui_GradientDialog:



Collaboration diagram for Ui_GradientDialog:



Public Member Functions

- void setupUi (QDialog *GradientDialog)
- void retranslateUi (QDialog *GradientDialog)

Data Fields

- QGridLayout * gridLayout
- QSpacerItem * verticalSpacer
- QSpacerItem * verticalSpacer_2
- GradientDialogWidget * widget
- QDialogButtonBox * buttonBox
- QGroupBox * stopBox
- QGridLayout * gridLayout_2
- QHBoxLayout * horizontalLayout_2
- ColorButton * stopColor
- QSpinBox * stopOpacity
- QSpacerItem * horizontalSpacer_2
- QLabel * label
- QHBoxLayout * horizontalLayout
- QSpinBox * stopPos
- QSpacerItem * horizontalSpacer
- QLabel * label_2
- QSpacerItem * horizontalSpacer_3
- QGroupBox * groupBox_2
- QGridLayout * gridLayout_3
- QListWidget * presetList
- QLineEdit * presetName
- QPushButton * presetSave
- QPushButton * presetDelete

12.89.1 Detailed Description

Definition at line 33 of file ui_gradientdialog.h.

12.89.2 Member Function Documentation

12.89.2.1 void Ui_GradientDialog::retranslateUi (QDialog * *GradientDialog*) [inline]

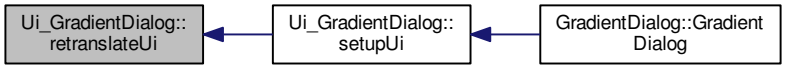
Definition at line 213 of file ui_gradientdialog.h.

References groupBox_2, label, label_2, presetDelete, presetName, presetSave, stopBox, stopColor, stopOpacity, and stopPos.

Referenced by setupUi().

```
214     {
215         GradientDialog->setWindowTitle(QApplication::translate("GradientDialog", "Pick Gradient", Q_NULLPTR
    ));
216         stopBox->setTitle(QApplication::translate("GradientDialog", "Point X of Y", Q_NULLPTR));
217         stopColor->setText(QApplication::translate("GradientDialog", "Pick Color...", Q_NULLPTR));
218         stopOpacity->setSuffix(QApplication::translate("GradientDialog", "%", Q_NULLPTR));
219         label->setText(QApplication::translate("GradientDialog", "Position:", Q_NULLPTR));
220         stopPos->setSuffix(QApplication::translate("GradientDialog", "%", Q_NULLPTR));
221         label_2->setText(QApplication::translate("GradientDialog", "Color:", Q_NULLPTR));
222         groupBox_2->setTitle(QApplication::translate("GradientDialog", "Presets", Q_NULLPTR));
223         presetName->setPlaceholderText(QApplication::translate("GradientDialog", "Name",
    Q_NULLPTR));
224         presetSave->setText(QApplication::translate("GradientDialog", "Save", Q_NULLPTR));
225         presetDelete->setText(QApplication::translate("GradientDialog", "Delete", Q_NULLPTR));
226     } // retranslateUi
```

Here is the caller graph for this function:



12.89.2.2 void Ui_GradientDialog::setupUi (QDialog * *GradientDialog*) [inline]

Definition at line 60 of file ui_gradientdialog.h.

References buttonBox, gridLayout, gridLayout_2, gridLayout_3, groupBox_2, horizontalLayout, horizontalLayout_2, horizontalSpacer, horizontalSpacer_2, horizontalSpacer_3, label, label_2, presetDelete, presetList, presetName, presetSave, retranslateUi(), stopBox, stopColor, stopOpacity, stopPos, verticalSpacer, verticalSpacer_2, and widget.

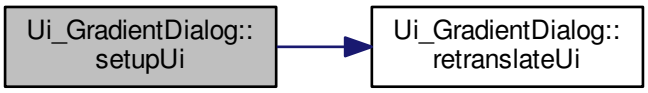
Referenced by GradientDialog::GradientDialog().

```
61     {
62         if (GradientDialog->objectName().isEmpty())
63             GradientDialog->setObjectName(QStringLiteral("GradientDialog"));
64         GradientDialog->resize(400, 440);
65         GradientDialog->setMinimumSize(QSize(400, 440));
66         GradientDialog->setMaximumSize(QSize(400, 440));
67         GradientDialog->setFocusPolicy(Qt::ClickFocus);
68         gridLayout = new QGridLayout(GradientDialog);
69         gridLayout->setObjectName(QStringLiteral("gridLayout"));
70         verticalSpacer = new QSpacerItem(0, 0, QSizePolicy::Minimum, QSizePolicy::Expanding);
71 
72         gridLayout->addItem(verticalSpacer, 1, 0, 1, 1);
73 
74         verticalSpacer_2 = new QSpacerItem(0, 15, QSizePolicy::Minimum, QSizePolicy::Fixed)
    ;
75 
76         gridLayout->addItem(verticalSpacer_2, 3, 0, 1, 1);
77 
78         widget = new GradientDialogWidget(GradientDialog);
79         widget->setObjectName(QStringLiteral("widget"));
80         QSizePolicy sizePolicy(QSizePolicy::Preferred, QSizePolicy::Fixed);
81         sizePolicy.setHorizontalStretch(0);
82         sizePolicy.setVerticalStretch(0);
83         sizePolicy.setHeightForWidth(widget->sizePolicy().hasHeightForWidth());
84         widget->setSizePolicy(sizePolicy);
85         widget->setMinimumSize(QSize(0, 74));
86         widget->setMaximumSize(QSize(16777215, 74));
87 
88         gridLayout->addWidget(widget, 2, 0, 1, 1);
89 
90         buttonBox = new QDialogButtonBox(GradientDialog);
91         buttonBox->setObjectName(QStringLiteral("buttonBox"));
92         buttonBox->setOrientation(Qt::Horizontal);
93         buttonBox->setStandardButtons(QDialogButtonBox::Cancel|QDialogButtonBox::Ok);
94 
95         gridLayout->addWidget(buttonBox, 5, 0, 1, 1);
96 
97         stopBox = new QGroupBox(GradientDialog);
98         stopBox->setObjectName(QStringLiteral("stopBox"));
99         sizePolicy.setHeightForWidth(stopBox->sizePolicy().hasHeightForWidth());
100         stopBox->setSizePolicy(sizePolicy);
101         gridLayout_2 = new QGridLayout(stopBox);
102         gridLayout_2->setObjectName(QStringLiteral("gridLayout_2"));
103         horizontalLayout_2 = new QHBoxLayout();
104         horizontalLayout_2->setObjectName(QStringLiteral("horizontalLayout_2"));
105         stopColor = new ColorButton(stopBox);
106         stopColor->setObjectName(QStringLiteral("stopColor"));
107 
108         horizontalLayout_2->addWidget(stopColor);
109 
110         stopOpacity = new QSpinBox(stopBox);
111         stopOpacity->setObjectName(QStringLiteral("stopOpacity"));
112         stopOpacity->setMaximum(100);
113 
```

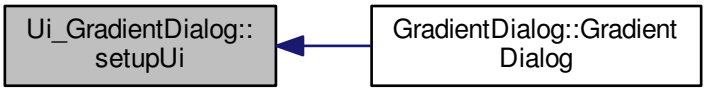
```
114         horizontalLayout_2->addWidget(stopOpacity);
115 
116         horizontalSpacer_2 = new QSpacerItem(40, 20, QSizePolicy::Expanding,
    QSizePolicy::Minimum);
117 
118         horizontalLayout_2->addItem(horizontalSpacer_2);
119 
120 
121         gridLayout_2->addLayout(horizontalLayout_2, 1, 2, 1, 1);
122 
123         label = new QLabel(stopBox);
124         label->setObjectName(QStringLiteral("label"));
125 
126         gridLayout_2->addWidget(label, 0, 0, 1, 1);
127 
128         horizontalLayout = new QHBoxLayout();
129         horizontalLayout->setObjectName(QStringLiteral("horizontalLayout"));
130         stopPos = new QSpinBox(stopBox);
131         stopPos->setObjectName(QStringLiteral("stopPos"));
132         stopPos->setMaximum(100);
133 
134         horizontalLayout->addWidget(stopPos);
135 
136         horizontalSpacer = new QSpacerItem(40, 20, QSizePolicy::Expanding,
    QSizePolicy::Minimum);
137 
138         horizontalLayout->addItem(horizontalSpacer);
139 
140 
141         gridLayout_2->addLayout(horizontalLayout, 0, 2, 1, 1);
142 
143         label_2 = new QLabel(stopBox);
144         label_2->setObjectName(QStringLiteral("label_2"));
145 
146         gridLayout_2->addWidget(label_2, 1, 0, 1, 1);
147 
148         horizontalSpacer_3 = new QSpacerItem(10, 20, QSizePolicy::Fixed,
    QSizePolicy::Minimum);
149 
150         gridLayout_2->addItem(horizontalSpacer_3, 0, 1, 1, 1);
151 
152 
153         gridLayout->addWidget(stopBox, 4, 0, 1, 1);
154 
155         groupBox_2 = new QGroupBox(GradientDialog);
156         groupBox_2->setObjectName(QStringLiteral("groupBox_2"));
157         sizePolicy.setHeightForWidth(groupBox_2->sizePolicy().hasHeightForWidth());
158         groupBox_2->setSizePolicy(sizePolicy);
159         gridLayout_3 = new QGridLayout(groupBox_2);
160         gridLayout_3->setObjectName(QStringLiteral("gridLayout_3"));
161         presetList = new QListWidget(groupBox_2);
162         presetList->setObjectName(QStringLiteral("presetList"));
163         QSizePolicy sizePolicy1(QSizePolicy::Expanding, QSizePolicy::Preferred);
164         sizePolicy1.setHorizontalStretch(0);
165         sizePolicy1.setVerticalStretch(0);
166         sizePolicy1.setHeightForWidth(presetList->sizePolicy().hasHeightForWidth());
167         presetList->setSizePolicy(sizePolicy1);
168         presetList->setFrameShadow(QFrame::Plain);
169         presetList->setHorizontalScrollBarPolicy(Qt::ScrollBarAlwaysOn);
170         presetList->setIconSize(QSize(64, 64));
171         presetList->setHorizontalScrollMode(QAbstractItemView::ScrollPerPixel);
172         presetList->setFlow(QListView::TopToBottom);
173         presetList->setViewMode(QListView::IconMode);
174         presetList->setUniformItemSizes(true);
175 
176         gridLayout_3->addWidget(presetList, 0, 0, 3, 1);
177 
178         presetName = new QLineEdit(groupBox_2);
179         presetName->setObjectName(QStringLiteral("presetName"));
180         presetName->setMaxLength(30);
181 
182         gridLayout_3->addWidget(presetName, 0, 1, 1, 1);
183 
184         presetSave = new QPushButton(groupBox_2);
185         presetSave->setObjectName(QStringLiteral("presetSave"));
186         presetSave->setAutoDefault(false);
187 
188         gridLayout_3->addWidget(presetSave, 1, 1, 1, 1);
189 
190         presetDelete = new QPushButton(groupBox_2);
191         presetDelete->setObjectName(QStringLiteral("presetDelete"));
192         presetDelete->setAutoDefault(false);
193 
194         gridLayout_3->addWidget(presetDelete, 2, 1, 1, 1);
195 
196 
197         gridLayout->addWidget(groupBox_2, 0, 0, 1, 1);
```

```
198
199         QWidget::setTabOrder(presetList, presetName);
200         QWidget::setTabOrder(presetName, presetSave);
201         QWidget::setTabOrder(presetSave, presetDelete);
202         QWidget::setTabOrder(presetDelete, stopPos);
203         QWidget::setTabOrder(stopPos, stopColor);
204         QWidget::setTabOrder(stopColor, stopOpacity);
205
206         retranslateUi(GradientDialog);
207         QObject::connect(buttonBox, SIGNAL(accepted()), GradientDialog, SLOT(accept()));
208         QObject::connect(buttonBox, SIGNAL(rejected()), GradientDialog, SLOT(reject()));
209
210         QMetaObject::connectSlotsByName(GradientDialog);
211     } // setupUi
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.89.3 Field Documentation

12.89.3.1 QDialogButtonBox* Ui_GradientDialog::buttonBox

Definition at line 40 of file ui_gradientdialog.h.

Referenced by setupUi().

12.89.3.2 QGridLayout* Ui_GradientDialog::gridLayout

Definition at line 36 of file ui_gradientdialog.h.

Referenced by setupUi().

12.89.3.3 QGridLayout* Ui_GradientDialog::gridLayout_2

Definition at line 42 of file ui_gradientdialog.h.

Referenced by setupUi().

12.89.3.4 QGridLayout* Ui_GradientDialog::gridLayout_3

Definition at line 54 of file ui_gradientdialog.h.

Referenced by setupUi().

12.89.3.5 QGroupBox* Ui_GradientDialog::groupBox_2

Definition at line 53 of file ui_gradientdialog.h.

Referenced by retranslateUi(), and setupUi().

12.89.3.6 QHBoxLayout* Ui_GradientDialog::horizontalLayout

Definition at line 48 of file ui_gradientdialog.h.

Referenced by setupUi().

12.89.3.7 QHBoxLayout* Ui_GradientDialog::horizontalLayout_2

Definition at line 43 of file ui_gradientdialog.h.

Referenced by setupUi().

12.89.3.8 QSpacerItem* Ui_GradientDialog::horizontalSpacer

Definition at line 50 of file ui_gradientdialog.h.

Referenced by setupUi().

12.89.3.9 QSpacerItem* Ui_GradientDialog::horizontalSpacer_2

Definition at line 46 of file ui_gradientdialog.h.

Referenced by setupUi().

12.89.3.10 QSpacerItem* Ui_GradientDialog::horizontalSpacer_3

Definition at line 52 of file ui_gradientdialog.h.

Referenced by setupUi().

12.89.3.11 QLabel* Ui_GradientDialog::label

Definition at line 47 of file ui_gradientdialog.h.

Referenced by retranslateUi(), and setupUi().

12.89.3.12 QLabel* Ui_GradientDialog::label_2

Definition at line 51 of file ui_gradientdialog.h.

Referenced by retranslateUi(), and setupUi().

12.89.3.13 QPushButton* Ui_GradientDialog::presetDelete

Definition at line 58 of file ui_gradientdialog.h.

Referenced by GradientDialog::getGradient(), GradientDialog::on_presetDelete_clicked(), GradientDialog::on_presetSave_clicked(), retranslateUi(), GradientDialog::setPreset(), and setupUi().

12.89.3.14 QListWidget* Ui_GradientDialog::presetList

Definition at line 55 of file ui_gradientdialog.h.

Referenced by GradientDialog::setPreset(), setupUi(), and GradientDialog::updatePresets().

12.89.3.15 QLineEdit* Ui_GradientDialog::presetName

Definition at line 56 of file ui_gradientdialog.h.

Referenced by GradientDialog::on_presetSave_clicked(), retranslateUi(), GradientDialog::setPreset(), and setupUi().

12.89.3.16 QPushButton* Ui_GradientDialog::presetSave

Definition at line 57 of file ui_gradientdialog.h.

Referenced by GradientDialog::getGradient(), GradientDialog::on_presetDelete_clicked(), GradientDialog::on_presetSave_clicked(), retranslateUi(), GradientDialog::setPreset(), and setupUi().

12.89.3.17 QGroupBox* Ui_GradientDialog::stopBox

Definition at line 41 of file ui_gradientdialog.h.

Referenced by GradientDialog::currentChanged(), retranslateUi(), and setupUi().

12.89.3.18 ColorButton* Ui_GradientDialog::stopColor

Definition at line 44 of file ui_gradientdialog.h.

Referenced by GradientDialog::currentChanged(), GradientDialog::GradientDialog(), GradientDialog::on_stopOpacity_valueChanged(), retranslateUi(), and setupUi().

12.89.3.19 QSpinBox* Ui_GradientDialog::stopOpacity

Definition at line 45 of file ui_gradientdialog.h.

Referenced by GradientDialog::colorChanged(), GradientDialog::currentChanged(), retranslateUi(), and setupUi().

12.89.3.20 QSpinBox* Ui_GradientDialog::stopPos

Definition at line 49 of file ui_gradientdialog.h.

Referenced by GradientDialog::currentChanged(), GradientDialog::on_stopPos_valueChanged(), retranslateUi(), and setupUi().

12.89.3.21 QSpacerItem* Ui_GradientDialog::verticalSpacer

Definition at line 37 of file ui_gradientdialog.h.

Referenced by setupUi().

12.89.3.22 QSpacerItem* Ui_GradientDialog::verticalSpacer_2

Definition at line 38 of file ui_gradientdialog.h.

Referenced by setupUi().

12.89.3.23 GradientDialogWidget* Ui_GradientDialog::widget

Definition at line 39 of file ui_gradientdialog.h.

Referenced by GradientDialog::colorChanged(), GradientDialog::currentChanged(), GradientDialog::getGradient(), GradientDialog::GradientDialog(), GradientDialog::on_presetSave_clicked(), GradientDialog::on_stopPos_valueChanged(), GradientDialog::setPreset(), and setupUi().

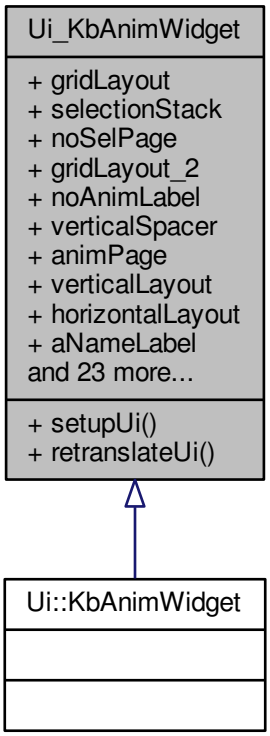
The documentation for this class was generated from the following file:

- src/ckb/ui_gradientdialog.h

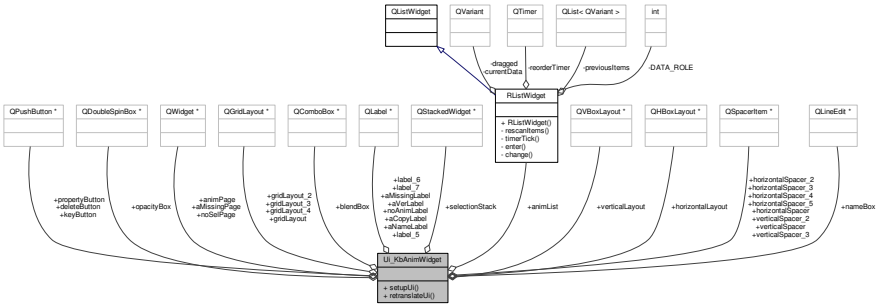
12.90 Ui_KbAnimWidget Class Reference

`#include <src/ckb/ui_kbanimwidget.h>`

Inheritance diagram for Ui_KbAnimWidget:



Collaboration diagram for Ui_KbAnimWidget:



Public Member Functions

- void setupUi (QWidget *KbAnimWidget)
- void retranslateUi (QWidget *KbAnimWidget)

Data Fields

- QGridLayout * gridLayout
- QStackedWidget * selectionStack
- QWidget * noSelPage
- QGridLayout * gridLayout_2
- QLabel * noAnimLabel
- QSpacerItem * verticalSpacer
- QWidget * animPage
- QVBoxLayout * verticalLayout
- QHBoxLayout * horizontalLayout
- QLabel * aNameLabel
- QLabel * aVerLabel
- QSpacerItem * horizontalSpacer
- QLabel * aCopyLabel
- QSpacerItem * verticalSpacer_3
- QGridLayout * gridLayout_3
- QLineEdit * nameBox
- QDoubleSpinBox * opacityBox
- QComboBox * blendBox
- QLabel * label_6
- QSpacerItem * horizontalSpacer_4
- QSpacerItem * horizontalSpacer_2
- QLabel * label_5
- QSpacerItem * horizontalSpacer_3
- QLabel * label_7
- QPushButton * keyButton
- QPushButton * deleteButton
- QPushButton * propertyButton
- QSpacerItem * horizontalSpacer_5
- QSpacerItem * verticalSpacer_2
- QWidget * aMissingPage
- QGridLayout * gridLayout_4
- QLabel * aMissingLabel
- RListWidget * animList

12.90.1 Detailed Description

Definition at line 32 of file ui_kbanimwidget.h.

12.90.2 Member Function Documentation

12.90.2.1 void Ui_KbAnimWidget::retranslateUi (QWidget * *KbAnimWidget*) [inline]

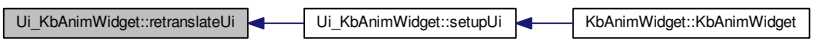
Definition at line 241 of file ui_kbanimwidget.h.

References aCopyLabel, aMissingLabel, aNameLabel, aVerLabel, blendBox, deleteButton, keyButton, label_5, label_6, label_7, noAnimLabel, opacityBox, and propertyButton.

Referenced by setupUi().

```
242     {
243         KbAnimWidget->setWindowTitle(QApplication::translate("KbAnimWidget", "Form", Q_NULLPTR));
244         noAnimLabel->setText(QApplication::translate("KbAnimWidget", "No animations in this mode
    ", Q_NULLPTR));
245         aNameLabel->setText(QApplication::translate("KbAnimWidget", "Animation name", Q_NULLPTR))
    ;
246         aVerLabel->setText(QApplication::translate("KbAnimWidget", "v0.0", Q_NULLPTR));
247         aCopyLabel->setText(QApplication::translate("KbAnimWidget", "Copyright \302\251",
    Q_NULLPTR));
248         opacityBox->setSuffix(QApplication::translate("KbAnimWidget", "%", Q_NULLPTR));
249         blendBox->clear();
250         blendBox->insertItems(0, QStringList()
251          << QApplication::translate("KbAnimWidget", "Normal", Q_NULLPTR)
252          << QApplication::translate("KbAnimWidget", "Add", Q_NULLPTR)
253          << QApplication::translate("KbAnimWidget", "Subtract", Q_NULLPTR)
254          << QApplication::translate("KbAnimWidget", "Multiply", Q_NULLPTR)
255          << QApplication::translate("KbAnimWidget", "Divide", Q_NULLPTR)
256         );
257         label_6->setText(QApplication::translate("KbAnimWidget", "Blend mode:", Q_NULLPTR));
258         label_5->setText(QApplication::translate("KbAnimWidget", "Opacity:", Q_NULLPTR));
259         label_7->setText(QApplication::translate("KbAnimWidget", "Name:", Q_NULLPTR));
260         keyButton->setText(QApplication::translate("KbAnimWidget", "Set keys", Q_NULLPTR));
261         deleteButton->setText(QApplication::translate("KbAnimWidget", "Delete animation",
    Q_NULLPTR));
262         propertyButton->setText(QApplication::translate("KbAnimWidget", "Edit properties",
    Q_NULLPTR));
263         aMissingLabel->setText(QApplication::translate("KbAnimWidget", "Mising animation",
    Q_NULLPTR));
264     } // retranslateUi
```

Here is the caller graph for this function:



12.90.2.2 void Ui_KbAnimWidget::setupUi (QWidget * *KbAnimWidget*) [inline]

Definition at line 69 of file ui_kbanimwidget.h.

References aCopyLabel, aMissingLabel, aMissingPage, aNameLabel, animList, animPage, aVerLabel, blendBox, deleteButton, gridLayout, gridLayout_2, gridLayout_3, gridLayout_4, horizontalLayout, horizontalSpacer, horizontalSpacer_2, horizontalSpacer_3, horizontalSpacer_4, horizontalSpacer_5, keyButton, label_5, label_6, label_7, nameBox, noAnimLabel, noSelPage, opacityBox, propertyButton, retranslateUi(), selectionStack, verticalLayout, verticalSpacer, verticalSpacer_2, and verticalSpacer_3.

Referenced by KbAnimWidget::KbAnimWidget().

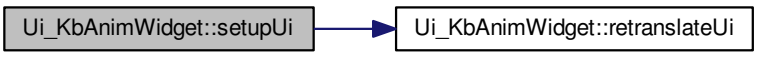
```
70     {
```

```
71          if (KbAnimWidget->objectName().isEmpty())
72              KbAnimWidget->setObjectName(QStringLiteral("KbAnimWidget"));
73          KbAnimWidget->resize(613, 300);
74          gridLayout = new QGridLayout(KbAnimWidget);
75          gridLayout->setObjectName(QStringLiteral("gridLayout"));
76          gridLayout->setContentsMargins(0, 0, 0, 0);
77          selectionStack = new QStackedWidget(KbAnimWidget);
78          selectionStack->setObjectName(QStringLiteral("selectionStack"));
79          noSelPage = new QWidget();
80          noSelPage->setObjectName(QStringLiteral("noSelPage"));
81          gridLayout_2 = new QGridLayout(noSelPage);
82          gridLayout_2->setObjectName(QStringLiteral("gridLayout_2"));
83          gridLayout_2->setContentsMargins(6, 6, 6, 6);
84          noAnimLabel = new QLabel(noSelPage);
85          noAnimLabel->setObjectName(QStringLiteral("noAnimLabel"));
86
87          gridLayout_2->addWidget(noAnimLabel, 0, 0, 1, 1);
88
89          verticalSpacer = new QSpacerItem(20, 40, QSizePolicy::Minimum, QSizePolicy::Expanding
    );
90
91          gridLayout_2->addItem(verticalSpacer, 1, 0, 1, 1);
92
93          selectionStack->addWidget(noSelPage);
94          animPage = new QWidget();
95          animPage->setObjectName(QStringLiteral("animPage"));
96          verticalLayout = new QVBoxLayout(animPage);
97          verticalLayout->setObjectName(QStringLiteral("verticalLayout"));
98          verticalLayout->setContentsMargins(6, 0, 0, 0);
99          horizontalLayout = new QHBoxLayout();
100         horizontalLayout->setObjectName(QStringLiteral("horizontalLayout"));
101         aNameLabel = new QLabel(animPage);
102         aNameLabel->setObjectName(QStringLiteral("aNameLabel"));
103         QFont font;
104         font.setBold(true);
105         font.setWeight(75);
106         aNameLabel->setFont(font);
107
108         horizontalLayout->addWidget(aNameLabel);
109
110         aVerLabel = new QLabel(animPage);
111         aVerLabel->setObjectName(QStringLiteral("aVerLabel"));
112
113         horizontalLayout->addWidget(aVerLabel);
114
115         horizontalSpacer = new QSpacerItem(40, 20, QSizePolicy::Expanding,
    QSizePolicy::Minimum);
116
117         horizontalLayout->addItem(horizontalSpacer);
118
119         aCopyLabel = new QLabel(animPage);
120         aCopyLabel->setObjectName(QStringLiteral("aCopyLabel"));
121
122         horizontalLayout->addWidget(aCopyLabel);
123
124
125         verticalLayout->addLayout(horizontalLayout);
126
127         verticalSpacer_3 = new QSpacerItem(20, 40, QSizePolicy::Minimum,
    QSizePolicy::Expanding);
128
129         verticalLayout->addItem(verticalSpacer_3);
130
131         gridLayout_3 = new QGridLayout();
132         gridLayout_3->setObjectName(QStringLiteral("gridLayout_3"));
133         nameBox = new QLineEdit(animPage);
134         nameBox->setObjectName(QStringLiteral("nameBox"));
135         QSizePolicy sizePolicy(QSizePolicy::Fixed, QSizePolicy::Fixed);
136         sizePolicy.setHorizontalStretch(0);
137         sizePolicy.setVerticalStretch(0);
138         sizePolicy.setHeightForWidth(nameBox->sizePolicy().hasHeightForWidth());
139         nameBox->setSizePolicy(sizePolicy);
140         nameBox->setMinimumSize(QSize(150, 0));
141         nameBox->setMaximumSize(QSize(150, 16777215));
142
143         gridLayout_3->addWidget(nameBox, 0, 3, 1, 2);
144
145         opacityBox = new QDoubleSpinBox(animPage);
146         opacityBox->setObjectName(QStringLiteral("opacityBox"));
147         opacityBox->setDecimals(1);
148         opacityBox->setValue(100);
149
150         gridLayout_3->addWidget(opacityBox, 1, 3, 1, 1);
151
152         blendBox = new QComboBox(animPage);
153         blendBox->setObjectName(QStringLiteral("blendBox"));
154
```

```
155         gridLayout_3->addWidget(blendBox, 2, 3, 1, 2);
156 
157         label_6 = new QLabel(animPage);
158         label_6->setObjectName(QStringLiteral("label_6"));
159 
160         gridLayout_3->addWidget(label_6, 2, 1, 1, 1);
161 
162         horizontalSpacer_4 = new QSpacerItem(0, 0, QSizePolicy::Fixed,
    QSizePolicy::Minimum);
163 
164         gridLayout_3->addItem(horizontalSpacer_4, 0, 0, 1, 1);
165 
166         horizontalSpacer_2 = new QSpacerItem(30, 20, QSizePolicy::Fixed,
    QSizePolicy::Minimum);
167 
168         gridLayout_3->addItem(horizontalSpacer_2, 0, 2, 1, 1);
169 
170         label_5 = new QLabel(animPage);
171         label_5->setObjectName(QStringLiteral("label_5"));
172 
173         gridLayout_3->addWidget(label_5, 1, 1, 1, 1);
174 
175         horizontalSpacer_3 = new QSpacerItem(40, 20, QSizePolicy::Expanding,
    QSizePolicy::Minimum);
176 
177         gridLayout_3->addItem(horizontalSpacer_3, 0, 5, 1, 1);
178 
179         label_7 = new QLabel(animPage);
180         label_7->setObjectName(QStringLiteral("label_7"));
181 
182         gridLayout_3->addWidget(label_7, 0, 1, 1, 1);
183 
184         keyButton = new QPushButton(animPage);
185         keyButton->setObjectName(QStringLiteral("keyButton"));
186 
187         gridLayout_3->addWidget(keyButton, 1, 6, 1, 1);
188 
189         deleteButton = new QPushButton(animPage);
190         deleteButton->setObjectName(QStringLiteral("deleteButton"));
191 
192         gridLayout_3->addWidget(deleteButton, 2, 6, 1, 1);
193 
194         propertyButton = new QPushButton(animPage);
195         propertyButton->setObjectName(QStringLiteral("propertyButton"));
196 
197         gridLayout_3->addWidget(propertyButton, 0, 6, 1, 1);
198 
199         horizontalSpacer_5 = new QSpacerItem(40, 20, QSizePolicy::Expanding,
    QSizePolicy::Minimum);
200 
201         gridLayout_3->addItem(horizontalSpacer_5, 1, 4, 1, 1);
202 
203 
204         verticalLayout->addLayout(gridLayout_3);
205 
206         verticalSpacer_2 = new QSpacerItem(20, 40, QSizePolicy::Minimum,
    QSizePolicy::Expanding);
207 
208         verticalLayout->addItem(verticalSpacer_2);
209 
210         selectionStack->addWidget(animPage);
211         aMissingPage = new QWidget();
212         aMissingPage->setObjectName(QStringLiteral("aMissingPage"));
213         gridLayout_4 = new QGridLayout(aMissingPage);
214         gridLayout_4->setObjectName(QStringLiteral("gridLayout_4"));
215         aMissingLabel = new QLabel(aMissingPage);
216         aMissingLabel->setObjectName(QStringLiteral("aMissingLabel"));
217         aMissingLabel->setFont(font);
218 
219         gridLayout_4->addWidget(aMissingLabel, 0, 0, 1, 1);
220 
221         selectionStack->addWidget(aMissingPage);
222 
223         gridLayout->addWidget(selectionStack, 0, 2, 2, 1);
224 
225         animList = new RListWidget(KbAnimWidget);
226         animList->setObjectName(QStringLiteral("animList"));
227         animList->setMaximumSize(QSize(130, 16777215));
228         animList->setContextMenuPolicy(Qt::CustomContextMenu);
229         animList->setFrameShadow(QFrame::Plain);
230         animList->setDragDropMode(QAbstractItemView::InternalMove);
231         animList->setMovement(QListView::Snap);
232 
233         gridLayout->addWidget(animList, 0, 0, 2, 1);
234 
235 
236         retranslateUi(KbAnimWidget);
```

```
237
238         QMetaObject::connectSlotsByName(KbAnimWidget);
239     } // setupUi
```

Here is the call graph for this function:



Here is the caller graph for this function:

12.90.3 Field Documentation

12.90.3.1 QLabel* Ui_KbAnimWidget::aCopyLabel

Definition at line 47 of file ui_kbanimwidget.h.

Referenced by retranslateUi(), KbAnimWidget::setCurrent(), and setupUi().

12.90.3.2 QLabel* Ui_KbAnimWidget::aMissingLabel

Definition at line 66 of file ui_kbanimwidget.h.

Referenced by retranslateUi(), KbAnimWidget::setCurrent(), and setupUi().

12.90.3.3 QWidget* Ui_KbAnimWidget::aMissingPage

Definition at line 64 of file ui_kbanimwidget.h.

Referenced by setupUi().

12.90.3.4 QLabel* Ui_KbAnimWidget::aNameLabel

Definition at line 44 of file ui_kbanimwidget.h.

Referenced by retranslateUi(), KbAnimWidget::setCurrent(), and setupUi().

12.90.3.5 RListWidget* Ui_KbAnimWidget::animList

Definition at line 67 of file ui_kbanimwidget.h.

Referenced by KbAnimWidget::addAnim(), KbAnimWidget::clearSelection(), KbAnimWidget::duplicateAnim(), KbAnimWidget::KbAnimWidget(), KbAnimWidget::on_animList_customContextMenuRequested(), KbAnimWidget::on_deleteButton_clicked(), KbAnimWidget::on_nameBox_textEdited(), KbAnimWidget::refreshList(), KbAnimWidget::reorderAnims(), and setupUi().

12.90.3.6 QWidget* Ui_KbAnimWidget::animPage

Definition at line 41 of file ui_kbanimwidget.h.

Referenced by setupUi().

12.90.3.7 QLabel* Ui_KbAnimWidget::aVerLabel

Definition at line 45 of file ui_kbanimwidget.h.

Referenced by retranslateUi(), KbAnimWidget::setCurrent(), and setupUi().

12.90.3.8 QComboBox* Ui_KbAnimWidget::blendBox

Definition at line 52 of file ui_kbanimwidget.h.

Referenced by retranslateUi(), KbAnimWidget::setCurrent(), and setupUi().

12.90.3.9 QPushButton* Ui_KbAnimWidget::deleteButton

Definition at line 60 of file ui_kbanimwidget.h.

Referenced by retranslateUi(), and setupUi().

12.90.3.10 QGridLayout* Ui_KbAnimWidget::gridLayout

Definition at line 35 of file ui_kbanimwidget.h.

Referenced by setupUi().

12.90.3.11 QGridLayout* Ui_KbAnimWidget::gridLayout_2

Definition at line 38 of file ui_kbanimwidget.h.

Referenced by setupUi().

12.90.3.12 QGridLayout* Ui_KbAnimWidget::gridLayout_3

Definition at line 49 of file ui_kbanimwidget.h.

Referenced by setupUi().

12.90.3.13 QGridLayout* Ui_KbAnimWidget::gridLayout_4

Definition at line 65 of file ui_kbanimwidget.h.

Referenced by setupUi().

12.90.3.14 QHBoxLayout* Ui_KbAnimWidget::horizontalLayout

Definition at line 43 of file ui_kbanimwidget.h.

Referenced by setupUi().

12.90.3.15 QSpacerItem* Ui_KbAnimWidget::horizontalSpacer

Definition at line 46 of file ui_kbanimwidget.h.

Referenced by setupUi().

12.90.3.16 QSpacerItem* Ui_KbAnimWidget::horizontalSpacer_2

Definition at line 55 of file ui_kbanimwidget.h.

Referenced by setupUi().

12.90.3.17 QSpacerItem* Ui_KbAnimWidget::horizontalSpacer_3

Definition at line 57 of file ui_kbanimwidget.h.

Referenced by setupUi().

12.90.3.18 QSpacerItem* Ui_KbAnimWidget::horizontalSpacer_4

Definition at line 54 of file ui_kbanimwidget.h.

Referenced by setupUi().

12.90.3.19 QSpacerItem* Ui_KbAnimWidget::horizontalSpacer_5

Definition at line 62 of file ui_kbanimwidget.h.

Referenced by setupUi().

12.90.3.20 QPushButton* Ui_KbAnimWidget::keyButton

Definition at line 59 of file ui_kbanimwidget.h.

Referenced by retranslateUi(), KbAnimWidget::setSelectedKeys(), and setupUi().

12.90.3.21 QLabel* Ui_KbAnimWidget::label_5

Definition at line 56 of file ui_kbanimwidget.h.

Referenced by retranslateUi(), and setupUi().

12.90.3.22 QLabel* Ui_KbAnimWidget::label_6

Definition at line 53 of file ui_kbanimwidget.h.

Referenced by retranslateUi(), and setupUi().

12.90.3.23 QLabel* Ui_KbAnimWidget::label_7

Definition at line 58 of file ui_kbanimwidget.h.

Referenced by retranslateUi(), and setupUi().

12.90.3.24 QLineEdit* Ui_KbAnimWidget::nameBox

Definition at line 50 of file ui_kbanimwidget.h.

Referenced by KbAnimWidget::on_animList_itemChanged(), KbAnimWidget::on_propertyButton_clicked(), KbAnimWidget::setCurrent(), and setupUi().

12.90.3.25 QLabel* Ui_KbAnimWidget::noAnimLabel

Definition at line 39 of file ui_kbanimwidget.h.

Referenced by KbAnimWidget::addAnim(), KbAnimWidget::duplicateAnim(), KbAnimWidget::on_deleteButton_clicked(), KbAnimWidget::refreshList(), retranslateUi(), and setupUi().

12.90.3.26 QWidget* Ui_KbAnimWidget::noSelPage

Definition at line 37 of file ui_kbanimwidget.h.

Referenced by setupUi().

12.90.3.27 QDoubleSpinBox* Ui_KbAnimWidget::opacityBox

Definition at line 51 of file ui_kbanimwidget.h.

Referenced by retranslateUi(), KbAnimWidget::setCurrent(), and setupUi().

12.90.3.28 QPushButton* Ui_KbAnimWidget::propertyButton

Definition at line 61 of file ui_kbanimwidget.h.

Referenced by retranslateUi(), and setupUi().

12.90.3.29 QStackedWidget* Ui_KbAnimWidget::selectionStack

Definition at line 36 of file ui_kbanimwidget.h.

Referenced by KbAnimWidget::setCurrent(), and setupUi().

12.90.3.30 QVBoxLayout* Ui_KbAnimWidget::verticalLayout

Definition at line 42 of file ui_kbanimwidget.h.

Referenced by setupUi().

12.90.3.31 QSpacerItem* Ui_KbAnimWidget::verticalSpacer

Definition at line 40 of file ui_kbanimwidget.h.

Referenced by setupUi().

12.90.3.32 QSpacerItem* Ui_KbAnimWidget::verticalSpacer_2

Definition at line 63 of file ui_kbanimwidget.h.

Referenced by setupUi().

12.90.3.33 QSpacerItem* Ui_KbAnimWidget::verticalSpacer_3

Definition at line 48 of file ui_kbanimwidget.h.

Referenced by setupUi().

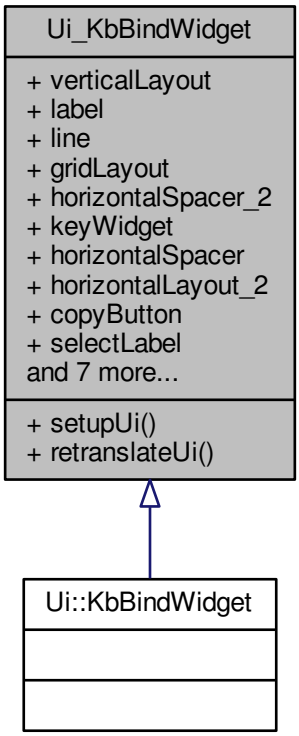
The documentation for this class was generated from the following file:

- src/ckb/ui_kbanimwidget.h

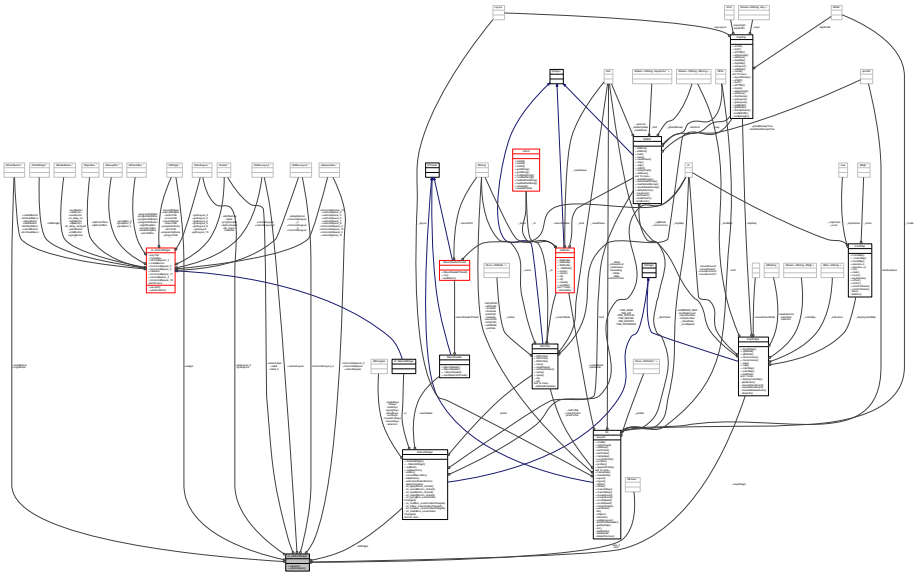
12.91 Ui_KbBindWidget Class Reference

`#include <src/ckb/ui_kbbindwidget.h>`

Inheritance diagram for Ui_KbBindWidget:



Collaboration diagram for Ui_KbBindWidget:



Public Member Functions

- void setupUi (QWidget *KbBindWidget)
- void retranslateUi (QWidget *KbBindWidget)

Data Fields

- QVBoxLayout * verticalLayout
- QLabel * label
- QFrame * line
- QGridLayout * gridLayout
- QSpacerItem * horizontalSpacer_2
- KeyWidget * keyWidget
- QSpacerItem * horizontalSpacer
- QHBoxLayout * horizontalLayout_2
- QPushButton * copyButton
- QLabel * selectLabel
- QPushButton * resetButton
- QSpacerItem * verticalSpacer
- QLabel * label_2
- QFrame * line_2
- QWidget * widget
- QGridLayout * gridLayout_2
- RebindWidget * rbWidget

12.91.1 Detailed Description

Definition at line 30 of file ui_kbbindwidget.h.

12.91.2 Member Function Documentation

12.91.2.1 void Ui_KbBindWidget::retranslateUi (QWidget * *KbBindWidget*) [inline]

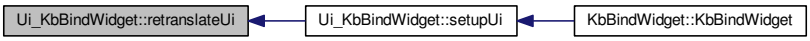
Definition at line 189 of file ui_kbbindwidget.h.

References copyButton, label, label_2, resetButton, and selectLabel.

Referenced by setupUi().

```
190     {
191         KbBindWidget->setWindowTitle(QApplication::translate("KbBindWidget", "Form", Q_NULLPTR));
192         label->setText(QApplication::translate("KbBindWidget", "Keys", Q_NULLPTR));
193         copyButton->setText(QApplication::translate("KbBindWidget", "Copy to mode...", Q_NULLPTR)
    );
194         selectLabel->setText(QApplication::translate("KbBindWidget", "Click to select keys",
    Q_NULLPTR));
195         resetButton->setText(QApplication::translate("KbBindWidget", "Reset", Q_NULLPTR));
196         label_2->setText(QApplication::translate("KbBindWidget", "Binding", Q_NULLPTR));
197     } // retranslateUi
```

Here is the caller graph for this function:



12.91.2.2 void Ui_KbBindWidget::setupUi (QWidget * *KbBindWidget*) [inline]

Definition at line 51 of file ui_kbbindwidget.h.

References copyButton, gridLayout, gridLayout_2, horizontalLayout_2, horizontalSpacer, horizontalSpacer_2, keyWidget, label, label_2, line, line_2, rbWidget, resetButton, retranslateUi(), selectLabel, verticalLayout, verticalSpacer, and widget.

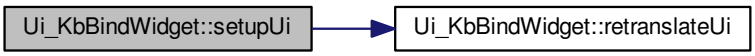
Referenced by KbBindWidget::KbBindWidget().

```
52     {
53         if (KbBindWidget->objectName().isEmpty())
54             KbBindWidget->setObjectName(QStringLiteral("KbBindWidget"));
55         KbBindWidget->resize(581, 473);
56         verticalLayout = new QVBoxLayout(KbBindWidget);
57         verticalLayout->setObjectName(QStringLiteral("verticalLayout"));
58         verticalLayout->setContentsMargins(0, 0, 0, -1);
59         label = new QLabel(KbBindWidget);
60         label->setObjectName(QStringLiteral("label"));
61         QFont font;
62         font.setBold(true);
63         font.setWeight(75);
64         label->setFont(font);
65
66         verticalLayout->addWidget(label);
67
68         line = new QFrame(KbBindWidget);
69         line->setObjectName(QStringLiteral("line"));
70         line->setFrameShape(QFrame::HLine);
71         line->setFrameShadow(QFrame::Sunken);
72
73         verticalLayout->addWidget(line);
74
75         gridLayout = new QGridLayout();
76         gridLayout->setObjectName(QStringLiteral("gridLayout"));
77         horizontalSpacer_2 = new QSpacerItem(40, 20, QSizePolicy::Expanding,
    QSizePolicy::Minimum);
78
79         gridLayout->addItem(horizontalSpacer_2, 0, 4, 1, 1);
80
81         keyWidget = new KeyWidget(KbBindWidget);
82         keyWidget->setObjectName(QStringLiteral("keyWidget"));
```

```
83          QSizePolicy sizePolicy(QSizePolicy::Preferred, QSizePolicy::Preferred);
84          sizePolicy.setHorizontalStretch(0);
85          sizePolicy.setVerticalStretch(0);
86          sizePolicy.setHeightForWidth(keyWidget->sizePolicy().hasHeightForWidth());
87          keyWidget->setSizePolicy(sizePolicy);
88
89          gridLayout->addWidget(keyWidget, 0, 1, 1, 3);
90
91          horizontalSpacer = new QSpacerItem(40, 20, QSizePolicy::Expanding,
    QSizePolicy::Minimum);
92
93          gridLayout->addItem(horizontalSpacer, 0, 0, 1, 1);
94
95          horizontalLayout_2 = new QHBoxLayout();
96          horizontalLayout_2->setSpacing(0);
97          horizontalLayout_2->setObjectName(QStringLiteral("horizontalLayout_2"));
98          copyButton = new QPushButton(KbBindWidget);
99          copyButton->setObjectName(QStringLiteral("copyButton"));
100          QSizePolicy sizePolicy1(QSizePolicy::Fixed, QSizePolicy::Fixed);
101          sizePolicy1.setHorizontalStretch(0);
102          sizePolicy1.setVerticalStretch(0);
103          sizePolicy1.setHeightForWidth(copyButton->sizePolicy().hasHeightForWidth());
104          copyButton->setSizePolicy(sizePolicy1);
105          copyButton->setMinimumSize(QSize(140, 0));
106          copyButton->setMaximumSize(QSize(140, 16777215));
107
108          horizontalLayout_2->addWidget(copyButton);
109
110          selectLabel = new QLabel(KbBindWidget);
111          selectLabel->setObjectName(QStringLiteral("selectLabel"));
112          QSizePolicy sizePolicy2(QSizePolicy::Expanding, QSizePolicy::Preferred);
113          sizePolicy2.setHorizontalStretch(0);
114          sizePolicy2.setVerticalStretch(0);
115          sizePolicy2.setHeightForWidth(selectLabel->sizePolicy().hasHeightForWidth());
116          selectLabel->setSizePolicy(sizePolicy2);
117          selectLabel->setAlignment(Qt::AlignCenter);
118
119          horizontalLayout_2->addWidget(selectLabel);
120
121          resetButton = new QPushButton(KbBindWidget);
122          resetButton->setObjectName(QStringLiteral("resetButton"));
123          QSizePolicy sizePolicy3(QSizePolicy::Expanding, QSizePolicy::Fixed);
124          sizePolicy3.setHorizontalStretch(0);
125          sizePolicy3.setVerticalStretch(0);
126          sizePolicy3.setHeightForWidth(resetButton->sizePolicy().hasHeightForWidth());
127          resetButton->setSizePolicy(sizePolicy3);
128          resetButton->setMinimumSize(QSize(140, 0));
129          resetButton->setMaximumSize(QSize(140, 16777215));
130
131          horizontalLayout_2->addWidget(resetButton);
132
133
134          gridLayout->addLayout(horizontalLayout_2, 1, 1, 1, 3);
135
136
137          verticalLayout->addLayout(gridLayout);
138
139          verticalSpacer = new QSpacerItem(20, 40, QSizePolicy::Minimum, QSizePolicy::Expanding
    );
140
141          verticalLayout->addItem(verticalSpacer);
142
143          label_2 = new QLabel(KbBindWidget);
144          label_2->setObjectName(QStringLiteral("label_2"));
145          label_2->setFont(font);
146
147          verticalLayout->addWidget(label_2);
148
149          line_2 = new QFrame(KbBindWidget);
150          line_2->setObjectName(QStringLiteral("line_2"));
151          line_2->setFrameShape(QFrame::HLine);
152          line_2->setFrameShadow(QFrame::Sunken);
153
154          verticalLayout->addWidget(line_2);
155
156          widget = new QWidget(KbBindWidget);
157          widget->setObjectName(QStringLiteral("widget"));
158          QSizePolicy sizePolicy4(QSizePolicy::Preferred, QSizePolicy::Fixed);
159          sizePolicy4.setHorizontalStretch(0);
160          sizePolicy4.setVerticalStretch(0);
161          sizePolicy4.setHeightForWidth(widget->sizePolicy().hasHeightForWidth());
162          widget->setSizePolicy(sizePolicy4);
163          widget->setMinimumSize(QSize(0, 280));
164          widget->setMaximumSize(QSize(16777215, 280));
165          gridLayout_2 = new QGridLayout(widget);
166          gridLayout_2->setSpacing(0);
167          gridLayout_2->setObjectName(QStringLiteral("gridLayout_2"));
```

```
168         gridLayout_2->setContentsMargins(0, 0, 0, 0);
169         rbWidget = new RebindWidget(widget);
170         rbWidget->setObjectName(QStringLiteral("rbWidget"));
171         QSizePolicy sizePolicy5(QSizePolicy::Expanding, QSizePolicy::Expanding);
172         sizePolicy5.setHorizontalStretch(0);
173         sizePolicy5.setVerticalStretch(0);
174         sizePolicy5.setHeightForWidth(rbWidget->sizePolicy().hasHeightForWidth());
175         rbWidget->setSizePolicy(sizePolicy5);
176
177         gridLayout_2->addWidget(rbWidget, 0, 0, 1, 1);
178
179
180         verticalLayout->addWidget(widget);
181
182         QWidget::setTabOrder(copyButton, resetButton);
183
184         retranslateUi(KbBindWidget);
185
186         QMetaObject::connectSlotsByName(KbBindWidget);
187     } // setupUi
```

Here is the call graph for this function:



Here is the caller graph for this function:

12.91.3 Field Documentation

12.91.3.1 QPushButton* Ui_KbBindWidget::copyButton

Definition at line 41 of file ui_kbbindwidget.h.

Referenced by retranslateUi(), and setupUi().

12.91.3.2 QGridLayout* Ui_KbBindWidget::gridLayout

Definition at line 36 of file ui_kbbindwidget.h.

Referenced by setupUi().

12.91.3.3 QGridLayout* Ui_KbBindWidget::gridLayout_2

Definition at line 48 of file ui_kbbindwidget.h.

Referenced by setupUi().

12.91.3.4 QHBoxLayout* Ui_KbBindWidget::horizontalLayout_2

Definition at line 40 of file ui_kbbindwidget.h.

Referenced by setupUi().

12.91.3.5 QSpacerItem* Ui_KbBindWidget::horizontalSpacer

Definition at line 39 of file ui_kbbindwidget.h.

Referenced by setupUi().

12.91.3.6 QSpacerItem* Ui_KbBindWidget::horizontalSpacer_2

Definition at line 37 of file ui_kbbindwidget.h.

Referenced by setupUi().

12.91.3.7 KeyWidget* Ui_KbBindWidget::keyWidget

Definition at line 38 of file ui_kbbindwidget.h.

Referenced by KbBindWidget::KbBindWidget(), KbBindWidget::newLayout(), KbBindWidget::setBind(), setupUi(), and KbBindWidget::updateBind().

12.91.3.8 QLabel* Ui_KbBindWidget::label

Definition at line 34 of file ui_kbbindwidget.h.

Referenced by retranslateUi(), and setupUi().

12.91.3.9 QLabel* Ui_KbBindWidget::label_2

Definition at line 45 of file ui_kbbindwidget.h.

Referenced by retranslateUi(), and setupUi().

12.91.3.10 QFrame* Ui_KbBindWidget::line

Definition at line 35 of file ui_kbbindwidget.h.

Referenced by setupUi().

12.91.3.11 QFrame* Ui_KbBindWidget::line_2

Definition at line 46 of file ui_kbbindwidget.h.

Referenced by setupUi().

12.91.3.12 RebindWidget* Ui_KbBindWidget::rbWidget

Definition at line 49 of file ui_kbbindwidget.h.

Referenced by KbBindWidget::newLayout(), KbBindWidget::newSelection(), KbBindWidget::setBind(), setupUi(), and KbBindWidget::updateBind().

12.91.3.13 QPushButton* Ui_KbBindWidget::resetButton

Definition at line 43 of file ui_kbbindwidget.h.

Referenced by retranslateUi(), and setupUi().

12.91.3.14 QLabel* Ui_KbBindWidget::selectLabel

Definition at line 42 of file ui_kbbindwidget.h.

Referenced by retranslateUi(), setupUi(), and KbBindWidget::updateSelDisplay().

12.91.3.15 QVBoxLayout* Ui_KbBindWidget::verticalLayout

Definition at line 33 of file ui_kbbindwidget.h.

Referenced by setupUi().

12.91.3.16 QSpacerItem* Ui_KbBindWidget::verticalSpacer

Definition at line 44 of file ui_kbbindwidget.h.

Referenced by setupUi().

12.91.3.17 QWidget* Ui_KbBindWidget::widget

Definition at line 47 of file ui_kbbindwidget.h.

Referenced by setupUi().

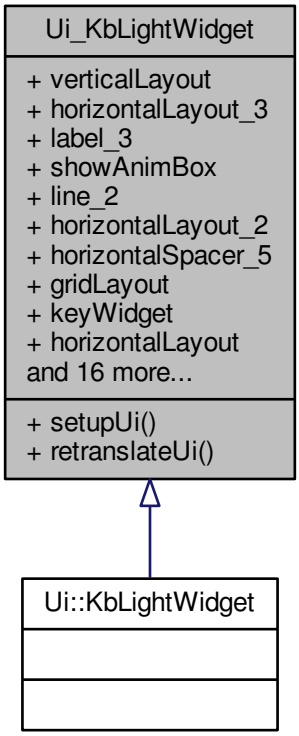
The documentation for this class was generated from the following file:

- src/ckb/ui_kbbindwidget.h

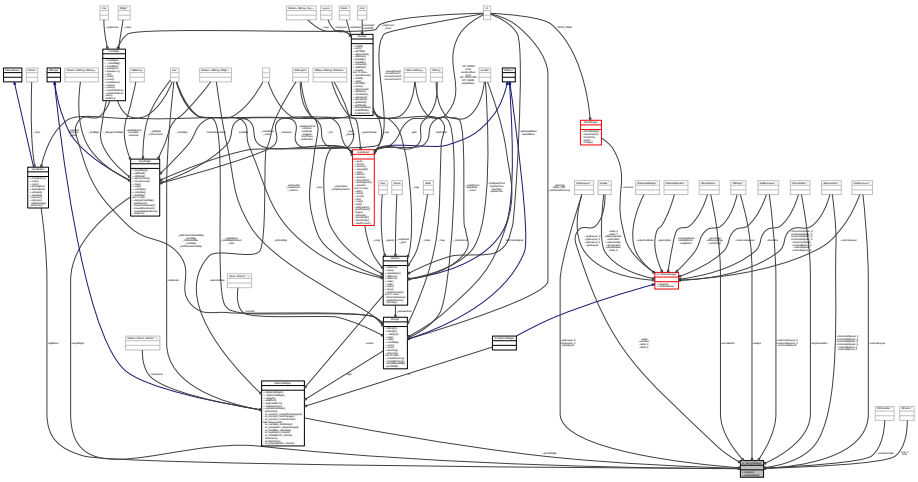
12.92 Ui_KbLightWidget Class Reference

`#include <src/ckb/ui_kblightwidget.h>`

Inheritance diagram for Ui_KbLightWidget:



Collaboration diagram for Ui_KbLightWidget:



Public Member Functions

- void setupUi (QWidget *KbLightWidget)
- void retranslateUi (QWidget *KbLightWidget)

Data Fields

- QVBoxLayout * verticalLayout
- QHBoxLayout * horizontalLayout_3
- QLabel * label_3
- QCheckBox * showAnimBox
- QFrame * line_2
- QHBoxLayout * horizontalLayout_2
- QSpacerItem * horizontalSpacer_5
- QGridLayout * gridLayout
- KeyWidget * keyWidget
- QHBoxLayout * horizontalLayout
- ColorButton * bgButton
- QLabel * selLabel
- QPushButton * animButton
- QSpacerItem * horizontalSpacer_6
- QSpacerItem * verticalSpacer_2
- QGridLayout * gridLayout_2
- QComboBox * brightnessBox
- QLabel * label
- QSpacerItem * horizontalSpacer_3
- QSpacerItem * horizontalSpacer_4
- QSpacerItem * verticalSpacer_3
- QLabel * label_2
- QFrame * line
- QWidget * widget
- QGridLayout * gridLayout_4
- KbAnimWidget * animWidget

12.92.1 Detailed Description

Definition at line 33 of file ui_kblightwidget.h.

12.92.2 Member Function Documentation

12.92.2.1 void Ui_KbLightWidget::retranslateUi (QWidget * *KbLightWidget*) [inline]

Definition at line 241 of file ui_kblightwidget.h.

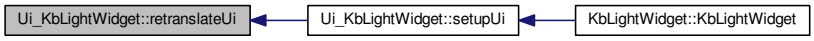
References animButton, bgButton, brightnessBox, label, label_2, label_3, selLabel, and showAnimBox.

Referenced by setupUi().

```
242     {
243         KbLightWidget->setWindowTitle(QApplication::translate("KbLightWidget", "Form", Q_NULLPTR));
244         label_3->setText(QApplication::translate("KbLightWidget", "Colors", Q_NULLPTR));
245 #ifndef QT_NO_TOOLTIP
246         showAnimBox->setToolTip(QApplication::translate("KbLightWidget", "If enabled, the same
      colors will be displayed here as on the keyboard. If disabled, the base colors will be shown.", Q_NULLPTR));
247 #endif // QT_NO_TOOLTIP
248         showAnimBox->setText(QApplication::translate("KbLightWidget", "Show animated", Q_NULLPTR
    ));
249         bgButton->setText(QApplication::translate("KbLightWidget", "Change color...", Q_NULLPTR));
250         selLabel->setText(QApplication::translate("KbLightWidget", "Click to select", Q_NULLPTR));
251         animButton->setText(QApplication::translate("KbLightWidget", "New animation...",
    Q_NULLPTR));
252         brightnessBox->clear();
253         brightnessBox->insertItems(0, QStringList()
254          << QApplication::translate("KbLightWidget", "100%", Q_NULLPTR)
255          << QApplication::translate("KbLightWidget", "67%", Q_NULLPTR)
256          << QApplication::translate("KbLightWidget", "33%", Q_NULLPTR)
257          << QApplication::translate("KbLightWidget", "Off", Q_NULLPTR)
```

```
258         );
259         label->setText(QApplication::translate("KbLightWidget", "\342\230\274", Q_NULLPTR));
260         label_2->setText(QApplication::translate("KbLightWidget", "Animations", Q_NULLPTR));
261     } // retranslateUi
```

Here is the caller graph for this function:



12.92.2.2 void Ui_KbLightWidget::setupUi (QWidget * *KbLightWidget*) [inline]

Definition at line 63 of file ui_kblightwidget.h.

References animButton, animWidget, bgButton, brightnessBox, gridLayout, gridLayout_2, gridLayout_4, horizontalLayout, horizontalLayout_2, horizontalLayout_3, horizontalSpacer_3, horizontalSpacer_4, horizontalSpacer_5, horizontalSpacer_6, keyWidget, label, label_2, label_3, line, line_2, retranslateUi(), selLabel, showAnimBox, verticalLayout, verticalSpacer_2, verticalSpacer_3, and widget.

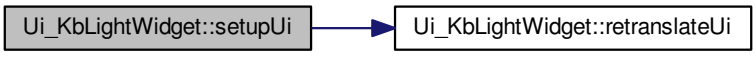
Referenced by KbLightWidget::KbLightWidget().

```
64     {
65         if (KbLightWidget->objectName().isEmpty())
66             KbLightWidget->setObjectName(QStringLiteral("KbLightWidget"));
67         KbLightWidget->resize(632, 540);
68         KbLightWidget->setFocusPolicy(Qt::ClickFocus);
69         verticalLayout = new QVBoxLayout(KbLightWidget);
70         verticalLayout->setObjectName(QStringLiteral("verticalLayout"));
71         verticalLayout->setContentsMargins(0, 0, 0, -1);
72         horizontalLayout_3 = new QHBoxLayout();
73         horizontalLayout_3->setObjectName(QStringLiteral("horizontalLayout_3"));
74         label_3 = new QLabel(KbLightWidget);
75         label_3->setObjectName(QStringLiteral("label_3"));
76         QSizePolicy sizePolicy(QSizePolicy::Expanding, QSizePolicy::Preferred);
77         sizePolicy.setHorizontalStretch(0);
78         sizePolicy.setVerticalStretch(0);
79         sizePolicy.setHeightForWidth(label_3->sizePolicy().hasHeightForWidth());
80         label_3->setSizePolicy(sizePolicy);
81         QFont font;
82         font.setBold(true);
83         font.setWeight(75);
84         label_3->setFont(font);
85
86         horizontalLayout_3->addWidget(label_3);
87
88         showAnimBox = new QCheckBox(KbLightWidget);
89         showAnimBox->setObjectName(QStringLiteral("showAnimBox"));
90         showAnimBox->setChecked(true);
91
92         horizontalLayout_3->addWidget(showAnimBox);
93
94
95         verticalLayout->addLayout(horizontalLayout_3);
96
97         line_2 = new QFrame(KbLightWidget);
98         line_2->setObjectName(QStringLiteral("line_2"));
99         line_2->setFrameShape(QFrame::HLine);
100         line_2->setFrameShadow(QFrame::Sunken);
101
102         verticalLayout->addWidget(line_2);
103
104         horizontalLayout_2 = new QHBoxLayout();
105         horizontalLayout_2->setObjectName(QStringLiteral("horizontalLayout_2"));
106         horizontalSpacer_5 = new QSpacerItem(40, 20, QSizePolicy::Expanding,
      QSizePolicy::Minimum);
107
108         horizontalLayout_2->addItem(horizontalSpacer_5);
109
110         gridLayout = new QGridLayout();
111         gridLayout->setObjectName(QStringLiteral("gridLayout"));
112         keyWidget = new KeyWidget(KbLightWidget);
```

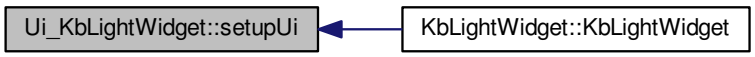
```
113         keyWidget->setObjectName(QStringLiteral("keyWidget"));
114         QSizePolicy sizePolicy1(QSizePolicy::Expanding, QSizePolicy::Expanding);
115         sizePolicy1.setHorizontalStretch(0);
116         sizePolicy1.setVerticalStretch(0);
117         sizePolicy1.setHeightForWidth(keyWidget->sizePolicy().hasHeightForWidth());
118         keyWidget->setSizePolicy(sizePolicy1);
119         keyWidget->setMinimumSize(QSize(100, 100));
120         keyWidget->setFocusPolicy(Qt::ClickFocus);
121 
122         gridLayout->addWidget(keyWidget, 0, 0, 1, 3);
123 
124         horizontalLayout = new QHBoxLayout();
125         horizontalLayout->setObjectName(QStringLiteral("horizontalLayout"));
126         bgButton = new ColorButton(KbLightWidget);
127         bgButton->setObjectName(QStringLiteral("bgButton"));
128         QSizePolicy sizePolicy2(QSizePolicy::Fixed, QSizePolicy::Fixed);
129         sizePolicy2.setHorizontalStretch(0);
130         sizePolicy2.setVerticalStretch(0);
131         sizePolicy2.setHeightForWidth(bgButton->sizePolicy().hasHeightForWidth());
132         bgButton->setSizePolicy(sizePolicy2);
133         bgButton->setMinimumSize(QSize(140, 0));
134         bgButton->setMaximumSize(QSize(140, 16777215));
135 
136         horizontalLayout->addWidget(bgButton);
137 
138         selLabel = new QLabel(KbLightWidget);
139         selLabel->setObjectName(QStringLiteral("selLabel"));
140         sizePolicy.setHeightForWidth(selLabel->sizePolicy().hasHeightForWidth());
141         selLabel->setSizePolicy(sizePolicy);
142         selLabel->setAlignment(Qt::AlignCenter);
143 
144         horizontalLayout->addWidget(selLabel);
145 
146         animButton = new QPushButton(KbLightWidget);
147         animButton->setObjectName(QStringLiteral("animButton"));
148         sizePolicy2.setHeightForWidth(animButton->sizePolicy().hasHeightForWidth());
149         animButton->setSizePolicy(sizePolicy2);
150         animButton->setMinimumSize(QSize(140, 0));
151         animButton->setMaximumSize(QSize(140, 16777215));
152 
153         horizontalLayout->addWidget(animButton);
154 
155 
156         gridLayout->addLayout(horizontalLayout, 1, 0, 1, 3);
157 
158 
159         horizontalLayout_2->addLayout(gridLayout);
160 
161         horizontalSpacer_6 = new QSpacerItem(40, 20, QSizePolicy::Expanding,
    QSizePolicy::Minimum);
162 
163         horizontalLayout_2->addItem(horizontalSpacer_6);
164 
165 
166         verticalLayout->addLayout(horizontalLayout_2);
167 
168         verticalSpacer_2 = new QSpacerItem(20, 40, QSizePolicy::Minimum,
    QSizePolicy::Expanding);
169 
170         verticalLayout->addItem(verticalSpacer_2);
171 
172         gridLayout_2 = new QGridLayout();
173         gridLayout_2->setObjectName(QStringLiteral("gridLayout_2"));
174         brightnessBox = new QComboBox(KbLightWidget);
175         brightnessBox->setObjectName(QStringLiteral("brightnessBox"));
176         sizePolicy2.setHeightForWidth(brightnessBox->sizePolicy().hasHeightForWidth());
177         brightnessBox->setSizePolicy(sizePolicy2);
178 
179         gridLayout_2->addWidget(brightnessBox, 1, 3, 1, 1);
180 
181         label = new QLabel(KbLightWidget);
182         label->setObjectName(QStringLiteral("label"));
183         QFont font1;
184         font1.setPointSize(20);
185         label->setFont(font1);
186         label->setAlignment(Qt::AlignRight|Qt::AlignTrailing|Qt::AlignVCenter);
187 
188         gridLayout_2->addWidget(label, 1, 2, 1, 1);
189 
190         horizontalSpacer_3 = new QSpacerItem(0, 0, QSizePolicy::Expanding,
    QSizePolicy::Minimum);
191 
192         gridLayout_2->addItem(horizontalSpacer_3, 0, 4, 1, 1);
193 
194         horizontalSpacer_4 = new QSpacerItem(0, 0, QSizePolicy::Expanding,
    QSizePolicy::Minimum);
195 
```

```
196         gridLayout_2->addItem(horizontalSpacer_4, 0, 2, 1, 1);
197
198
199         verticalLayout->addLayout(gridLayout_2);
200
201         verticalSpacer_3 = new QSpacerItem(20, 40, QSizePolicy::Minimum,
    QSizePolicy::Expanding);
202
203         verticalLayout->addItem(verticalSpacer_3);
204
205         label_2 = new QLabel(KbLightWidget);
206         label_2->setObjectName(QStringLiteral("label_2"));
207         label_2->setFont(font);
208
209         verticalLayout->addWidget(label_2);
210
211         line = new QFrame(KbLightWidget);
212         line->setObjectName(QStringLiteral("line"));
213         line->setFrameShape(QFrame::HLine);
214         line->setFrameShadow(QFrame::Sunken);
215
216         verticalLayout->addWidget(line);
217
218         widget = new QWidget(KbLightWidget);
219         widget->setObjectName(QStringLiteral("widget"));
220         widget->setMinimumSize(QSize(0, 170));
221         gridLayout_4 = new QGridLayout(widget);
222         gridLayout_4->setObjectName(QStringLiteral("gridLayout_4"));
223         gridLayout_4->setContentsMargins(-1, 0, -1, 0);
224         animWidget = new KbAnimWidget(widget);
225         animWidget->setObjectName(QStringLiteral("animWidget"));
226
227         gridLayout_4->addWidget(animWidget, 0, 0, 1, 1);
228
229
230         verticalLayout->addWidget(widget);
231
232         QWidget::setTabOrder(showAnimBox, bgButton);
233         QWidget::setTabOrder(bgButton, animButton);
234         QWidget::setTabOrder(animButton, brightnessBox);
235
236         retranslateUi(KbLightWidget);
237
238         QMetaObject::connectSlotsByName(KbLightWidget);
239     } // setupUi
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.92.3 Field Documentation

12.92.3.1 QPushButton* Ui_KbLightWidget::animButton

Definition at line 48 of file ui_kblightwidget.h.

Referenced by KbLightWidget::KbLightWidget(), retranslateUi(), and setupUi().

12.92.3.2 KbAnimWidget* Ui_KbLightWidget::animWidget

Definition at line 61 of file ui_kblightwidget.h.

Referenced by KbLightWidget::KbLightWidget(), KbLightWidget::newSelection(), KbLightWidget::on_animButton_-clicked(), KbLightWidget::setLight(), and setupUi().

12.92.3.3 ColorButton* Ui_KbLightWidget::bgButton

Definition at line 46 of file ui_kblightwidget.h.

Referenced by KbLightWidget::KbLightWidget(), KbLightWidget::newSelection(), retranslateUi(), and setupUi().

12.92.3.4 QComboBox* Ui_KbLightWidget::brightnessBox

Definition at line 52 of file ui_kblightwidget.h.

Referenced by retranslateUi(), setupUi(), and KbLightWidget::updateLight().

12.92.3.5 QGridLayout* Ui_KbLightWidget::gridLayout

Definition at line 43 of file ui_kblightwidget.h.

Referenced by setupUi().

12.92.3.6 QGridLayout* Ui_KbLightWidget::gridLayout_2

Definition at line 51 of file ui_kblightwidget.h.

Referenced by setupUi().

12.92.3.7 QGridLayout* Ui_KbLightWidget::gridLayout_4

Definition at line 60 of file ui_kblightwidget.h.

Referenced by setupUi().

12.92.3.8 QHBoxLayout* Ui_KbLightWidget::horizontalLayout

Definition at line 45 of file ui_kblightwidget.h.

Referenced by setupUi().

12.92.3.9 QHBoxLayout* Ui_KbLightWidget::horizontalLayout_2

Definition at line 41 of file ui_kblightwidget.h.

Referenced by setupUi().

12.92.3.10 QHBoxLayout* Ui_KbLightWidget::horizontalLayout_3

Definition at line 37 of file ui_kblightwidget.h.

Referenced by setupUi().

12.92.3.11 QSpacerItem* Ui_KbLightWidget::horizontalSpacer_3

Definition at line 54 of file ui_kblightwidget.h.

Referenced by setupUi().

12.92.3.12 QSpacerItem* Ui_KbLightWidget::horizontalSpacer_4

Definition at line 55 of file ui_kblightwidget.h.

Referenced by setupUi().

12.92.3.13 QSpacerItem* Ui_KbLightWidget::horizontalSpacer_5

Definition at line 42 of file ui_kblightwidget.h.

Referenced by setupUi().

12.92.3.14 QSpacerItem* Ui_KbLightWidget::horizontalSpacer_6

Definition at line 49 of file ui_kblightwidget.h.

Referenced by setupUi().

12.92.3.15 KeyWidget* Ui_KbLightWidget::keyWidget

Definition at line 44 of file ui_kblightwidget.h.

Referenced by KbLightWidget::changeAnim(), KbLightWidget::changeAnimKeys(), KbLightWidget::changeColor(), KbLightWidget::KbLightWidget(), KbLightWidget::on_animButton_clicked(), KbLightWidget::on_bgButton_clicked(), KbLightWidget::on_showAnimBox_clicked(), KbLightWidget::setLight(), KbLightWidget::setMonochrome(), setupUi(), KbLightWidget::toggleSidelight(), and KbLightWidget::updateLight().

12.92.3.16 QLabel* Ui_KbLightWidget::label

Definition at line 53 of file ui_kblightwidget.h.

Referenced by retranslateUi(), and setupUi().

12.92.3.17 QLabel* Ui_KbLightWidget::label_2

Definition at line 57 of file ui_kblightwidget.h.

Referenced by retranslateUi(), and setupUi().

12.92.3.18 QLabel* Ui_KbLightWidget::label_3

Definition at line 38 of file ui_kblightwidget.h.

Referenced by retranslateUi(), and setupUi().

12.92.3.19 QFrame* Ui_KbLightWidget::line

Definition at line 58 of file ui_kblightwidget.h.

Referenced by setupUi().

12.92.3.20 QFrame* Ui_KbLightWidget::line_2

Definition at line 40 of file ui_kblightwidget.h.

Referenced by setupUi().

12.92.3.21 QLabel* Ui_KbLightWidget::selLabel

Definition at line 47 of file ui_kblightwidget.h.

Referenced by KbLightWidget::newSelection(), retranslateUi(), and setupUi().

12.92.3.22 QCheckBox* Ui_KbLightWidget::showAnimBox

Definition at line 39 of file ui_kblightwidget.h.

Referenced by KbLightWidget::KbLightWidget(), retranslateUi(), KbLightWidget::setLight(), and setupUi().

12.92.3.23 QVBoxLayout* Ui_KbLightWidget::verticalLayout

Definition at line 36 of file ui_kblightwidget.h.

Referenced by setupUi().

12.92.3.24 QSpacerItem* Ui_KbLightWidget::verticalSpacer_2

Definition at line 50 of file ui_kblightwidget.h.

Referenced by setupUi().

12.92.3.25 QSpacerItem* Ui_KbLightWidget::verticalSpacer_3

Definition at line 56 of file ui_kblightwidget.h.

Referenced by setupUi().

12.92.3.26 QWidget* Ui_KbLightWidget::widget

Definition at line 59 of file ui_kblightwidget.h.

Referenced by setupUi().

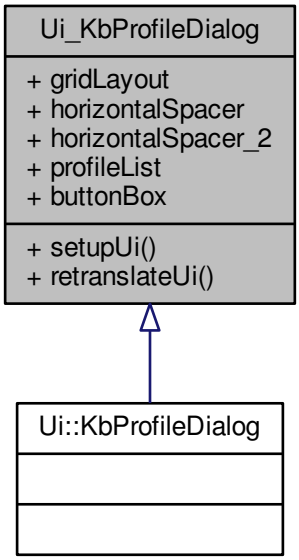
The documentation for this class was generated from the following file:

- src/ckb/ui_kblightwidget.h

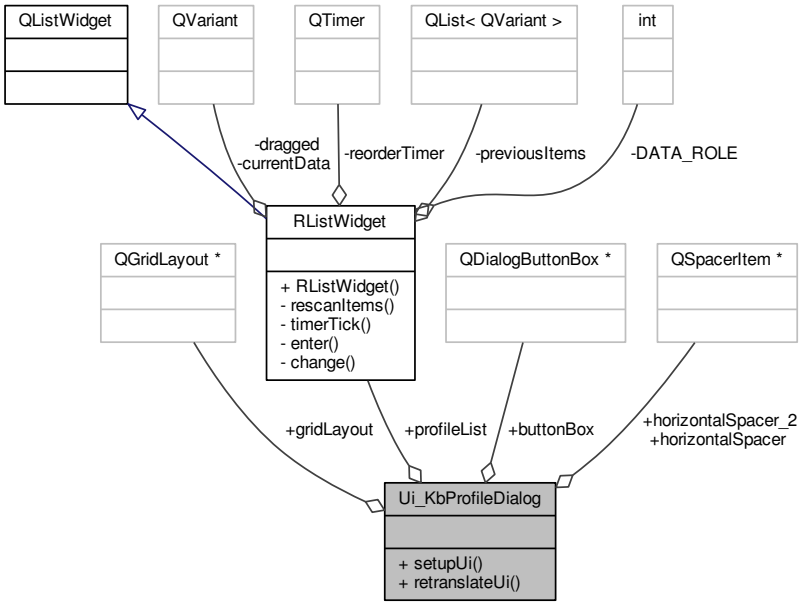
12.93 Ui_KbProfileDialog Class Reference

`#include <src/ckb/ui_kbprofiledialog.h>`

Inheritance diagram for Ui_KbProfileDialog:



Collaboration diagram for Ui_KbProfileDialog:



Public Member Functions

- void setupUi (QDialog *KbProfileDialog)
- void retranslateUi (QDialog *KbProfileDialog)

Data Fields

- QGridLayout * gridLayout
- QSpacerItem * horizontalSpacer
- QSpacerItem * horizontalSpacer_2
- RListWidget * profileList
- QDialogButtonBox * buttonBox

12.93.1 Detailed Description

Definition at line 25 of file ui_kbprofiledialog.h.

12.93.2 Member Function Documentation

12.93.2.1 void Ui_KbProfileDialog::retranslateUi (QDialog * *KbProfileDialog*) [inline]

Definition at line 73 of file ui_kbprofiledialog.h.

Referenced by setupUi().

```
74     {
75         KbProfileDialog->setWindowTitle(QApplication::translate("KbProfileDialog", "Keyboard Profiles",
       Q_NULLPTR));
76     } // retranslateUi
```

Here is the caller graph for this function:

12.93.2.2 void Ui_KbProfileDialog::setupUi (QDialog * *KbProfileDialog*) [inline]

Definition at line 34 of file ui_kbprofiledialog.h.

References buttonBox, gridLayout, horizontalSpacer, horizontalSpacer_2, profileList, and retranslateUi().

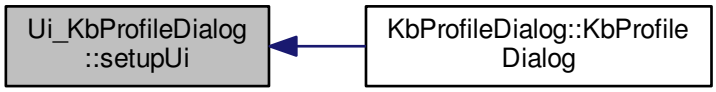
Referenced by KbProfileDialog::KbProfileDialog().

```
35     {
36         if (KbProfileDialog->objectName().isEmpty())
37             KbProfileDialog->setObjectName(QStringLiteral("KbProfileDialog"));
38         KbProfileDialog->resize(400, 300);
39         KbProfileDialog->setFocusPolicy(Qt::ClickFocus);
40         gridLayout = new QGridLayout(KbProfileDialog);
41         gridLayout->setObjectName(QStringLiteral("gridLayout"));
42         horizontalSpacer = new QSpacerItem(40, 20, QSizePolicy::Expanding,
       QSizePolicy::Minimum);
43
44         gridLayout->addItem(horizontalSpacer, 0, 0, 1, 1);
```

```
45
46          horizontalSpacer_2 = new QSpacerItem(40, 20, QSizePolicy::Expanding,
    QSizePolicy::Minimum);
47
48          gridLayout->addItem(horizontalSpacer_2, 0, 2, 1, 1);
49
50          profileList = new RListWidget(KbProfileDialog);
51          profileList->setObjectName(QStringLiteral("profileList"));
52          profileList->setContextMenuPolicy(Qt::CustomContextMenu);
53          profileList->setFrameShadow(QFrame::Plain);
54          profileList->setIconSize(QSize(24, 24));
55
56          gridLayout->addWidget(profileList, 0, 1, 1, 1);
57
58          buttonBox = new QDialogButtonBox(KbProfileDialog);
59          buttonBox->setObjectName(QStringLiteral("buttonBox"));
60          buttonBox->setOrientation(Qt::Horizontal);
61          buttonBox->setStandardButtons(QDialogButtonBox::Ok);
62
63          gridLayout->addWidget(buttonBox, 1, 0, 1, 3);
64
65
66          retranslateUi(KbProfileDialog);
67          QObject::connect(buttonBox, SIGNAL(accepted()), KbProfileDialog, SLOT(accept()));
68          QObject::connect(buttonBox, SIGNAL(rejected()), KbProfileDialog, SLOT(reject()));
69
70          QMetaObject::connectSlotsByName(KbProfileDialog);
71      } // setupUi
```

Here is the call graph for this function:

Here is the caller graph for this function:



12.93.3 Field Documentation

12.93.3.1 QDialogButtonBox* Ui_KbProfileDialog::buttonBox

Definition at line 32 of file ui_kbprofiledialog.h.

Referenced by setupUi().

12.93.3.2 QGridLayout* Ui_KbProfileDialog::gridLayout

Definition at line 28 of file ui_kbprofiledialog.h.

Referenced by setupUi().

12.93.3.3 QSpacerItem* Ui_KbProfileDialog::horizontalSpacer

Definition at line 29 of file ui_kbprofiledialog.h.

Referenced by setupUi().

12.93.3.4 QSpacerItem* Ui_KbProfileDialog::horizontalSpacer_2

Definition at line 30 of file ui_kbprofiledialog.h.

Referenced by setupUi().

12.93.3.5 RListWidget* Ui_KbProfileDialog::profileList

Definition at line 31 of file ui_kbprofiledialog.h.

Referenced by KbProfileDialog::addNewProfileItem(), KbProfileDialog::KbProfileDialog(), KbProfileDialog::on_profileList_customContextMenuRequested(), KbProfileDialog::on_profileList_itemClicked(), KbProfileDialog::profileList_reordered(), KbProfileDialog::repopulate(), and setupUi().

The documentation for this class was generated from the following file:

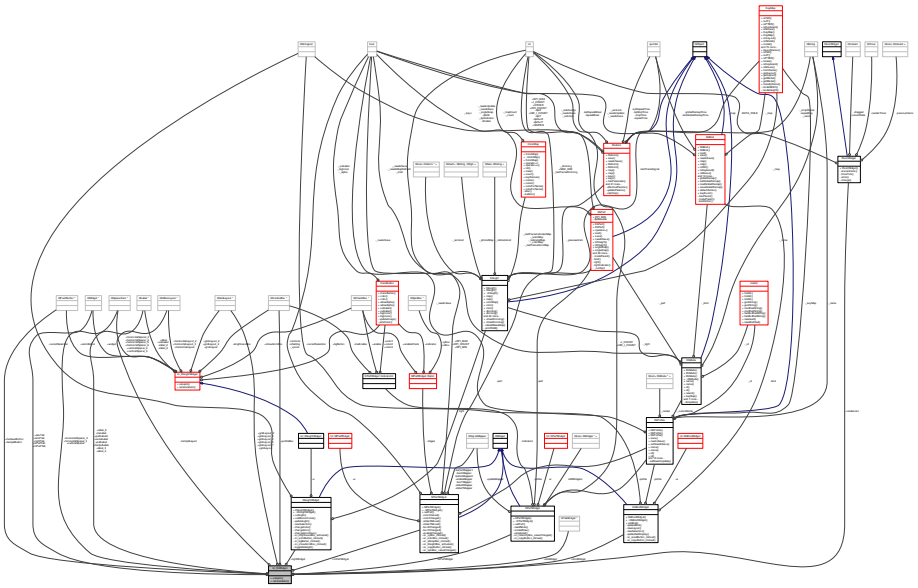
- src/ckb/ui_kbprofiledialog.h

12.94 Ui_KbWidget Class Reference

`#include <src/ckb/ui_kbwidget.h>`

Inheritance diagram for Ui_KbWidget:

Collaboration diagram for Ui_KbWidget:



Public Member Functions

- void setupUi (QWidget *KbWidget)
- void retranslateUi (QWidget *KbWidget)

Data Fields

- QGridLayout * gridLayout
- QComboBox * profileBox
- QTabWidget * tabWidget
- QWidget * lightTab
- QGridLayout * gridLayout_2
- KbLightWidget * lightWidget
- QLabel * label_3
- QWidget * bindTab
- QGridLayout * gridLayout_5
- KbBindWidget * bindWidget
- QWidget * kPerfTab
- QGridLayout * gridLayout_4
- KPerfWidget * kPerfWidget
- QWidget * mPerfTab
- QGridLayout * gridLayout_7
- MPerfWidget * mPerfWidget
- QWidget * devTab
- QGridLayout * gridLayout_3
- QLabel * fwUpdLabel
- QSpacerItem * verticalSpacer
- QLabel * fwLabel
- QLabel * label_4
- QSpacerItem * horizontalSpacer_2
- QLabel * label_9
- QLabel * serialLabel
- QHBoxLayout * fwUpdLayout
- QPushButton * fwUpdButton
- QSpacerItem * horizontalSpacer_3
- QLabel * pollLabel2
- QLabel * pollLabel
- RListWidget * modesList
- QPushButton * hwSaveButton

12.94.1 Detailed Description

Definition at line 33 of file ui_kbwidget.h.

12.94.2 Member Function Documentation

12.94.2.1 void Ui_KbWidget::retranslateUi (QWidget * *KbWidget*) [inline]

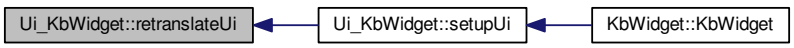
Definition at line 268 of file ui_kbwidget.h.

References bindTab, devTab, fwLabel, fwUpdButton, fwUpdLabel, hwSaveButton, kPerfTab, label_3, label_4, label_9, lightTab, mPerfTab, pollLabel, pollLabel2, serialLabel, and tabWidget.

Referenced by setupUi().

```
269     {
270         KbWidget->setWindowTitle(QApplication::translate("KbWidget", "Form", Q_NULLPTR));
271         label_3->setText(QApplication::translate("KbWidget", "Tip: Drag+drop items to reorder.
     Right-click for menu.", Q_NULLPTR));
272         tabWidget->setTabText(tabWidget->indexOf(lightTab),
    QApplication::translate("KbWidget", "Lighting", Q_NULLPTR));
273         tabWidget->setTabText(tabWidget->indexOf(bindTab), QApplication::translate
    ("KbWidget", "Binding", Q_NULLPTR));
274         tabWidget->setTabText(tabWidget->indexOf(kPerfTab),
    QApplication::translate("KbWidget", "Performance", Q_NULLPTR));
275         tabWidget->setTabText(tabWidget->indexOf(mPerfTab),
    QApplication::translate("KbWidget", "Performance", Q_NULLPTR));
276         fwUpdLabel->setText(QApplication::translate("KbWidget", "Status:", Q_NULLPTR));
277         fwLabel->setText(QApplication::translate("KbWidget", "N/A", Q_NULLPTR));
278         label_4->setText(QApplication::translate("KbWidget", "Serial Number:", Q_NULLPTR));
279         label_9->setText(QApplication::translate("KbWidget", "Firmware version:", Q_NULLPTR));
280         serialLabel->setText(QApplication::translate("KbWidget", "N/A", Q_NULLPTR));
281         fwUpdButton->setText(QApplication::translate("KbWidget", "Check for updates", Q_NULLPTR)
    );
282         pollLabel2->setText(QApplication::translate("KbWidget", "Poll rate:", Q_NULLPTR));
283         pollLabel->setText(QApplication::translate("KbWidget", "N/A", Q_NULLPTR));
284         tabWidget->setTabText(tabWidget->indexOf(devTab), QApplication::translate("
    KbWidget", "Device", Q_NULLPTR));
285         hwSaveButton->setText(QApplication::translate("KbWidget", "Save to hardware", Q_NULLPTR
    ));
286     } // retranslateUi
```

Here is the caller graph for this function:



12.94.2.2 void Ui_KbWidget::setupUi (QWidget * *KbWidget*) [inline]

Definition at line 69 of file ui_kbwidget.h.

References bindTab, bindWidget, devTab, fwLabel, fwUpdButton, fwUpdLabel, fwUpdLayout, gridLayout, gridLayout_2, gridLayout_3, gridLayout_4, gridLayout_5, gridLayout_7, horizontalSpacer_2, horizontalSpacer_3, hwSaveButton, kPerfTab, kPerfWidget, label_3, label_4, label_9, lightTab, lightWidget, modesList, mPerfTab, mPerfWidget, pollLabel, pollLabel2, profileBox, retranslateUi(), serialLabel, tabWidget, and verticalSpacer.

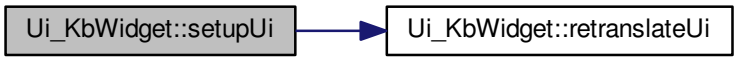
Referenced by KbWidget::KbWidget().

```
70     {
71         if (KbWidget->objectName().isEmpty())
72             KbWidget->setObjectName(QStringLiteral("KbWidget"));
73         KbWidget->resize(773, 539);
74         gridLayout = new QGridLayout(KbWidget);
75         gridLayout->setSpacing(9);
76         gridLayout->setObjectName(QStringLiteral("gridLayout"));
77         gridLayout->setContentsMargins(9, -1, 9, 9);
78         profileBox = new QComboBox(KbWidget);
79         profileBox->setObjectName(QStringLiteral("profileBox"));
80         QSizePolicy sizePolicy(QSizePolicy::Preferred, QSizePolicy::Fixed);
81         sizePolicy.setHorizontalStretch(18);
82         sizePolicy.setVerticalStretch(0);
83         sizePolicy.setHeightForWidth(profileBox->sizePolicy().hasHeightForWidth());
84         profileBox->setSizePolicy(sizePolicy);
85         profileBox->setIconSize(QSize(24, 24));
86
87         gridLayout->addWidget(profileBox, 0, 0, 1, 1);
88
89         tabWidget = new QTabWidget(KbWidget);
90         tabWidget->setObjectName(QStringLiteral("tabWidget"));
91         QSizePolicy sizePolicy1(QSizePolicy::Expanding, QSizePolicy::Expanding);
92         sizePolicy1.setHorizontalStretch(82);
93         sizePolicy1.setVerticalStretch(0);
94         sizePolicy1.setHeightForWidth(tabWidget->sizePolicy().hasHeightForWidth());
95         tabWidget->setSizePolicy(sizePolicy1);
96         lightTab = new QWidget();
```

```
97          lightTab->setObjectName(QStringLiteral("lightTab"));
98          gridLayout_2 = new QGridLayout(lightTab);
99          gridLayout_2->setObjectName(QStringLiteral("gridLayout_2"));
100          gridLayout_2->setHorizontalSpacing(0);
101          gridLayout_2->setVerticalSpacing(6);
102          gridLayout_2->setContentsMargins(6, 6, 6, 6);
103          lightWidget = new KbLightWidget(lightTab);
104          lightWidget->setObjectName(QStringLiteral("lightWidget"));
105          QSizePolicy sizePolicy2(QSizePolicy::Preferred, QSizePolicy::Expanding);
106          sizePolicy2.setHorizontalStretch(0);
107          sizePolicy2.setVerticalStretch(0);
108          sizePolicy2.setHeightForWidth(lightWidget->sizePolicy().hasHeightForWidth());
109          lightWidget->setSizePolicy(sizePolicy2);
110
111          gridLayout_2->addWidget(lightWidget, 0, 0, 1, 1);
112
113          label_3 = new QLabel(lightTab);
114          label_3->setObjectName(QStringLiteral("label_3"));
115          QSizePolicy sizePolicy3(QSizePolicy::Preferred, QSizePolicy::Minimum);
116          sizePolicy3.setHorizontalStretch(0);
117          sizePolicy3.setVerticalStretch(0);
118          sizePolicy3.setHeightForWidth(label_3->sizePolicy().hasHeightForWidth());
119          label_3->setSizePolicy(sizePolicy3);
120
121          gridLayout_2->addWidget(label_3, 1, 0, 1, 1);
122
123          tabWidget->addTab(lightTab, QString());
124          bindTab = new QWidget();
125          bindTab->setObjectName(QStringLiteral("bindTab"));
126          gridLayout_5 = new QGridLayout(bindTab);
127          gridLayout_5->setSpacing(0);
128          gridLayout_5->setObjectName(QStringLiteral("gridLayout_5"));
129          gridLayout_5->setContentsMargins(6, 6, 6, 6);
130          bindWidget = new KbBindWidget(bindTab);
131          bindWidget->setObjectName(QStringLiteral("bindWidget"));
132          sizePolicy2.setHeightForWidth(bindWidget->sizePolicy().hasHeightForWidth());
133          bindWidget->setSizePolicy(sizePolicy2);
134
135          gridLayout_5->addWidget(bindWidget, 0, 0, 1, 1);
136
137          tabWidget->addTab(bindTab, QString());
138          kPerfTab = new QWidget();
139          kPerfTab->setObjectName(QStringLiteral("kPerfTab"));
140          gridLayout_4 = new QGridLayout(kPerfTab);
141          gridLayout_4->setObjectName(QStringLiteral("gridLayout_4"));
142          gridLayout_4->setContentsMargins(6, 6, 6, 6);
143          kPerfWidget = new KPerfWidget(kPerfTab);
144          kPerfWidget->setObjectName(QStringLiteral("kPerfWidget"));
145
146          gridLayout_4->addWidget(kPerfWidget, 0, 0, 1, 1);
147
148          tabWidget->addTab(kPerfTab, QString());
149          mPerfTab = new QWidget();
150          mPerfTab->setObjectName(QStringLiteral("mPerfTab"));
151          gridLayout_7 = new QGridLayout(mPerfTab);
152          gridLayout_7->setObjectName(QStringLiteral("gridLayout_7"));
153          gridLayout_7->setContentsMargins(6, 6, 6, 6);
154          mPerfWidget = new MPerfWidget(mPerfTab);
155          mPerfWidget->setObjectName(QStringLiteral("mPerfWidget"));
156
157          gridLayout_7->addWidget(mPerfWidget, 0, 0, 1, 1);
158
159          tabWidget->addTab(mPerfTab, QString());
160          devTab = new QWidget();
161          devTab->setObjectName(QStringLiteral("devTab"));
162          gridLayout_3 = new QGridLayout(devTab);
163          gridLayout_3->setObjectName(QStringLiteral("gridLayout_3"));
164          gridLayout_3->setContentsMargins(6, 6, 6, 6);
165          fwUpdLabel = new QLabel(devTab);
166          fwUpdLabel->setObjectName(QStringLiteral("fwUpdLabel"));
167          fwUpdLabel->setMinimumSize(QSize(0, 34));
168
169          gridLayout_3->addWidget(fwUpdLabel, 4, 0, 1, 1);
170
171          verticalSpacer = new QSpacerItem(20, 40, QSizePolicy::Minimum, QSizePolicy::Expanding
    );
172
173          gridLayout_3->addItem(verticalSpacer, 6, 2, 1, 1);
174
175          fwLabel = new QLabel(devTab);
176          fwLabel->setObjectName(QStringLiteral("fwLabel"));
177          fwLabel->setCursor(QCursor(Qt::IBeamCursor));
178          fwLabel->setTextInteractionFlags(Qt::LinksAccessibleByMouse|Qt::TextSelectableByMouse);
179
180          gridLayout_3->addWidget(fwLabel, 2, 2, 1, 1);
181
182          label_4 = new QLabel(devTab);
```

```
183         label_4->setObjectName(QStringLiteral("label_4"));
184         label_4->setMinimumSize(QSize(0, 34));
185 
186         gridLayout_3->addWidget(label_4, 0, 0, 1, 1);
187 
188         horizontalSpacer_2 = new QSpacerItem(40, 1, QSizePolicy::Fixed,
    QSizePolicy::Minimum);
189 
190         gridLayout_3->addItem(horizontalSpacer_2, 2, 1, 1, 1);
191 
192         label_9 = new QLabel(devTab);
193         label_9->setObjectName(QStringLiteral("label_9"));
194         label_9->setMinimumSize(QSize(0, 34));
195 
196         gridLayout_3->addWidget(label_9, 2, 0, 1, 1);
197 
198         serialLabel = new QLabel(devTab);
199         serialLabel->setObjectName(QStringLiteral("serialLabel"));
200         serialLabel->setCursor(QCursor(Qt::IBeamCursor));
201         serialLabel->setTextInteractionFlags(Qt::LinksAccessibleByMouse|
    Qt::TextSelectableByMouse);
202 
203         gridLayout_3->addWidget(serialLabel, 0, 2, 1, 1);
204 
205         fwUpdLayout = new QHBoxLayout();
206         fwUpdLayout->setObjectName(QStringLiteral("fwUpdLayout"));
207         fwUpdButton = new QPushButton(devTab);
208         fwUpdButton->setObjectName(QStringLiteral("fwUpdButton"));
209 
210         fwUpdLayout->addWidget(fwUpdButton);
211 
212         horizontalSpacer_3 = new QSpacerItem(40, 20, QSizePolicy::Expanding,
    QSizePolicy::Minimum);
213 
214         fwUpdLayout->addItem(horizontalSpacer_3);
215 
216 
217         gridLayout_3->addLayout(fwUpdLayout, 4, 2, 1, 1);
218 
219         pollLabel2 = new QLabel(devTab);
220         pollLabel2->setObjectName(QStringLiteral("pollLabel2"));
221         pollLabel2->setMinimumSize(QSize(0, 34));
222 
223         gridLayout_3->addWidget(pollLabel2, 1, 0, 1, 1);
224 
225         pollLabel = new QLabel(devTab);
226         pollLabel->setObjectName(QStringLiteral("pollLabel"));
227 
228         gridLayout_3->addWidget(pollLabel, 1, 2, 1, 1);
229 
230         tabWidget->addTab(devTab, QString());
231 
232         gridLayout->addWidget(tabWidget, 0, 1, 3, 1);
233 
234         modesList = new RListWidget(KbWidget);
235         modesList->setObjectName(QStringLiteral("modesList"));
236         QSizePolicy sizePolicy4(QSizePolicy::Expanding, QSizePolicy::Expanding);
237         sizePolicy4.setHorizontalStretch(18);
238         sizePolicy4.setVerticalStretch(0);
239         sizePolicy4.setHeightForWidth(modesList->sizePolicy().hasHeightForWidth());
240         modesList->setSizePolicy(sizePolicy4);
241         modesList->setContextMenuPolicy(Qt::CustomContextMenu);
242         modesList->setFrameShadow(QFrame::Plain);
243         modesList->setDragDropMode(QAbstractItemView::InternalMove);
244         modesList->setIconSize(QSize(24, 24));
245         modesList->setMovement(QListView::Snap);
246 
247         gridLayout->addWidget(modesList, 1, 0, 1, 1);
248 
249         hwSaveButton = new QPushButton(KbWidget);
250         hwSaveButton->setObjectName(QStringLiteral("hwSaveButton"));
251         QSizePolicy sizePolicy5(QSizePolicy::Minimum, QSizePolicy::Fixed);
252         sizePolicy5.setHorizontalStretch(18);
253         sizePolicy5.setVerticalStretch(0);
254         sizePolicy5.setHeightForWidth(hwSaveButton->sizePolicy().hasHeightForWidth());
255         hwSaveButton->setSizePolicy(sizePolicy5);
256 
257         gridLayout->addWidget(hwSaveButton, 2, 0, 1, 1);
258 
259 
260         retranslateUi(KbWidget);
261 
262         tabWidget->setCurrentIndex(0);
263 
264 
265         QMetaObject::connectSlotsByName(KbWidget);
266     } // setupUi
```

Here is the call graph for this function:



Here is the caller graph for this function:

12.94.3 Field Documentation

12.94.3.1 QWidget* Ui_KbWidget::bindTab

Definition at line 43 of file ui_kbwidget.h.

Referenced by retranslateUi(), and setupUi().

12.94.3.2 KbBindWidget* Ui_KbWidget::bindWidget

Definition at line 45 of file ui_kbwidget.h.

Referenced by KbWidget::modeChanged(), and setupUi().

12.94.3.3 QWidget* Ui_KbWidget::devTab

Definition at line 52 of file ui_kbwidget.h.

Referenced by retranslateUi(), and setupUi().

12.94.3.4 QLabel* Ui_KbWidget::fwLabel

Definition at line 56 of file ui_kbwidget.h.

Referenced by KbWidget::devUpdate(), retranslateUi(), and setupUi().

12.94.3.5 QPushButton* Ui_KbWidget::fwUpdButton

Definition at line 62 of file ui_kbwidget.h.

Referenced by KbWidget::KbWidget(), KbWidget::on_fwUpdButton_clicked(), retranslateUi(), setupUi(), and KbWidget::updateFwButton().

12.94.3.6 QLabel* Ui_KbWidget::fwUpdLabel

Definition at line 54 of file ui_kbwidget.h.

Referenced by KbWidget::KbWidget(), retranslateUi(), and setupUi().

12.94.3.7 QHBoxLayout* Ui_KbWidget::fwUpdLayout

Definition at line 61 of file ui_kbwidget.h.

Referenced by KbWidget::KbWidget(), and setupUi().

12.94.3.8 QGridLayout* Ui_KbWidget::gridLayout

Definition at line 36 of file ui_kbwidget.h.

Referenced by setupUi().

12.94.3.9 QGridLayout* Ui_KbWidget::gridLayout_2

Definition at line 40 of file ui_kbwidget.h.

Referenced by setupUi().

12.94.3.10 QGridLayout* Ui_KbWidget::gridLayout_3

Definition at line 53 of file ui_kbwidget.h.

Referenced by setupUi().

12.94.3.11 QGridLayout* Ui_KbWidget::gridLayout_4

Definition at line 47 of file ui_kbwidget.h.

Referenced by setupUi().

12.94.3.12 QGridLayout* Ui_KbWidget::gridLayout_5

Definition at line 44 of file ui_kbwidget.h.

Referenced by setupUi().

12.94.3.13 QGridLayout* Ui_KbWidget::gridLayout_7

Definition at line 50 of file ui_kbwidget.h.

Referenced by setupUi().

12.94.3.14 QSpacerItem* Ui_KbWidget::horizontalSpacer_2

Definition at line 58 of file ui_kbwidget.h.

Referenced by setupUi().

12.94.3.15 QSpacerItem* Ui_KbWidget::horizontalSpacer_3

Definition at line 63 of file ui_kbwidget.h.

Referenced by setupUi().

12.94.3.16 QPushButton* Ui_KbWidget::hwSaveButton

Definition at line 67 of file ui_kbwidget.h.

Referenced by retranslateUi(), and setupUi().

12.94.3.17 QWidget* Ui_KbWidget::kPerfTab

Definition at line 46 of file ui_kbwidget.h.

Referenced by KbWidget::KbWidget(), retranslateUi(), and setupUi().

12.94.3.18 KPerfWidget* Ui_KbWidget::kPerfWidget

Definition at line 48 of file ui_kbwidget.h.

Referenced by KbWidget::modeChanged(), and setupUi().

12.94.3.19 QLabel* Ui_KbWidget::label_3

Definition at line 42 of file ui_kbwidget.h.

Referenced by retranslateUi(), and setupUi().

12.94.3.20 QLabel* Ui_KbWidget::label_4

Definition at line 57 of file ui_kbwidget.h.

Referenced by retranslateUi(), and setupUi().

12.94.3.21 QLabel* Ui_KbWidget::label_9

Definition at line 59 of file ui_kbwidget.h.

Referenced by retranslateUi(), and setupUi().

12.94.3.22 QWidget* Ui_KbWidget::lightTab

Definition at line 39 of file ui_kbwidget.h.

Referenced by KbWidget::KbWidget(), retranslateUi(), and setupUi().

12.94.3.23 KbLightWidget* Ui_KbWidget::lightWidget

Definition at line 41 of file ui_kbwidget.h.

Referenced by KbWidget::KbWidget(), KbWidget::modeChanged(), and setupUi().

12.94.3.24 RListWidget* Ui_KbWidget::modesList

Definition at line 66 of file ui_kbwidget.h.

Referenced by KbWidget::addNewModeItem(), KbWidget::KbWidget(), KbWidget::modeChanged(), KbWidget::modesList_reordered(), KbWidget::on_modesList_customContextMenuRequested(), KbWidget::on_modesList_itemClicked(), KbWidget::profileChanged(), and setupUi().

12.94.3.25 QWidget* Ui_KbWidget::mPerfTab

Definition at line 49 of file ui_kbwidget.h.

Referenced by KbWidget::KbWidget(), retranslateUi(), and setupUi().

12.94.3.26 MPerfWidget* Ui_KbWidget::mPerfWidget

Definition at line 51 of file ui_kbwidget.h.

Referenced by KbWidget::modeChanged(), and setupUi().

12.94.3.27 QLabel* Ui_KbWidget::pollLabel

Definition at line 65 of file ui_kbwidget.h.

Referenced by KbWidget::devUpdate(), KbWidget::KbWidget(), retranslateUi(), and setupUi().

12.94.3.28 QLabel* Ui_KbWidget::pollLabel2

Definition at line 64 of file ui_kbwidget.h.

Referenced by KbWidget::KbWidget(), retranslateUi(), and setupUi().

12.94.3.29 QComboBox* Ui_KbWidget::profileBox

Definition at line 37 of file ui_kbwidget.h.

Referenced by setupUi(), and KbWidget::updateProfileList().

12.94.3.30 QLabel* Ui_KbWidget::serialLabel

Definition at line 60 of file ui_kbwidget.h.

Referenced by KbWidget::devUpdate(), retranslateUi(), and setupUi().

12.94.3.31 QTabWidget* Ui_KbWidget::tabWidget

Definition at line 38 of file ui_kbwidget.h.

Referenced by KbWidget::KbWidget(), KbWidget::on_tabWidget_currentChanged(), retranslateUi(), setupUi(), KbWidget::showFirstTab(), and KbWidget::showLastTab().

12.94.3.32 QSpacerItem* Ui_KbWidget::verticalSpacer

Definition at line 55 of file ui_kbwidget.h.

Referenced by setupUi().

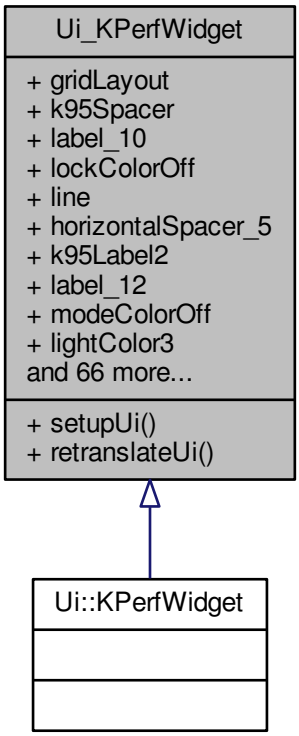
The documentation for this class was generated from the following file:

- src/ckb/ui_kbwidget.h

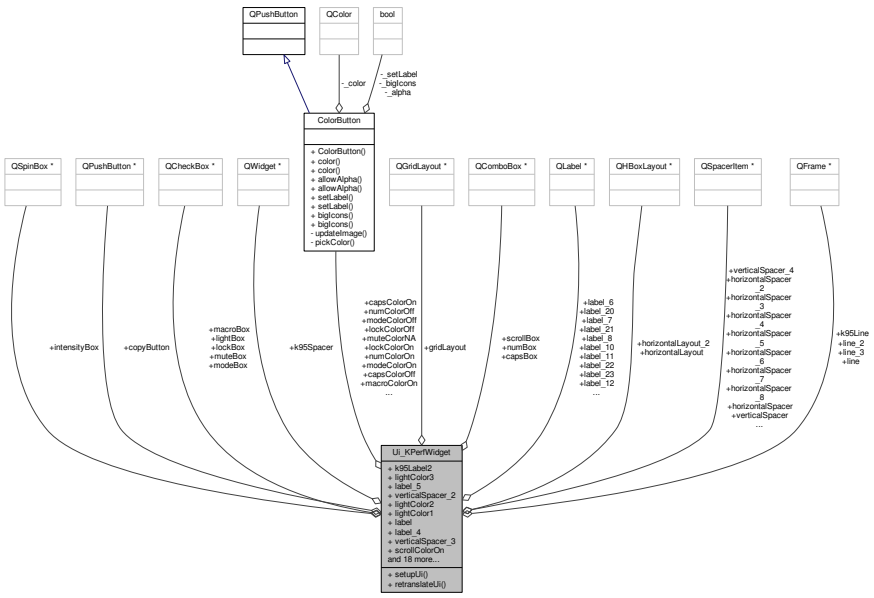
12.95 Ui_KPerfWidget Class Reference

`#include <src/ckb/ui_kperfwidget.h>`

Inheritance diagram for Ui_KPerfWidget:



Collaboration diagram for Ui_KPerfWidget:



Public Member Functions

- void setupUi (QWidget *KPerfWidget)
- void retranslateUi (QWidget *KPerfWidget)

Data Fields

- QGridLayout * gridLayout
- QWidget * k95Spacer
- QLabel * label_10
- ColorButton * lockColorOff
- QFrame * line
- QSpacerItem * horizontalSpacer_5
- QLabel * k95Label2
- QLabel * label_12
- ColorButton * modeColorOff
- ColorButton * lightColor3
- QSpacerItem * horizontalSpacer
- QCheckBox * lockBox
- ColorButton * modeColorOn
- QCheckBox * macroBox
- ColorButton * numColorOn
- QComboBox * numBox
- QLabel * label_5
- QSpacerItem * verticalSpacer_2
- ColorButton * lightColor2
- QSpacerItem * horizontalSpacer_7
- QLabel * label_8
- QCheckBox * lightBox
- ColorButton * lightColor1

- QSpacerItem * verticalSpacer_4
- QLabel * label_6
- QLabel * label
- QSpacerItem * horizontalSpacer_6
- ColorButton * lockColorOn
- QLabel * label_4
- QSpacerItem * verticalSpacer
- QSpacerItem * verticalSpacer_3
- QCheckBox * modeBox
- ColorButton * capsColorOff
- ColorButton * numColorOff
- ColorButton * capsColorOn
- QFrame * line_3
- QLabel * label_22
- QComboBox * scrollBox
- QLabel * label_7
- ColorButton * scrollColorOn
- QComboBox * capsBox
- ColorButton * scrollColorOff
- QCheckBox * muteBox
- ColorButton * muteColorOn
- ColorButton * macroColorOff
- QLabel * label_3
- ColorButton * muteColorOff
- QLabel * k95Label3
- ColorButton * macroColorOn
- ColorButton * muteColorNA
- QLabel * label_19
- QFrame * k95Line
- QLabel * label_21
- QLabel * label_24
- QLabel * label_20
- QLabel * label_16
- QLabel * label_14
- QLabel * label_18
- QSpacerItem * horizontalSpacer_2
- QLabel * label_15
- QLabel * k95Label4
- QLabel * label_17
- QLabel * label_11
- QLabel * label_23
- QLabel * k95Label1
- QHBoxLayout * horizontalLayout_2
- QPushButton * copyButton
- QSpacerItem * horizontalSpacer_8
- QHBoxLayout * horizontalLayout
- QLabel * label_2
- QSpacerItem * horizontalSpacer_4
- QSpinBox * intensityBox
- QSpacerItem * horizontalSpacer_3
- QLabel * k95Label5
- QFrame * line_2
- QLabel * k95Label6

12.95.1 Detailed Description

Definition at line 31 of file ui_kperfwidget.h.

12.95.2 Member Function Documentation

12.95.2.1 void Ui_KPerfWidget::retranslateUi (QWidget * *KPerfWidget*) [inline]

Definition at line 616 of file ui_kperfwidget.h.

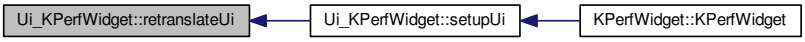
References capsBox, capsColorOff, capsColorOn, copyButton, intensityBox, k95Label1, k95Label2, k95Label3, k95Label4, k95Label5, k95Label6, label, label_10, label_11, label_12, label_14, label_15, label_16, label_17, label_18, label_19, label_2, label_20, label_21, label_22, label_23, label_24, label_3, label_4, label_5, label_6, label_7, label_8, lightBox, lightColor1, lightColor2, lightColor3, lockBox, lockColorOff, lockColorOn, macroBox, macroColorOff, macroColorOn, modeBox, modeColorOff, modeColorOn, muteBox, muteColorNA, muteColorOff, muteColorOn, numBox, numColorOff, numColorOn, scrollBox, scrollColorOff, and scrollColorOn.

Referenced by setupUi().

```
617     {
618         KPerfWidget->setWindowTitle(QApplication::translate("KPerfWidget", "Form", Q_NULLPTR));
619         label_10->setText(QApplication::translate("KPerfWidget", "100%", Q_NULLPTR));
620         lockColorOff->setText(QString());
621         k95Label2->setText(QApplication::translate("KPerfWidget", "On", Q_NULLPTR));
622         label_12->setText(QApplication::translate("KPerfWidget", "On", Q_NULLPTR));
623         modeColorOff->setText(QString());
624         lightColor3->setText(QString());
625         lockBox->setText(QApplication::translate("KPerfWidget", "Indicate Windows Lock:", Q_NULLPTR)
    );
626         modeColorOn->setText(QString());
627         macroBox->setText(QApplication::translate("KPerfWidget", "Indicate macro recording:",
    Q_NULLPTR));
628         numColorOn->setText(QString());
629         numBox->clear();
630         numBox->insertItems(0, QStringList()
631          << QApplication::translate("KPerfWidget", "Normal", Q_NULLPTR)
632          << QApplication::translate("KPerfWidget", "Always on", Q_NULLPTR)
633          << QApplication::translate("KPerfWidget", "Always off", Q_NULLPTR)
634          << QApplication::translate("KPerfWidget", "RGB indicator", Q_NULLPTR)
635          << QApplication::translate("KPerfWidget", "RGB + normal", Q_NULLPTR)
636         );
637         label_5->setText(QApplication::translate("KPerfWidget", "Num lock:", Q_NULLPTR));
638         lightColor2->setText(QString());
639         label_8->setText(QApplication::translate("KPerfWidget", "Note: many desktop environments do
     not use scroll lock. The indicator may not turn on even when enabled.", Q_NULLPTR));
640         lightBox->setText(QApplication::translate("KPerfWidget", "Indicate brightness:", Q_NULLPTR)
    );
641         lightColor1->setText(QString());
642         label_6->setText(QApplication::translate("KPerfWidget", "Caps lock:", Q_NULLPTR));
643         label->setText(QApplication::translate("KPerfWidget", "Miscellaneous", Q_NULLPTR));
644         lockColorOn->setText(QString());
645         label_4->setText(QApplication::translate("KPerfWidget", "Lock lights", Q_NULLPTR));
646         modeBox->setText(QApplication::translate("KPerfWidget", "Indicate current mode:", Q_NULLPTR)
    );
647         capsColorOff->setText(QString());
648         numColorOff->setText(QString());
649         capsColorOn->setText(QString());
650         label_22->setText(QApplication::translate("KPerfWidget", "Off", Q_NULLPTR));
651         scrollBox->clear();
652         scrollBox->insertItems(0, QStringList()
653          << QApplication::translate("KPerfWidget", "Normal", Q_NULLPTR)
654          << QApplication::translate("KPerfWidget", "Always on", Q_NULLPTR)
655          << QApplication::translate("KPerfWidget", "Always off", Q_NULLPTR)
656          << QApplication::translate("KPerfWidget", "RGB indicator", Q_NULLPTR)
657          << QApplication::translate("KPerfWidget", "RGB + normal", Q_NULLPTR)
658         );
659         label_7->setText(QApplication::translate("KPerfWidget", "Scroll lock:", Q_NULLPTR));
660         scrollColorOn->setText(QString());
661         capsBox->clear();
662         capsBox->insertItems(0, QStringList()
663          << QApplication::translate("KPerfWidget", "Normal", Q_NULLPTR)
664          << QApplication::translate("KPerfWidget", "Always on", Q_NULLPTR)
665          << QApplication::translate("KPerfWidget", "Always off", Q_NULLPTR)
666          << QApplication::translate("KPerfWidget", "RGB indicator", Q_NULLPTR)
667          << QApplication::translate("KPerfWidget", "RGB + normal", Q_NULLPTR)
668         );
669         scrollColorOff->setText(QString());
670         muteBox->setText(QApplication::translate("KPerfWidget", "Indicate mute:", Q_NULLPTR));
```

```
671         muteColorOn->setText(QString());
672         macroColorOff->setText(QString());
673         label_3->setText(QApplication::translate("KPerfWidget", "Top row", Q_NULLPTR));
674         muteColorOff->setText(QString());
675         k95Label3->setText(QApplication::translate("KPerfWidget", "Off", Q_NULLPTR));
676         macroColorOn->setText(QString());
677 #ifndef QT_NO_TOOLTIP
678         muteColorNA->setToolTip(QApplication::translate("KPerfWidget", "If the sound device does
    not support muting, or if mute status is unknown, this color will be displayed.", Q_NULLPTR));
679 #endif // QT_NO_TOOLTIP
680         muteColorNA->setText(QString());
681         label_19->setText(QApplication::translate("KPerfWidget", "On", Q_NULLPTR));
682         label_21->setText(QApplication::translate("KPerfWidget", "On", Q_NULLPTR));
683         label_24->setText(QApplication::translate("KPerfWidget", "Off", Q_NULLPTR));
684         label_20->setText(QApplication::translate("KPerfWidget", "On", Q_NULLPTR));
685         label_16->setText(QApplication::translate("KPerfWidget", "67%", Q_NULLPTR));
686         label_14->setText(QApplication::translate("KPerfWidget", "Off", Q_NULLPTR));
687         label_18->setText(QApplication::translate("KPerfWidget", "33%", Q_NULLPTR));
688         label_15->setText(QApplication::translate("KPerfWidget", "Off", Q_NULLPTR));
689         k95Label4->setText(QApplication::translate("KPerfWidget", "On", Q_NULLPTR));
690         label_17->setText(QApplication::translate("KPerfWidget", "Unknown", Q_NULLPTR));
691         label_11->setText(QApplication::translate("KPerfWidget", "On", Q_NULLPTR));
692         label_23->setText(QApplication::translate("KPerfWidget", "Off", Q_NULLPTR));
693         k95Label1->setText(QApplication::translate("KPerfWidget", "K95 buttons", Q_NULLPTR));
694         copyButton->setText(QApplication::translate("KPerfWidget", "Copy to mode...", Q_NULLPTR))
    ;
695         label_2->setText(QApplication::translate("KPerfWidget", "Indicator intensity:", Q_NULLPTR));
696         intensityBox->setSuffix(QApplication::translate("KPerfWidget", "%", Q_NULLPTR));
697         k95Label5->setText(QApplication::translate("KPerfWidget", "Off", Q_NULLPTR));
698         k95Label6->setText(QApplication::translate("KPerfWidget", "Note: macro recording not yet
    implemented. Indicator will always display \"off\".", Q_NULLPTR));
699     } // retranslateUi
```

Here is the caller graph for this function:



12.95.2.2 void Ui_KPerfWidget::setupUi (QWidget * *KPerfWidget*) [inline]

Definition at line 111 of file ui_kperfwidget.h.

References capsBox, capsColorOff, capsColorOn, copyButton, gridLayout, horizontalLayout, horizontalLayout_2, horizontalSpacer, horizontalSpacer_2, horizontalSpacer_3, horizontalSpacer_4, horizontalSpacer_5, horizontalSpacer_6, horizontalSpacer_7, horizontalSpacer_8, intensityBox, k95Label1, k95Label2, k95Label3, k95Label4, k95Label5, k95Label6, k95Line, k95Spacer, label, label_10, label_11, label_12, label_14, label_15, label_16, label_17, label_18, label_19, label_2, label_20, label_21, label_22, label_23, label_24, label_3, label_4, label_5, label_6, label_7, label_8, lightBox, lightColor1, lightColor2, lightColor3, line, line_2, line_3, lockBox, lockColorOff, lockColorOn, macroBox, macroColorOff, macroColorOn, modeBox, modeColorOff, modeColorOn, muteBox, muteColorNA, muteColorOff, muteColorOn, numBox, numColorOff, numColorOn, retranslateUi(), scrollBox, scrollColorOff, scrollColorOn, verticalSpacer, verticalSpacer_2, verticalSpacer_3, and verticalSpacer_4.

Referenced by KPerfWidget::KPerfWidget().

```
112     {
113         if (KPerfWidget->objectName().isEmpty())
114             KPerfWidget->setObjectName(QStringLiteral("KPerfWidget"));
115         KPerfWidget->resize(840, 679);
116         gridLayout = new QGridLayout(KPerfWidget);
117         gridLayout->setObjectName(QStringLiteral("gridLayout"));
118         k95Spacer = new QWidget(KPerfWidget);
119         k95Spacer->setObjectName(QStringLiteral("k95Spacer"));
120         QSizePolicy sizePolicy(QSizePolicy::Fixed, QSizePolicy::Fixed);
121         sizePolicy.setHorizontalStretch(0);
122         sizePolicy.setVerticalStretch(0);
123         sizePolicy.setHeightForWidth(k95Spacer->sizePolicy().hasHeightForWidth());
124         k95Spacer->setSizePolicy(sizePolicy);
125         k95Spacer->setMinimumSize(QSize(15, 10));
```

```
126         k95Spacer->setMaximumSize(QSize(15, 10));
127 
128         gridLayout->addWidget(k95Spacer, 12, 4, 1, 1);
129 
130         label_10 = new QLabel(KPerfWidget);
131         label_10->setObjectName(QStringLiteral("label_10"));
132 
133         gridLayout->addWidget(label_10, 15, 5, 1, 1);
134 
135         lockColorOff = new ColorButton(KPerfWidget);
136         lockColorOff->setObjectName(QStringLiteral("lockColorOff"));
137         sizePolicy.setHeightForWidth(lockColorOff->sizePolicy().hasHeightForWidth());
138         lockColorOff->setSizePolicy(sizePolicy);
139         lockColorOff->setMinimumSize(QSize(0, 40));
140         lockColorOff->setMaximumSize(QSize(16777215, 40));
141 
142         gridLayout->addWidget(lockColorOff, 16, 7, 1, 1);
143 
144         line = new QFrame(KPerfWidget);
145         line->setObjectName(QStringLiteral("line"));
146         line->setFrameShape(QFrame::HLine);
147         line->setFrameShadow(QFrame::Sunken);
148 
149         gridLayout->addWidget(line, 2, 0, 1, 13);
150 
151         horizontalSpacer_5 = new QSpacerItem(10, 20, QSizePolicy::Fixed,
    QSizePolicy::Minimum);
152 
153         gridLayout->addItem(horizontalSpacer_5, 17, 6, 1, 1);
154 
155         k95Label2 = new QLabel(KPerfWidget);
156         k95Label2->setObjectName(QStringLiteral("k95Label2"));
157 
158         gridLayout->addWidget(k95Label2, 9, 5, 1, 1);
159 
160         label_12 = new QLabel(KPerfWidget);
161         label_12->setObjectName(QStringLiteral("label_12"));
162 
163         gridLayout->addWidget(label_12, 17, 5, 1, 1);
164 
165         modeColorOff = new ColorButton(KPerfWidget);
166         modeColorOff->setObjectName(QStringLiteral("modeColorOff"));
167         sizePolicy.setHeightForWidth(modeColorOff->sizePolicy().hasHeightForWidth());
168         modeColorOff->setSizePolicy(sizePolicy);
169         modeColorOff->setMinimumSize(QSize(0, 40));
170         modeColorOff->setMaximumSize(QSize(16777215, 40));
171 
172         gridLayout->addWidget(modeColorOff, 9, 7, 1, 1);
173 
174         lightColor3 = new ColorButton(KPerfWidget);
175         lightColor3->setObjectName(QStringLiteral("lightColor3"));
176         sizePolicy.setHeightForWidth(lightColor3->sizePolicy().hasHeightForWidth());
177         lightColor3->setSizePolicy(sizePolicy);
178         lightColor3->setMinimumSize(QSize(0, 40));
179         lightColor3->setMaximumSize(QSize(16777215, 40));
180 
181         gridLayout->addWidget(lightColor3, 15, 4, 1, 1);
182 
183         horizontalSpacer = new QSpacerItem(40, 20, QSizePolicy::Expanding,
    QSizePolicy::Minimum);
184 
185         gridLayout->addItem(horizontalSpacer, 15, 12, 1, 1);
186 
187         lockBox = new QCheckBox(KPerfWidget);
188         lockBox->setObjectName(QStringLiteral("lockBox"));
189 
190         gridLayout->addWidget(lockBox, 16, 1, 1, 2);
191 
192         modeColorOn = new ColorButton(KPerfWidget);
193         modeColorOn->setObjectName(QStringLiteral("modeColorOn"));
194         sizePolicy.setHeightForWidth(modeColorOn->sizePolicy().hasHeightForWidth());
195         modeColorOn->setSizePolicy(sizePolicy);
196         modeColorOn->setMinimumSize(QSize(0, 40));
197         modeColorOn->setMaximumSize(QSize(16777215, 40));
198 
199         gridLayout->addWidget(modeColorOn, 9, 4, 1, 1);
200 
201         macroBox = new QCheckBox(KPerfWidget);
202         macroBox->setObjectName(QStringLiteral("macroBox"));
203 
204         gridLayout->addWidget(macroBox, 10, 1, 1, 2);
205 
206         numColorOn = new ColorButton(KPerfWidget);
207         numColorOn->setObjectName(QStringLiteral("numColorOn"));
208         sizePolicy.setHeightForWidth(numColorOn->sizePolicy().hasHeightForWidth());
209         numColorOn->setSizePolicy(sizePolicy);
210         numColorOn->setMinimumSize(QSize(0, 40));
```

```
211         numColorOn->setMaximumSize(QSize(16777215, 40));
212
213         gridLayout->addWidget(numColorOn, 21, 4, 1, 1);
214
215         numBox = new QComboBox(KPerfWidget);
216         numBox->setObjectName(QStringLiteral("numBox"));
217
218         gridLayout->addWidget(numBox, 21, 2, 1, 1);
219
220         label_5 = new QLabel(KPerfWidget);
221         label_5->setObjectName(QStringLiteral("label_5"));
222
223         gridLayout->addWidget(label_5, 21, 1, 1, 1);
224
225         verticalSpacer_2 = new QSpacerItem(20, 20, QSizePolicy::Minimum, QSizePolicy::Fixed
    );
226
227         gridLayout->addItem(verticalSpacer_2, 4, 1, 1, 1);
228
229         lightColor2 = new ColorButton(KPerfWidget);
230         lightColor2->setObjectName(QStringLiteral("lightColor2"));
231         sizePolicy.setHeightForWidth(lightColor2->sizePolicy().hasHeightForWidth());
232         lightColor2->setSizePolicy(sizePolicy);
233         lightColor2->setMinimumSize(QSize(0, 40));
234         lightColor2->setMaximumSize(QSize(16777215, 40));
235
236         gridLayout->addWidget(lightColor2, 15, 7, 1, 1);
237
238         horizontalSpacer_7 = new QSpacerItem(10, 20, QSizePolicy::Fixed,
    QSizePolicy::Minimum);
239
240         gridLayout->addItem(horizontalSpacer_7, 17, 3, 1, 1);
241
242         label_8 = new QLabel(KPerfWidget);
243         label_8->setObjectName(QStringLiteral("label_8"));
244         QFont font;
245         font.setPointSize(8);
246         font.setItalic(true);
247         label_8->setFont(font);
248
249         gridLayout->addWidget(label_8, 25, 1, 1, 12);
250
251         lightBox = new QCheckBox(KPerfWidget);
252         lightBox->setObjectName(QStringLiteral("lightBox"));
253
254         gridLayout->addWidget(lightBox, 15, 1, 1, 2);
255
256         lightColor1 = new ColorButton(KPerfWidget);
257         lightColor1->setObjectName(QStringLiteral("lightColor1"));
258         sizePolicy.setHeightForWidth(lightColor1->sizePolicy().hasHeightForWidth());
259         lightColor1->setSizePolicy(sizePolicy);
260         lightColor1->setMinimumSize(QSize(0, 40));
261         lightColor1->setMaximumSize(QSize(16777215, 40));
262
263         gridLayout->addWidget(lightColor1, 15, 10, 1, 1);
264
265         verticalSpacer_4 = new QSpacerItem(0, 5, QSizePolicy::Minimum, QSizePolicy::Fixed);
266
267         gridLayout->addItem(verticalSpacer_4, 24, 4, 1, 1);
268
269         label_6 = new QLabel(KPerfWidget);
270         label_6->setObjectName(QStringLiteral("label_6"));
271
272         gridLayout->addWidget(label_6, 22, 1, 1, 1);
273
274         label = new QLabel(KPerfWidget);
275         label->setObjectName(QStringLiteral("label"));
276         QFont font1;
277         font1.setBold(true);
278         font1.setWeight(75);
279         label->setFont(font1);
280
281         gridLayout->addWidget(label, 1, 0, 1, 13);
282
283         horizontalSpacer_6 = new QSpacerItem(10, 20, QSizePolicy::Fixed,
    QSizePolicy::Minimum);
284
285         gridLayout->addItem(horizontalSpacer_6, 17, 9, 1, 1);
286
287         lockColorOn = new ColorButton(KPerfWidget);
288         lockColorOn->setObjectName(QStringLiteral("lockColorOn"));
289         sizePolicy.setHeightForWidth(lockColorOn->sizePolicy().hasHeightForWidth());
290         lockColorOn->setSizePolicy(sizePolicy);
291         lockColorOn->setMinimumSize(QSize(0, 40));
292         lockColorOn->setMaximumSize(QSize(16777215, 40));
293
294         gridLayout->addWidget(lockColorOn, 16, 4, 1, 1);
```

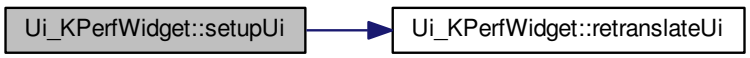
```
295
296         label_4 = new QLabel(KPerfWidget);
297         label_4->setObjectName(QStringLiteral("label_4"));
298         label_4->setFont(font1);
299
300         gridLayout->addWidget(label_4, 19, 0, 1, 13);
301
302         verticalSpacer = new QSpacerItem(20, 40, QSizePolicy::Minimum, QSizePolicy::Expanding
    );
303
304         gridLayout->addItem(verticalSpacer, 26, 2, 1, 1);
305
306         verticalSpacer_3 = new QSpacerItem(20, 20, QSizePolicy::Minimum, QSizePolicy::Fixed
    );
307
308         gridLayout->addItem(verticalSpacer_3, 18, 4, 1, 1);
309
310         modeBox = new QCheckBox(KPerfWidget);
311         modeBox->setObjectName(QStringLiteral("modeBox"));
312
313         gridLayout->addWidget(modeBox, 9, 1, 1, 2);
314
315         capsColorOff = new ColorButton(KPerfWidget);
316         capsColorOff->setObjectName(QStringLiteral("capsColorOff"));
317         sizePolicy.setHeightForWidth(capsColorOff->sizePolicy().hasHeightForWidth());
318         capsColorOff->setSizePolicy(sizePolicy);
319         capsColorOff->setMinimumSize(QSize(0, 40));
320         capsColorOff->setMaximumSize(QSize(16777215, 40));
321
322         gridLayout->addWidget(capsColorOff, 22, 7, 1, 1);
323
324         numColorOff = new ColorButton(KPerfWidget);
325         numColorOff->setObjectName(QStringLiteral("numColorOff"));
326         QSizePolicy sizePolicy1(QSizePolicy::Fixed, QSizePolicy::Fixed);
327         sizePolicy1.setHorizontalStretch(0);
328         sizePolicy1.setVerticalStretch(40);
329         sizePolicy1.setHeightForWidth(numColorOff->sizePolicy().hasHeightForWidth());
330         numColorOff->setSizePolicy(sizePolicy1);
331         numColorOff->setMaximumSize(QSize(16777215, 40));
332
333         gridLayout->addWidget(numColorOff, 21, 7, 1, 1);
334
335         capsColorOn = new ColorButton(KPerfWidget);
336         capsColorOn->setObjectName(QStringLiteral("capsColorOn"));
337         sizePolicy.setHeightForWidth(capsColorOn->sizePolicy().hasHeightForWidth());
338         capsColorOn->setSizePolicy(sizePolicy);
339         capsColorOn->setMinimumSize(QSize(0, 40));
340         capsColorOn->setMaximumSize(QSize(16777215, 40));
341
342         gridLayout->addWidget(capsColorOn, 22, 4, 1, 1);
343
344         line_3 = new QFrame(KPerfWidget);
345         line_3->setObjectName(QStringLiteral("line_3"));
346         line_3->setFrameShape(QFrame::HLine);
347         line_3->setFrameShadow(QFrame::Sunken);
348
349         gridLayout->addWidget(line_3, 20, 0, 1, 13);
350
351         label_22 = new QLabel(KPerfWidget);
352         label_22->setObjectName(QStringLiteral("label_22"));
353
354         gridLayout->addWidget(label_22, 21, 8, 1, 1);
355
356         scrollBox = new QComboBox(KPerfWidget);
357         scrollBox->setObjectName(QStringLiteral("scrollBox"));
358
359         gridLayout->addWidget(scrollBox, 23, 2, 1, 1);
360
361         label_7 = new QLabel(KPerfWidget);
362         label_7->setObjectName(QStringLiteral("label_7"));
363
364         gridLayout->addWidget(label_7, 23, 1, 1, 1);
365
366         scrollColorOn = new ColorButton(KPerfWidget);
367         scrollColorOn->setObjectName(QStringLiteral("scrollColorOn"));
368         sizePolicy.setHeightForWidth(scrollColorOn->sizePolicy().hasHeightForWidth());
369         scrollColorOn->setSizePolicy(sizePolicy);
370         scrollColorOn->setMinimumSize(QSize(0, 40));
371         scrollColorOn->setMaximumSize(QSize(16777215, 40));
372
373         gridLayout->addWidget(scrollColorOn, 23, 4, 1, 1);
374
375         capsBox = new QComboBox(KPerfWidget);
376         capsBox->setObjectName(QStringLiteral("capsBox"));
377
378         gridLayout->addWidget(capsBox, 22, 2, 1, 1);
379
```

```
380         scrollColorOff = new ColorButton(KPerfWidget);
381         scrollColorOff->setObjectName(QStringLiteral("scrollColorOff"));
382         sizePolicy.setHeightForWidth(scrollColorOff->sizePolicy().hasHeightForWidth());
383         scrollColorOff->setSizePolicy(sizePolicy);
384         scrollColorOff->setMinimumSize(QSize(0, 40));
385         scrollColorOff->setMaximumSize(QSize(16777215, 40));
386 
387         gridLayout->addWidget(scrollColorOff, 23, 7, 1, 1);
388 
389         muteBox = new QCheckBox(KPerfWidget);
390         muteBox->setObjectName(QStringLiteral("muteBox"));
391 
392         gridLayout->addWidget(muteBox, 17, 1, 1, 2);
393 
394         muteColorOn = new ColorButton(KPerfWidget);
395         muteColorOn->setObjectName(QStringLiteral("muteColorOn"));
396         sizePolicy.setHeightForWidth(muteColorOn->sizePolicy().hasHeightForWidth());
397         muteColorOn->setSizePolicy(sizePolicy);
398         muteColorOn->setMinimumSize(QSize(0, 40));
399         muteColorOn->setMaximumSize(QSize(16777215, 40));
400 
401         gridLayout->addWidget(muteColorOn, 17, 4, 1, 1);
402 
403         macroColorOff = new ColorButton(KPerfWidget);
404         macroColorOff->setObjectName(QStringLiteral("macroColorOff"));
405         QSizePolicy sizePolicy2(QSizePolicy::Minimum, QSizePolicy::Fixed);
406         sizePolicy2.setHorizontalStretch(0);
407         sizePolicy2.setVerticalStretch(40);
408         sizePolicy2.setHeightForWidth(macroColorOff->sizePolicy().hasHeightForWidth());
409         macroColorOff->setSizePolicy(sizePolicy2);
410         macroColorOff->setMinimumSize(QSize(0, 40));
411 
412         gridLayout->addWidget(macroColorOff, 10, 7, 1, 1);
413 
414         label_3 = new QLabel(KPerfWidget);
415         label_3->setObjectName(QStringLiteral("label_3"));
416         label_3->setFont(font1);
417 
418         gridLayout->addWidget(label_3, 13, 0, 1, 3);
419 
420         muteColorOff = new ColorButton(KPerfWidget);
421         muteColorOff->setObjectName(QStringLiteral("muteColorOff"));
422         sizePolicy.setHeightForWidth(muteColorOff->sizePolicy().hasHeightForWidth());
423         muteColorOff->setSizePolicy(sizePolicy);
424         muteColorOff->setMinimumSize(QSize(0, 40));
425         muteColorOff->setMaximumSize(QSize(16777215, 40));
426 
427         gridLayout->addWidget(muteColorOff, 17, 7, 1, 1);
428 
429         k95Label3 = new QLabel(KPerfWidget);
430         k95Label3->setObjectName(QStringLiteral("k95Label3"));
431 
432         gridLayout->addWidget(k95Label3, 9, 8, 1, 1);
433 
434         macroColorOn = new ColorButton(KPerfWidget);
435         macroColorOn->setObjectName(QStringLiteral("macroColorOn"));
436         sizePolicy2.setHeightForWidth(macroColorOn->sizePolicy().hasHeightForWidth());
437         macroColorOn->setSizePolicy(sizePolicy2);
438         macroColorOn->setMinimumSize(QSize(0, 40));
439 
440         gridLayout->addWidget(macroColorOn, 10, 4, 1, 1);
441 
442         muteColorNA = new ColorButton(KPerfWidget);
443         muteColorNA->setObjectName(QStringLiteral("muteColorNA"));
444         muteColorNA->setMinimumSize(QSize(0, 40));
445         muteColorNA->setMaximumSize(QSize(16777215, 40));
446 
447         gridLayout->addWidget(muteColorNA, 17, 10, 1, 1);
448 
449         label_19 = new QLabel(KPerfWidget);
450         label_19->setObjectName(QStringLiteral("label_19"));
451 
452         gridLayout->addWidget(label_19, 21, 5, 1, 1);
453 
454         k95Line = new QFrame(KPerfWidget);
455         k95Line->setObjectName(QStringLiteral("k95Line"));
456         k95Line->setFrameShape(QFrame::HLine);
457         k95Line->setFrameShadow(QFrame::Sunken);
458 
459         gridLayout->addWidget(k95Line, 8, 0, 1, 13);
460 
461         label_21 = new QLabel(KPerfWidget);
462         label_21->setObjectName(QStringLiteral("label_21"));
463 
464         gridLayout->addWidget(label_21, 23, 5, 1, 1);
465 
466         label_24 = new QLabel(KPerfWidget);
```

```
467         label_24->setObjectName(QStringLiteral("label_24"));
468
469         gridLayout->addWidget(label_24, 23, 8, 1, 1);
470
471         label_20 = new QLabel(KPerfWidget);
472         label_20->setObjectName(QStringLiteral("label_20"));
473
474         gridLayout->addWidget(label_20, 22, 5, 1, 1);
475
476         label_16 = new QLabel(KPerfWidget);
477         label_16->setObjectName(QStringLiteral("label_16"));
478
479         gridLayout->addWidget(label_16, 15, 8, 1, 1);
480
481         label_14 = new QLabel(KPerfWidget);
482         label_14->setObjectName(QStringLiteral("label_14"));
483
484         gridLayout->addWidget(label_14, 17, 8, 1, 1);
485
486         label_18 = new QLabel(KPerfWidget);
487         label_18->setObjectName(QStringLiteral("label_18"));
488
489         gridLayout->addWidget(label_18, 15, 11, 1, 1);
490
491         horizontalSpacer_2 = new QSpacerItem(5, 20, QSizePolicy::Fixed,
    QSizePolicy::Minimum);
492
493         gridLayout->addItem(horizontalSpacer_2, 15, 0, 1, 1);
494
495         label_15 = new QLabel(KPerfWidget);
496         label_15->setObjectName(QStringLiteral("label_15"));
497
498         gridLayout->addWidget(label_15, 16, 8, 1, 1);
499
500         k95Label4 = new QLabel(KPerfWidget);
501         k95Label4->setObjectName(QStringLiteral("k95Label4"));
502
503         gridLayout->addWidget(k95Label4, 10, 5, 1, 1);
504
505         label_17 = new QLabel(KPerfWidget);
506         label_17->setObjectName(QStringLiteral("label_17"));
507
508         gridLayout->addWidget(label_17, 17, 11, 1, 1);
509
510         label_11 = new QLabel(KPerfWidget);
511         label_11->setObjectName(QStringLiteral("label_11"));
512
513         gridLayout->addWidget(label_11, 16, 5, 1, 1);
514
515         label_23 = new QLabel(KPerfWidget);
516         label_23->setObjectName(QStringLiteral("label_23"));
517
518         gridLayout->addWidget(label_23, 22, 8, 1, 1);
519
520         k95Label1 = new QLabel(KPerfWidget);
521         k95Label1->setObjectName(QStringLiteral("k95Label1"));
522         k95Label1->setFont(font1);
523
524         gridLayout->addWidget(k95Label1, 7, 0, 1, 5);
525
526         horizontalLayout_2 = new QHBoxLayout();
527         horizontalLayout_2->setObjectName(QStringLiteral("horizontalLayout_2"));
528         copyButton = new QPushButton(KPerfWidget);
529         copyButton->setObjectName(QStringLiteral("copyButton"));
530
531         horizontalLayout_2->addWidget(copyButton);
532
533         horizontalSpacer_8 = new QSpacerItem(40, 20, QSizePolicy::Expanding,
    QSizePolicy::Minimum);
534
535         horizontalLayout_2->addItem(horizontalSpacer_8);
536
537
538         gridLayout->addLayout(horizontalLayout_2, 3, 10, 1, 3);
539
540         horizontalLayout = new QHBoxLayout();
541         horizontalLayout->setObjectName(QStringLiteral("horizontalLayout"));
542         label_2 = new QLabel(KPerfWidget);
543         label_2->setObjectName(QStringLiteral("label_2"));
544
545         horizontalLayout->addWidget(label_2);
546
547         horizontalSpacer_4 = new QSpacerItem(10, 20, QSizePolicy::Fixed,
    QSizePolicy::Minimum);
548
549         horizontalLayout->addItem(horizontalSpacer_4);
550
```

```
551         intensityBox = new QSpinBox(KPerfWidget);
552         intensityBox->setObjectName(QStringLiteral("intensityBox"));
553         intensityBox->setMaximum(100);
554         intensityBox->setValue(100);
555 
556         horizontalLayout->addWidget(intensityBox);
557 
558         horizontalSpacer_3 = new QSpacerItem(40, 20, QSizePolicy::Expanding,
    QSizePolicy::Minimum);
559 
560         horizontalLayout->addItem(horizontalSpacer_3);
561 
562 
563         gridLayout->addLayout(horizontalLayout, 3, 1, 1, 9);
564 
565         k95Label5 = new QLabel(KPerfWidget);
566         k95Label5->setObjectName(QStringLiteral("k95Label5"));
567 
568         gridLayout->addWidget(k95Label5, 10, 8, 1, 1);
569 
570         line_2 = new QFrame(KPerfWidget);
571         line_2->setObjectName(QStringLiteral("line_2"));
572         line_2->setFrameShape(QFrame::HLine);
573         line_2->setFrameShadow(QFrame::Sunken);
574 
575         gridLayout->addWidget(line_2, 14, 0, 1, 13);
576 
577         k95Label6 = new QLabel(KPerfWidget);
578         k95Label6->setObjectName(QStringLiteral("k95Label6"));
579         k95Label6->setFont(font);
580 
581         gridLayout->addWidget(k95Label6, 11, 1, 1, 10);
582 
583         QWidget::setTabOrder(intensityBox, copyButton);
584         QWidget::setTabOrder(copyButton, modeBox);
585         QWidget::setTabOrder(modeBox, modeColorOn);
586         QWidget::setTabOrder(modeColorOn, modeColorOff);
587         QWidget::setTabOrder(modeColorOff, macroBox);
588         QWidget::setTabOrder(macroBox, macroColorOn);
589         QWidget::setTabOrder(macroColorOn, macroColorOff);
590         QWidget::setTabOrder(macroColorOff, lightBox);
591         QWidget::setTabOrder(lightBox, lightColor3);
592         QWidget::setTabOrder(lightColor3, lightColor2);
593         QWidget::setTabOrder(lightColor2, lightColor1);
594         QWidget::setTabOrder(lightColor1, lockBox);
595         QWidget::setTabOrder(lockBox, lockColorOn);
596         QWidget::setTabOrder(lockColorOn, lockColorOff);
597         QWidget::setTabOrder(lockColorOff, muteBox);
598         QWidget::setTabOrder(muteBox, muteColorOn);
599         QWidget::setTabOrder(muteColorOn, muteColorOff);
600         QWidget::setTabOrder(muteColorOff, muteColorNA);
601         QWidget::setTabOrder(muteColorNA, numBox);
602         QWidget::setTabOrder(numBox, numColorOn);
603         QWidget::setTabOrder(numColorOn, numColorOff);
604         QWidget::setTabOrder(numColorOff, capsBox);
605         QWidget::setTabOrder(capsBox, capsColorOn);
606         QWidget::setTabOrder(capsColorOn, capsColorOff);
607         QWidget::setTabOrder(capsColorOff, scrollBox);
608         QWidget::setTabOrder(scrollBox, scrollColorOn);
609         QWidget::setTabOrder(scrollColorOn, scrollColorOff);
610 
611         retranslateUi(KPerfWidget);
612 
613         QMetaObject::connectSlotsByName(KPerfWidget);
614     } // setupUi
```

Here is the call graph for this function:



Here is the caller graph for this function:

12.95.3 Field Documentation

12.95.3.1 QComboBox* Ui_KPerfWidget::capsBox

Definition at line 74 of file ui_kperfwidget.h.

Referenced by KPerfWidget::KPerfWidget(), retranslateUi(), and setupUi().

12.95.3.2 ColorButton* Ui_KPerfWidget::capsColorOff

Definition at line 66 of file ui_kperfwidget.h.

Referenced by KPerfWidget::KPerfWidget(), retranslateUi(), and setupUi().

12.95.3.3 ColorButton* Ui_KPerfWidget::capsColorOn

Definition at line 68 of file ui_kperfwidget.h.

Referenced by KPerfWidget::KPerfWidget(), retranslateUi(), and setupUi().

12.95.3.4 QPushButton* Ui_KPerfWidget::copyButton

Definition at line 100 of file ui_kperfwidget.h.

Referenced by retranslateUi(), and setupUi().

12.95.3.5 QGridLayout* Ui_KPerfWidget::gridLayout

Definition at line 34 of file ui_kperfwidget.h.

Referenced by setupUi().

12.95.3.6 QHBoxLayout* Ui_KPerfWidget::horizontalLayout

Definition at line 102 of file ui_kperfwidget.h.

Referenced by setupUi().

12.95.3.7 QHBoxLayout* Ui_KPerfWidget::horizontalLayout_2

Definition at line 99 of file ui_kperfwidget.h.

Referenced by setupUi().

12.95.3.8 QSpacerItem* Ui_KPerfWidget::horizontalSpacer

Definition at line 44 of file ui_kperfwidget.h.

Referenced by setupUi().

12.95.3.9 QSpacerItem* Ui_KPerfWidget::horizontalSpacer_2

Definition at line 92 of file ui_kperfwidget.h.

Referenced by setupUi().

12.95.3.10 QSpacerItem* Ui_KPerfWidget::horizontalSpacer_3

Definition at line 106 of file ui_kperfwidget.h.

Referenced by setupUi().

12.95.3.11 QSpacerItem* Ui_KPerfWidget::horizontalSpacer_4

Definition at line 104 of file ui_kperfwidget.h.

Referenced by setupUi().

12.95.3.12 QSpacerItem* Ui_KPerfWidget::horizontalSpacer_5

Definition at line 39 of file ui_kperfwidget.h.

Referenced by setupUi().

12.95.3.13 QSpacerItem* Ui_KPerfWidget::horizontalSpacer_6

Definition at line 60 of file ui_kperfwidget.h.

Referenced by setupUi().

12.95.3.14 QSpacerItem* Ui_KPerfWidget::horizontalSpacer_7

Definition at line 53 of file ui_kperfwidget.h.

Referenced by setupUi().

12.95.3.15 QSpacerItem* Ui_KPerfWidget::horizontalSpacer_8

Definition at line 101 of file ui_kperfwidget.h.

Referenced by setupUi().

12.95.3.16 QSpinBox* Ui_KPerfWidget::intensityBox

Definition at line 105 of file ui_kperfwidget.h.

Referenced by retranslateUi(), KPerfWidget::setPerf(), and setupUi().

12.95.3.17 QLabel* Ui_KPerfWidget::k95Label1

Definition at line 98 of file ui_kperfwidget.h.

Referenced by KPerfWidget::KPerfWidget(), retranslateUi(), and setupUi().

12.95.3.18 QLabel* Ui_KPerfWidget::k95Label2

Definition at line 40 of file ui_kperfwidget.h.

Referenced by KPerfWidget::KPerfWidget(), retranslateUi(), and setupUi().

12.95.3.19 QLabel* Ui_KPerfWidget::k95Label3

Definition at line 81 of file ui_kperfwidget.h.

Referenced by KPerfWidget::KPerfWidget(), retranslateUi(), and setupUi().

12.95.3.20 QLabel* Ui_KPerfWidget::k95Label4

Definition at line 94 of file ui_kperfwidget.h.

Referenced by KPerfWidget::KPerfWidget(), retranslateUi(), and setupUi().

12.95.3.21 QLabel* Ui_KPerfWidget::k95Label5

Definition at line 107 of file ui_kperfwidget.h.

Referenced by KPerfWidget::KPerfWidget(), retranslateUi(), and setupUi().

12.95.3.22 QLabel* Ui_KPerfWidget::k95Label6

Definition at line 109 of file ui_kperfwidget.h.

Referenced by KPerfWidget::KPerfWidget(), retranslateUi(), and setupUi().

12.95.3.23 QFrame* Ui_KPerfWidget::k95Line

Definition at line 85 of file ui_kperfwidget.h.

Referenced by KPerfWidget::KPerfWidget(), and setupUi().

12.95.3.24 QWidget* Ui_KPerfWidget::k95Spacer

Definition at line 35 of file ui_kperfwidget.h.

Referenced by KPerfWidget::KPerfWidget(), and setupUi().

12.95.3.25 QLabel* Ui_KPerfWidget::label

Definition at line 59 of file ui_kperfwidget.h.

Referenced by retranslateUi(), and setupUi().

12.95.3.26 QLabel* Ui_KPerfWidget::label_10

Definition at line 36 of file ui_kperfwidget.h.

Referenced by retranslateUi(), and setupUi().

12.95.3.27 QLabel* Ui_KPerfWidget::label_11

Definition at line 96 of file ui_kperfwidget.h.

Referenced by retranslateUi(), and setupUi().

12.95.3.28 QLabel* Ui_KPerfWidget::label_12

Definition at line 41 of file ui_kperfwidget.h.

Referenced by retranslateUi(), KPerfWidget::setPerf(), and setupUi().

12.95.3.29 QLabel* Ui_KPerfWidget::label_14

Definition at line 90 of file ui_kperfwidget.h.

Referenced by retranslateUi(), KPerfWidget::setPerf(), and setupUi().

12.95.3.30 QLabel* Ui_KPerfWidget::label_15

Definition at line 93 of file ui_kperfwidget.h.

Referenced by retranslateUi(), and setupUi().

12.95.3.31 QLabel* Ui_KPerfWidget::label_16

Definition at line 89 of file ui_kperfwidget.h.

Referenced by retranslateUi(), and setupUi().

12.95.3.32 QLabel* Ui_KPerfWidget::label_17

Definition at line 95 of file ui_kperfwidget.h.

Referenced by retranslateUi(), KPerfWidget::setPerf(), and setupUi().

12.95.3.33 QLabel* Ui_KPerfWidget::label_18

Definition at line 91 of file ui_kperfwidget.h.

Referenced by retranslateUi(), and setupUi().

12.95.3.34 QLabel* Ui_KPerfWidget::label_19

Definition at line 84 of file ui_kperfwidget.h.

Referenced by retranslateUi(), and setupUi().

12.95.3.35 QLabel* Ui_KPerfWidget::label_2

Definition at line 103 of file ui_kperfwidget.h.

Referenced by retranslateUi(), and setupUi().

12.95.3.36 QLabel* Ui_KPerfWidget::label_20

Definition at line 88 of file ui_kperfwidget.h.

Referenced by retranslateUi(), and setupUi().

12.95.3.37 QLabel* Ui_KPerfWidget::label_21

Definition at line 86 of file ui_kperfwidget.h.

Referenced by retranslateUi(), and setupUi().

12.95.3.38 QLabel* Ui_KPerfWidget::label_22

Definition at line 70 of file ui_kperfwidget.h.

Referenced by retranslateUi(), and setupUi().

12.95.3.39 QLabel* Ui_KPerfWidget::label_23

Definition at line 97 of file ui_kperfwidget.h.

Referenced by retranslateUi(), and setupUi().

12.95.3.40 QLabel* Ui_KPerfWidget::label_24

Definition at line 87 of file ui_kperfwidget.h.

Referenced by retranslateUi(), and setupUi().

12.95.3.41 QLabel* Ui_KPerfWidget::label_3

Definition at line 79 of file ui_kperfwidget.h.

Referenced by retranslateUi(), and setupUi().

12.95.3.42 QLabel* Ui_KPerfWidget::label_4

Definition at line 62 of file ui_kperfwidget.h.

Referenced by retranslateUi(), and setupUi().

12.95.3.43 QLabel* Ui_KPerfWidget::label_5

Definition at line 50 of file ui_kperfwidget.h.

Referenced by retranslateUi(), and setupUi().

12.95.3.44 QLabel* Ui_KPerfWidget::label_6

Definition at line 58 of file ui_kperfwidget.h.

Referenced by retranslateUi(), and setupUi().

12.95.3.45 QLabel* Ui_KPerfWidget::label_7

Definition at line 72 of file ui_kperfwidget.h.

Referenced by retranslateUi(), and setupUi().

12.95.3.46 QLabel* Ui_KPerfWidget::label_8

Definition at line 54 of file ui_kperfwidget.h.

Referenced by retranslateUi(), and setupUi().

12.95.3.47 QCheckBox* Ui_KPerfWidget::lightBox

Definition at line 55 of file ui_kperfwidget.h.

Referenced by KPerfWidget::KPerfWidget(), retranslateUi(), and setupUi().

12.95.3.48 ColorButton* Ui_KPerfWidget::lightColor1

Definition at line 56 of file ui_kperfwidget.h.

Referenced by KPerfWidget::KPerfWidget(), retranslateUi(), and setupUi().

12.95.3.49 ColorButton* Ui_KPerfWidget::lightColor2

Definition at line 52 of file ui_kperfwidget.h.

Referenced by KPerfWidget::KPerfWidget(), retranslateUi(), and setupUi().

12.95.3.50 ColorButton* Ui_KPerfWidget::lightColor3

Definition at line 43 of file ui_kperfwidget.h.

Referenced by KPerfWidget::KPerfWidget(), retranslateUi(), and setupUi().

12.95.3.51 QFrame* Ui_KPerfWidget::line

Definition at line 38 of file ui_kperfwidget.h.

Referenced by setupUi().

12.95.3.52 QFrame* Ui_KPerfWidget::line_2

Definition at line 108 of file ui_kperfwidget.h.

Referenced by setupUi().

12.95.3.53 QFrame* Ui_KPerfWidget::line_3

Definition at line 69 of file ui_kperfwidget.h.

Referenced by setupUi().

12.95.3.54 QCheckBox* Ui_KPerfWidget::lockBox

Definition at line 45 of file ui_kperfwidget.h.

Referenced by KPerfWidget::KPerfWidget(), retranslateUi(), and setupUi().

12.95.3.55 ColorButton* Ui_KPerfWidget::lockColorOff

Definition at line 37 of file ui_kperfwidget.h.

Referenced by KPerfWidget::KPerfWidget(), retranslateUi(), and setupUi().

12.95.3.56 ColorButton* Ui_KPerfWidget::lockColorOn

Definition at line 61 of file ui_kperfwidget.h.

Referenced by KPerfWidget::KPerfWidget(), retranslateUi(), and setupUi().

12.95.3.57 QCheckBox* Ui_KPerfWidget::macroBox

Definition at line 47 of file ui_kperfwidget.h.

Referenced by KPerfWidget::KPerfWidget(), retranslateUi(), and setupUi().

12.95.3.58 ColorButton* Ui_KPerfWidget::macroColorOff

Definition at line 78 of file ui_kperfwidget.h.

Referenced by KPerfWidget::KPerfWidget(), retranslateUi(), and setupUi().

12.95.3.59 ColorButton* Ui_KPerfWidget::macroColorOn

Definition at line 82 of file ui_kperfwidget.h.

Referenced by KPerfWidget::KPerfWidget(), retranslateUi(), and setupUi().

12.95.3.60 QCheckBox* Ui_KPerfWidget::modeBox

Definition at line 65 of file ui_kperfwidget.h.

Referenced by KPerfWidget::KPerfWidget(), retranslateUi(), and setupUi().

12.95.3.61 ColorButton* Ui_KPerfWidget::modeColorOff

Definition at line 42 of file ui_kperfwidget.h.

Referenced by KPerfWidget::KPerfWidget(), retranslateUi(), and setupUi().

12.95.3.62 ColorButton* Ui_KPerfWidget::modeColorOn

Definition at line 46 of file ui_kperfwidget.h.

Referenced by KPerfWidget::KPerfWidget(), retranslateUi(), and setupUi().

12.95.3.63 QCheckBox* Ui_KPerfWidget::muteBox

Definition at line 76 of file ui_kperfwidget.h.

Referenced by KPerfWidget::KPerfWidget(), retranslateUi(), KPerfWidget::setPerf(), and setupUi().

12.95.3.64 ColorButton* Ui_KPerfWidget::muteColorNA

Definition at line 83 of file ui_kperfwidget.h.

Referenced by KPerfWidget::KPerfWidget(), retranslateUi(), KPerfWidget::setPerf(), and setupUi().

12.95.3.65 ColorButton* Ui_KPerfWidget::muteColorOff

Definition at line 80 of file ui_kperfwidget.h.

Referenced by KPerfWidget::KPerfWidget(), retranslateUi(), KPerfWidget::setPerf(), and setupUi().

12.95.3.66 ColorButton* Ui_KPerfWidget::muteColorOn

Definition at line 77 of file ui_kperfwidget.h.

Referenced by KPerfWidget::KPerfWidget(), retranslateUi(), KPerfWidget::setPerf(), and setupUi().

12.95.3.67 QComboBox* Ui_KPerfWidget::numBox

Definition at line 49 of file ui_kperfwidget.h.

Referenced by KPerfWidget::KPerfWidget(), retranslateUi(), and setupUi().

12.95.3.68 ColorButton* Ui_KPerfWidget::numColorOff

Definition at line 67 of file ui_kperfwidget.h.

Referenced by KPerfWidget::KPerfWidget(), retranslateUi(), and setupUi().

12.95.3.69 ColorButton* Ui_KPerfWidget::numColorOn

Definition at line 48 of file ui_kperfwidget.h.

Referenced by KPerfWidget::KPerfWidget(), retranslateUi(), and setupUi().

12.95.3.70 QComboBox* Ui_KPerfWidget::scrollBox

Definition at line 71 of file ui_kperfwidget.h.

Referenced by KPerfWidget::KPerfWidget(), retranslateUi(), and setupUi().

12.95.3.71 ColorButton* Ui_KPerfWidget::scrollColorOff

Definition at line 75 of file ui_kperfwidget.h.

Referenced by KPerfWidget::KPerfWidget(), retranslateUi(), and setupUi().

12.95.3.72 ColorButton* Ui_KPerfWidget::scrollColorOn

Definition at line 73 of file ui_kperfwidget.h.

Referenced by KPerfWidget::KPerfWidget(), retranslateUi(), and setupUi().

12.95.3.73 QSpacerItem* Ui_KPerfWidget::verticalSpacer

Definition at line 63 of file ui_kperfwidget.h.

Referenced by setupUi().

12.95.3.74 QSpacerItem* Ui_KPerfWidget::verticalSpacer_2

Definition at line 51 of file ui_kperfwidget.h.

Referenced by setupUi().

12.95.3.75 QSpacerItem* Ui_KPerfWidget::verticalSpacer_3

Definition at line 64 of file ui_kperfwidget.h.

Referenced by setupUi().

12.95.3.76 QSpacerItem* Ui_KPerfWidget::verticalSpacer_4

Definition at line 57 of file ui_kperfwidget.h.

Referenced by setupUi().

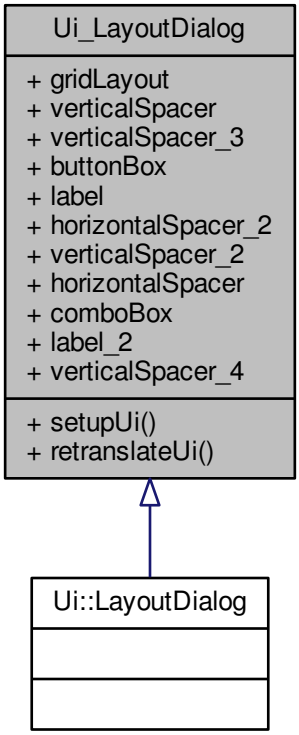
The documentation for this class was generated from the following file:

- src/ckb/ui_kperfwidget.h

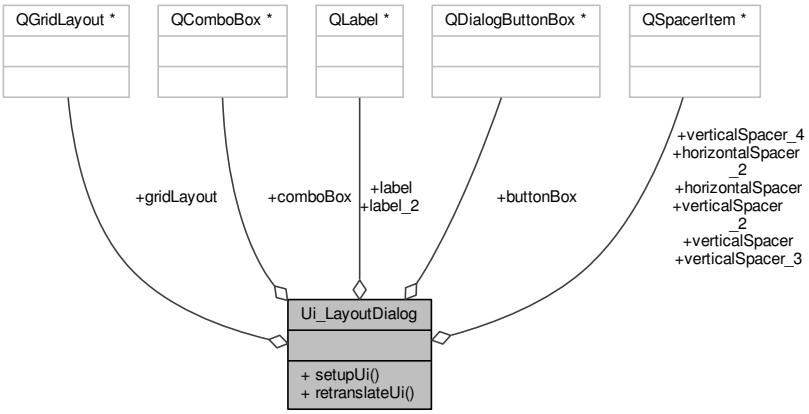
12.96 Ui_LayoutDialog Class Reference

`#include <src/ckb/ui_layoutdialog.h>`

Inheritance diagram for Ui_LayoutDialog:



Collaboration diagram for Ui_LayoutDialog:



Public Member Functions

- void setupUi (QDialog *LayoutDialog)

- void retranslateUi (QDialog *LayoutDialog)

Data Fields

- QGridLayout * gridLayout
- QSpacerItem * verticalSpacer
- QSpacerItem * verticalSpacer_3
- QDialogButtonBox * buttonBox
- QLabel * label
- QSpacerItem * horizontalSpacer_2
- QSpacerItem * verticalSpacer_2
- QSpacerItem * horizontalSpacer
- QComboBox * comboBox
- QLabel * label_2
- QSpacerItem * verticalSpacer_4

12.96.1 Detailed Description

Definition at line 26 of file ui_layoutdialog.h.

12.96.2 Member Function Documentation

12.96.2.1 void Ui_LayoutDialog::retranslateUi (QDialog * *LayoutDialog*) [inline]

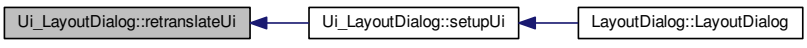
Definition at line 109 of file ui_layoutdialog.h.

References label, and label_2.

Referenced by setupUi().

```
110     {
111         LayoutDialog->setWindowTitle(QApplication::translate("LayoutDialog", "Keyboard Layout", Q_NULLPTR))
    ;
112         label->setText(QApplication::translate("LayoutDialog", "Please select your keyboard layout:",
    Q_NULLPTR));
113         label_2->setText(QApplication::translate("LayoutDialog", "Note: this only controls the
     layout within the ckb application. If your keys are typing the wrong characters, check your system preferences.",
     Q_NULLPTR));
114     } // retranslateUi
```

Here is the caller graph for this function:



12.96.2.2 void Ui_LayoutDialog::setupUi (QDialog * *LayoutDialog*) [inline]

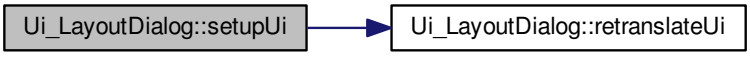
Definition at line 41 of file ui_layoutdialog.h.

References buttonBox, comboBox, gridLayout, horizontalSpacer, horizontalSpacer_2, label, label_2, retranslateUi(), verticalSpacer, verticalSpacer_2, verticalSpacer_3, and verticalSpacer_4.

Referenced by LayoutDialog::LayoutDialog().

```
42      {
43          if (LayoutDialog->objectName().isEmpty())
44              LayoutDialog->setObjectName(QStringLiteral("LayoutDialog"));
45          LayoutDialog->resize(400, 240);
46          gridLayout = new QGridLayout(LayoutDialog);
47          gridLayout->setObjectName(QStringLiteral("gridLayout"));
48          verticalSpacer = new QSpacerItem(20, 40, QSizePolicy::Minimum, QSizePolicy::Expanding
    );
49
50          gridLayout->addItem(verticalSpacer, 6, 0, 1, 3);
51
52          verticalSpacer_3 = new QSpacerItem(20, 40, QSizePolicy::Minimum,
    QSizePolicy::Expanding);
53
54          gridLayout->addItem(verticalSpacer_3, 0, 1, 1, 1);
55
56          buttonBox = new QDialogButtonBox(LayoutDialog);
57          buttonBox->setObjectName(QStringLiteral("buttonBox"));
58          buttonBox->setOrientation(Qt::Horizontal);
59          buttonBox->setStandardButtons(QDialogButtonBox::Ok);
60
61          gridLayout->addWidget(buttonBox, 7, 0, 1, 3);
62
63          label = new QLabel(LayoutDialog);
64          label->setObjectName(QStringLiteral("label"));
65          label->setAlignment(Qt::AlignCenter);
66
67          gridLayout->addWidget(label, 1, 0, 1, 3);
68
69          horizontalSpacer_2 = new QSpacerItem(40, 20, QSizePolicy::Expanding,
    QSizePolicy::Minimum);
70
71          gridLayout->addItem(horizontalSpacer_2, 3, 2, 1, 1);
72
73          verticalSpacer_2 = new QSpacerItem(20, 15, QSizePolicy::Minimum, QSizePolicy::Fixed
    );
74
75          gridLayout->addItem(verticalSpacer_2, 4, 1, 1, 1);
76
77          horizontalSpacer = new QSpacerItem(40, 20, QSizePolicy::Expanding,
    QSizePolicy::Minimum);
78
79          gridLayout->addItem(horizontalSpacer, 3, 0, 1, 1);
80
81          comboBox = new QComboBox(LayoutDialog);
82          comboBox->setObjectName(QStringLiteral("comboBox"));
83
84          gridLayout->addWidget(comboBox, 3, 1, 1, 1);
85
86          label_2 = new QLabel(LayoutDialog);
87          label_2->setObjectName(QStringLiteral("label_2"));
88          QFont font;
89          font.setPointSize(8);
90          font.setItalic(true);
91          label_2->setFont(font);
92          label_2->setAlignment(Qt::AlignCenter);
93          label_2->setWordWrap(true);
94
95          gridLayout->addWidget(label_2, 5, 0, 1, 3);
96
97          verticalSpacer_4 = new QSpacerItem(20, 5, QSizePolicy::Minimum, QSizePolicy::Fixed)
    ;
98
99          gridLayout->addItem(verticalSpacer_4, 2, 1, 1, 1);
100
101
102          retranslateUi(LayoutDialog);
103          QObject::connect(buttonBox, SIGNAL(accepted()), LayoutDialog, SLOT(accept()));
104          QObject::connect(buttonBox, SIGNAL(rejected()), LayoutDialog, SLOT(reject()));
105
106          QMetaObject::connectSlotsByName(LayoutDialog);
107      } // setupUi
```

Here is the call graph for this function:



Here is the caller graph for this function:

12.96.3 Field Documentation

12.96.3.1 QDialogButtonBox* Ui_LayoutDialog::buttonBox

Definition at line 32 of file ui_layoutdialog.h.

Referenced by setupUi().

12.96.3.2 QComboBox* Ui_LayoutDialog::comboBox

Definition at line 37 of file ui_layoutdialog.h.

Referenced by LayoutDialog::LayoutDialog(), LayoutDialog::selected(), and setupUi().

12.96.3.3 QGridLayout* Ui_LayoutDialog::gridLayout

Definition at line 29 of file ui_layoutdialog.h.

Referenced by setupUi().

12.96.3.4 QSpacerItem* Ui_LayoutDialog::horizontalSpacer

Definition at line 36 of file ui_layoutdialog.h.

Referenced by setupUi().

12.96.3.5 QSpacerItem* Ui_LayoutDialog::horizontalSpacer_2

Definition at line 34 of file ui_layoutdialog.h.

Referenced by setupUi().

12.96.3.6 QLabel* Ui_LayoutDialog::label

Definition at line 33 of file ui_layoutdialog.h.

Referenced by retranslateUi(), and setupUi().

12.96.3.7 QLabel* Ui_LayoutDialog::label_2

Definition at line 38 of file ui_layoutdialog.h.

Referenced by retranslateUi(), and setupUi().

12.96.3.8 QSpacerItem* Ui_LayoutDialog::verticalSpacer

Definition at line 30 of file ui_layoutdialog.h.

Referenced by setupUi().

12.96.3.9 QSpacerItem* Ui_LayoutDialog::verticalSpacer_2

Definition at line 35 of file ui_layoutdialog.h.

Referenced by setupUi().

12.96.3.10 QSpacerItem* Ui_LayoutDialog::verticalSpacer_3

Definition at line 31 of file ui_layoutdialog.h.

Referenced by setupUi().

12.96.3.11 QSpacerItem* Ui_LayoutDialog::verticalSpacer_4

Definition at line 39 of file ui_layoutdialog.h.

Referenced by setupUi().

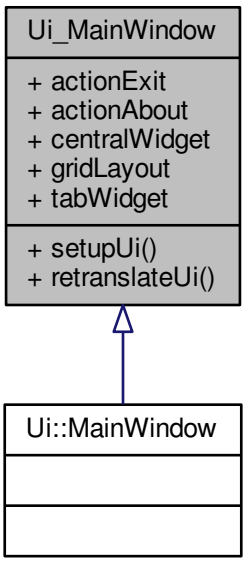
The documentation for this class was generated from the following file:

- src/ckb/ui_layoutdialog.h

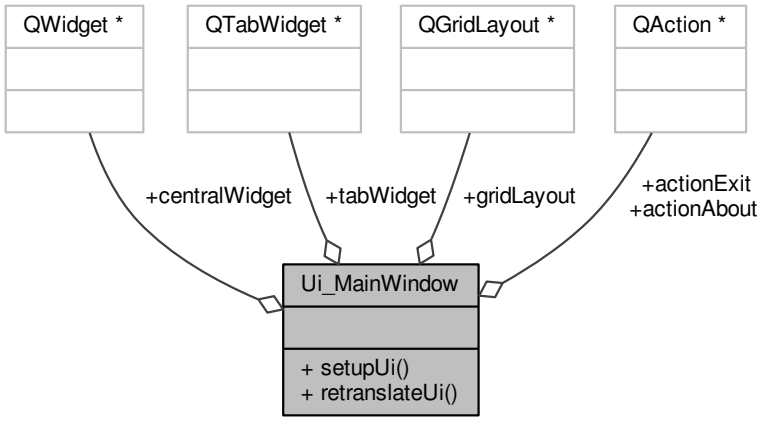
12.97 Ui_MainWindow Class Reference

`#include <src/ckb/ui_mainwindow.h>`

Inheritance diagram for Ui_MainWindow:



Collaboration diagram for Ui_MainWindow:



Public Member Functions

- void setupUi (QMainWindow *MainWindow)
- void retranslateUi (QMainWindow *MainWindow)

Data Fields

- QAction * actionExit
- QAction * actionAbout
- QWidget * centralWidget
- QGridLayout * gridLayout
- QTabWidget * tabWidget

12.97.1 Detailed Description

Definition at line 24 of file ui_mainwindow.h.

12.97.2 Member Function Documentation

12.97.2.1 void Ui_MainWindow::retranslateUi (QMainWindow * *MainWindow*) [inline]

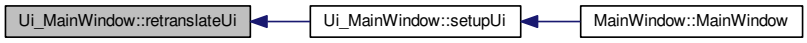
Definition at line 68 of file ui_mainwindow.h.

References actionAbout, and actionExit.

Referenced by setupUi().

```
69     {
70         MainWindow->setWindowTitle(QApplication::translate("MainWindow", "ckb-next", Q_NULLPTR));
71         actionExit->setText(QApplication::translate("MainWindow", "Exit", Q_NULLPTR));
72         actionExit->setShortcut(QApplication::translate("MainWindow", "Ctrl+Q", Q_NULLPTR));
73         actionAbout->setText(QApplication::translate("MainWindow", "About", Q_NULLPTR));
74     } // retranslateUi
```

Here is the caller graph for this function:



12.97.2.2 void Ui_MainWindow::setupUi (QMainWindow * *MainWindow*) [inline]

Definition at line 33 of file ui_mainwindow.h.

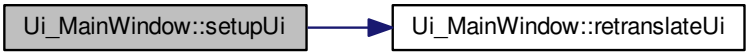
References actionAbout, actionExit, centralWidget, gridLayout, retranslateUi(), and tabWidget.

Referenced by MainWindow::MainWindow().

```
34     {
35         if (MainWindow->objectName().isEmpty())
36             MainWindow->setObjectName(QStringLiteral("MainWindow"));
37         MainWindow->resize(1000, 750);
38         MainWindow->setMinimumSize(QSize(1000, 750));
39         QIcon icon;
40         icon.addFile(QStringLiteral(":/img/ckb-logo.png"), QSize(), QIcon::Normal, QIcon::Off);
41         MainWindow->setWindowIcon(icon);
42         MainWindow->setUnifiedTitleAndToolBarOnMac(true);
43         actionExit = new QAction(MainWindow);
44         actionExit->setObjectName(QStringLiteral("actionExit"));
45         actionAbout = new QAction(MainWindow);
46         actionAbout->setObjectName(QStringLiteral("actionAbout"));
47         centralWidget = new QWidget(MainWindow);
48         centralWidget->setObjectName(QStringLiteral("centralWidget"));
49         gridLayout = new QGridLayout(centralWidget);
50         gridLayout->setSpacing(6);
```

```
51          gridLayout->setContentsMargins(11, 11, 11, 11);
52          gridLayout->setObjectName(QStringLiteral("gridLayout"));
53          tabWidget = new QTabWidget(centralWidget);
54          tabWidget->setObjectName(QStringLiteral("tabWidget"));
55
56          gridLayout->addWidget(tabWidget, 0, 0, 1, 1);
57
58          MainWindow->setCentralWidget(centralWidget);
59
60          retranslateUi(MainWindow);
61
62          tabWidget->setCurrentIndex(-1);
63
64
65          QMetaObject::connectSlotsByName(MainWindow);
66      } // setupUi
```

Here is the call graph for this function:



Here is the caller graph for this function:

12.97.3 Field Documentation

12.97.3.1 QAction* Ui_MainWindow::actionAbout

Definition at line 28 of file ui_mainwindow.h.

Referenced by retranslateUi(), and setupUi().

12.97.3.2 QAction* Ui_MainWindow::actionExit

Definition at line 27 of file ui_mainwindow.h.

Referenced by MainWindow::MainWindow(), retranslateUi(), and setupUi().

12.97.3.3 QWidget* Ui_MainWindow::centralWidget

Definition at line 29 of file ui_mainwindow.h.

Referenced by setupUi().

12.97.3.4 QGridLayout* Ui_MainWindow::gridLayout

Definition at line 30 of file ui_mainwindow.h.

Referenced by setupUi().

12.97.3.5 QTabWidget* Ui_MainWindow::tabWidget

Definition at line 31 of file ui_mainwindow.h.

Referenced by MainWindow::addDevice(), MainWindow::MainWindow(), MainWindow::removeDevice(), setupUi(), and MainWindow::showFwUpdateNotification().

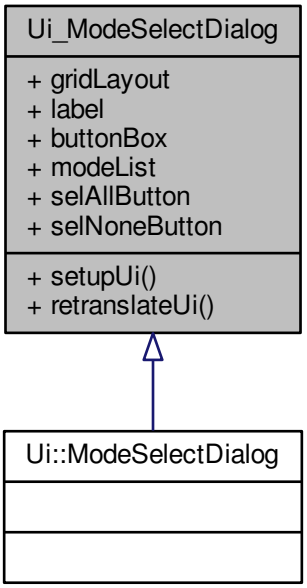
The documentation for this class was generated from the following file:

- src/ckb/ui_mainwindow.h

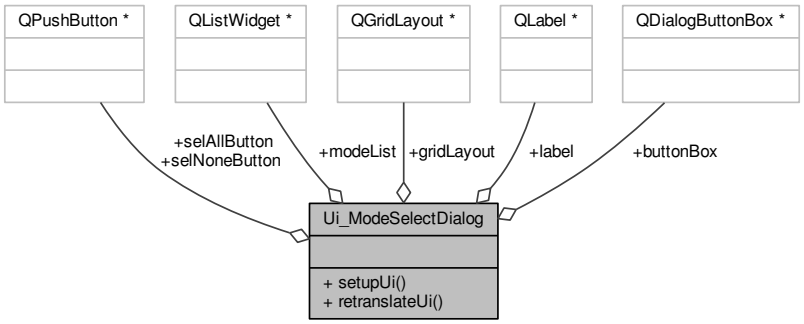
12.98 Ui_ModeSelectDialog Class Reference

`#include <src/ckb/ui_modeselectdialog.h>`

Inheritance diagram for Ui_ModeSelectDialog:



Collaboration diagram for Ui_ModeSelectDialog:



Public Member Functions

- void setupUi (QDialog *ModeSelectDialog)
- void retranslateUi (QDialog *ModeSelectDialog)

Data Fields

- QGridLayout * gridLayout
- QLabel * label
- QDialogButtonBox * buttonBox
- QListWidget * modeList
- QPushButton * selAllButton
- QPushButton * selNoneButton

12.98.1 Detailed Description

Definition at line 26 of file ui_modeselectdialog.h.

12.98.2 Member Function Documentation

12.98.2.1 void Ui_ModeSelectDialog::retranslateUi (QDialog * *ModeSelectDialog*) [inline]

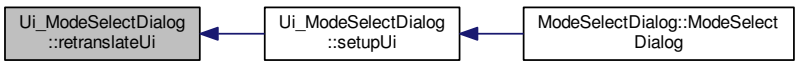
Definition at line 84 of file ui_modeselectdialog.h.

References label, selAllButton, and selNoneButton.

Referenced by setupUi().

```
85     {
86         ModeSelectDialog->setWindowTitle(QApplication::translate("ModeSelectDialog", "Select Modes",
      Q_NULLPTR));
87         label->setText(QApplication::translate("ModeSelectDialog", "TextLabel", Q_NULLPTR));
88         selAllButton->setText(QApplication::translate("ModeSelectDialog", "Select All",
      Q_NULLPTR));
89         selNoneButton->setText(QApplication::translate("ModeSelectDialog", "Select None",
      Q_NULLPTR));
90     } // retranslateUi
```

Here is the caller graph for this function:



12.98.2.2 void Ui_ModeSelectDialog::setupUi (QDialog * *ModeSelectDialog*) [inline]

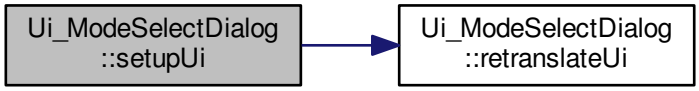
Definition at line 36 of file ui_modeselectdialog.h.

References buttonBox, gridLayout, label, modeList, retranslateUi(), selAllButton, and selNoneButton.

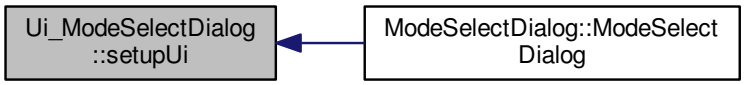
Referenced by ModeSelectDialog::ModeSelectDialog().

```
37     {
38         if (ModeSelectDialog->objectName().isEmpty())
39             ModeSelectDialog->setObjectName(QStringLiteral("ModeSelectDialog"));
40         ModeSelectDialog->resize(320, 320);
41         gridLayout = new QGridLayout(ModeSelectDialog);
42         gridLayout->setObjectName(QStringLiteral("gridLayout"));
43         label = new QLabel(ModeSelectDialog);
44         label->setObjectName(QStringLiteral("label"));
45         label->setAlignment(Qt::AlignLeading|Qt::AlignLeft|Qt::AlignTop);
46 
47         gridLayout->addWidget(label, 0, 0, 1, 2);
48 
49         buttonBox = new QDialogButtonBox(ModeSelectDialog);
50         buttonBox->setObjectName(QStringLiteral("buttonBox"));
51         buttonBox->setOrientation(Qt::Vertical);
52         buttonBox->setStandardButtons(QDialogButtonBox::Cancel|QDialogButtonBox::Ok);
53         buttonBox->setCenterButtons(false);
54 
55         gridLayout->addWidget(buttonBox, 0, 2, 3, 1);
56 
57         modeList = new QListWidget(ModeSelectDialog);
58         modeList->setObjectName(QStringLiteral("modeList"));
59         modeList->setFrameShadow(QFrame::Plain);
60         modeList->setSelectionMode(QAbstractItemView::ExtendedSelection);
61 
62         gridLayout->addWidget(modeList, 1, 0, 1, 2);
63 
64         selAllButton = new QPushButton(ModeSelectDialog);
65         selAllButton->setObjectName(QStringLiteral("selAllButton"));
66 
67         gridLayout->addWidget(selAllButton, 2, 0, 1, 1);
68 
69         selNoneButton = new QPushButton(ModeSelectDialog);
70         selNoneButton->setObjectName(QStringLiteral("selNoneButton"));
71 
72         gridLayout->addWidget(selNoneButton, 2, 1, 1, 1);
73 
74         QWidget::setTabOrder(modeList, selAllButton);
75         QWidget::setTabOrder(selAllButton, selNoneButton);
76 
77         retranslateUi(ModeSelectDialog);
78         QObject::connect(buttonBox, SIGNAL(accepted()), ModeSelectDialog, SLOT(accept()));
79         QObject::connect(buttonBox, SIGNAL(rejected()), ModeSelectDialog, SLOT(reject()));
80 
81         QMetaObject::connectSlotsByName(ModeSelectDialog);
82     } // setupUi
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.98.3 Field Documentation

12.98.3.1 QDialogButtonBox* Ui_ModeSelectDialog::buttonBox

Definition at line 31 of file ui_modeselectdialog.h.

Referenced by setupUi().

12.98.3.2 QGridLayout* Ui_ModeSelectDialog::gridLayout

Definition at line 29 of file ui_modeselectdialog.h.

Referenced by setupUi().

12.98.3.3 QLabel* Ui_ModeSelectDialog::label

Definition at line 30 of file ui_modeselectdialog.h.

Referenced by ModeSelectDialog::ModeSelectDialog(), retranslateUi(), and setupUi().

12.98.3.4 QListWidget* Ui_ModeSelectDialog::modeList

Definition at line 32 of file ui_modeselectdialog.h.

Referenced by ModeSelectDialog::ModeSelectDialog(), ModeSelectDialog::on_selAllButton_clicked(), ModeSelectDialog::on_selNoneButton_clicked(), ModeSelectDialog::selection(), and setupUi().

12.98.3.5 QPushButton* Ui_ModeSelectDialog::selAllButton

Definition at line 33 of file ui_modeselectdialog.h.

Referenced by retranslateUi(), and setupUi().

12.98.3.6 QPushButton* Ui_ModeSelectDialog::selNoneButton

Definition at line 34 of file ui_modeselectdialog.h.

Referenced by retranslateUi(), and setupUi().

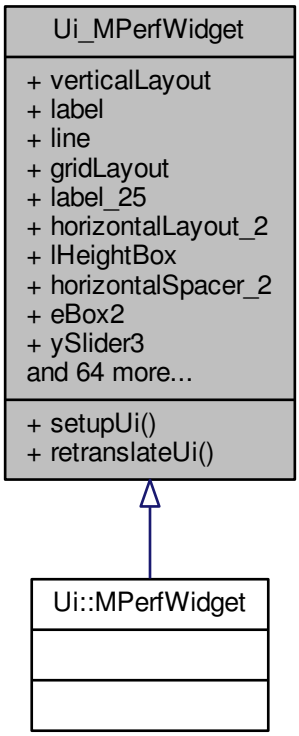
The documentation for this class was generated from the following file:

- src/ckb/ui_modeselectdialog.h

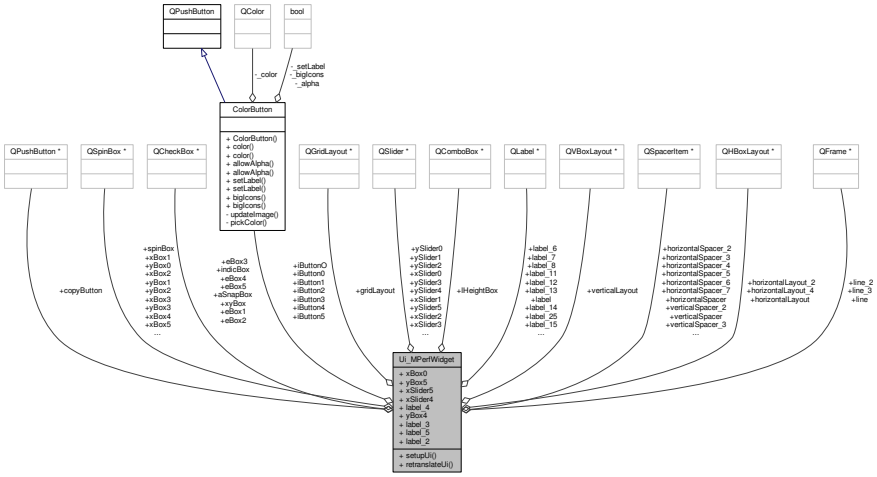
12.99 Ui_MPerfWidget Class Reference

`#include <src/ckb/ui_mperfwidget.h>`

Inheritance diagram for Ui_MPerfWidget:



Collaboration diagram for Ui_MPerfWidget:



Public Member Functions

- void setupUi (QWidget *MPerfWidget)
- void retranslateUi (QWidget *MPerfWidget)

Data Fields

- QVBoxLayout * verticalLayout
- QLabel * label
- QFrame * line
- QGridLayout * gridLayout
- QLabel * label_25
- QHBoxLayout * horizontalLayout_2
- QComboBox * lHeightBox
- QSpacerItem * horizontalSpacer_2
- QCheckBox * eBox2
- QSlider * ySlider3
- QSlider * xSlider2
- QSpinBox * xBox3
- QSpinBox * yBox0
- QSlider * ySlider0
- ColorButton * iButton4
- ColorButton * iButton5
- QSpacerItem * horizontalSpacer_4
- QSpacerItem * verticalSpacer_3
- QSpacerItem * horizontalSpacer_5
- ColorButton * iButton3
- QLabel * label_12
- QSpinBox * xBox4
- QSpinBox * xBox0
- QSpinBox * yBox5
- QCheckBox * eBox5
- QSlider * xSlider5
- QSpinBox * xBox1

- QSpinBox * yBox2
- QSpinBox * yBox3
- QSlider * xSlider4
- QLabel * label_4
- QSlider * xSlider0
- ColorButton * iButton2
- QFrame * line_3
- ColorButton * iButtonO
- QSpinBox * yBox1
- QSlider * ySlider2
- QSpinBox * xBox2
- QSlider * xSlider1
- QSpacerItem * horizontalSpacer
- QSpinBox * yBox4
- ColorButton * iButton1
- QSlider * ySlider4
- QSlider * ySlider1
- ColorButton * iButton0
- QSlider * xSlider3
- QSpinBox * xBox5
- QLabel * label_3
- QLabel * label_6
- QSlider * ySlider5
- QLabel * label_7
- QLabel * label_8
- QLabel * label_5
- QLabel * label_15
- QLabel * label_13
- QLabel * label_14
- QCheckBox * eBox4
- QLabel * label_11
- QCheckBox * eBox3
- QCheckBox * eBox1
- QLabel * label_2
- QSpacerItem * verticalSpacer
- QCheckBox * aSnapBox
- QFrame * line_2
- QHBoxLayout * horizontalLayout_4
- QCheckBox * indicBox
- QSpacerItem * horizontalSpacer_3
- QSpinBox * spinBox
- QSpacerItem * horizontalSpacer_7
- QCheckBox * xyBox
- QSpacerItem * verticalSpacer_2
- QHBoxLayout * horizontalLayout
- QPushButton * copyButton
- QSpacerItem * horizontalSpacer_6

12.99.1 Detailed Description

Definition at line 33 of file ui_mperfwidget.h.

12.99.2 Member Function Documentation

12.99.2.1 void Ui_MPerfWidget::retranslateUi (QWidget * *MPerfWidget*) [inline]

Definition at line 682 of file ui_mperfwidget.h.

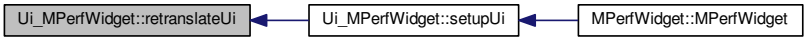
References aSnapBox, copyButton, eBox1, eBox2, eBox3, eBox4, eBox5, iButton0, iButton1, iButton2, iButton3, iButton4, iButton5, iButtonO, indicBox, label, label_11, label_12, label_13, label_14, label_15, label_2, label_25, label_3, label_4, label_5, label_6, label_7, label_8, lHeightBox, spinBox, and xyBox.

Referenced by setupUi().

```
683     {
684         MPerfWidget->setWindowTitle(QApplication::translate("MPerfWidget", "Form", Q_NULLPTR));
685         label->setText(QApplication::translate("MPerfWidget", "DPI", Q_NULLPTR));
686         label_25->setText(QApplication::translate("MPerfWidget", "Lift height:", Q_NULLPTR));
687         lHeightBox->clear();
688         lHeightBox->insertItems(0, QStringList()
689          << QApplication::translate("MPerfWidget", "Low", Q_NULLPTR)
690          << QApplication::translate("MPerfWidget", "Medium-low", Q_NULLPTR)
691          << QApplication::translate("MPerfWidget", "Medium", Q_NULLPTR)
692          << QApplication::translate("MPerfWidget", "Medium-high", Q_NULLPTR)
693          << QApplication::translate("MPerfWidget", "High", Q_NULLPTR)
694         );
695 #ifndef QT_NO_TOOLTIP
696         eBox2->setToolTip(QApplication::translate("MPerfWidget", "Uncheck this box to skip the stage
       while moving up or down.", Q_NULLPTR));
697 #endif // QT_NO_TOOLTIP
698         eBox2->setText(QString());
699 #ifndef QT_NO_TOOLTIP
700         iButton4->setToolTip(QApplication::translate("MPerfWidget", "Tip: alt+click to set all
       colors at once", Q_NULLPTR));
701 #endif // QT_NO_TOOLTIP
702         iButton4->setText(QString());
703 #ifndef QT_NO_TOOLTIP
704         iButton5->setToolTip(QApplication::translate("MPerfWidget", "Tip: alt+click to set all
       colors at once", Q_NULLPTR));
705 #endif // QT_NO_TOOLTIP
706         iButton5->setText(QString());
707 #ifndef QT_NO_TOOLTIP
708         iButton3->setToolTip(QApplication::translate("MPerfWidget", "Tip: alt+click to set all
       colors at once", Q_NULLPTR));
709 #endif // QT_NO_TOOLTIP
710         iButton3->setText(QString());
711         label_12->setText(QApplication::translate("MPerfWidget", "X Value:", Q_NULLPTR));
712 #ifndef QT_NO_TOOLTIP
713         eBox5->setToolTip(QApplication::translate("MPerfWidget", "Uncheck this box to skip the stage
       while moving up or down.", Q_NULLPTR));
714 #endif // QT_NO_TOOLTIP
715         eBox5->setText(QString());
716         label_4->setText(QApplication::translate("MPerfWidget", "Y Value:", Q_NULLPTR));
717 #ifndef QT_NO_TOOLTIP
718         iButton2->setToolTip(QApplication::translate("MPerfWidget", "Tip: alt+click to set all
       colors at once", Q_NULLPTR));
719 #endif // QT_NO_TOOLTIP
720         iButton2->setText(QString());
721 #ifndef QT_NO_TOOLTIP
722         iButtonO->setToolTip(QApplication::translate("MPerfWidget", "Tip: alt+click to set all
       colors at once", Q_NULLPTR));
723 #endif // QT_NO_TOOLTIP
724         iButtonO->setText(QString());
725 #ifndef QT_NO_TOOLTIP
726         iButton1->setToolTip(QApplication::translate("MPerfWidget", "Tip: alt+click to set all
       colors at once", Q_NULLPTR));
727 #endif // QT_NO_TOOLTIP
728         iButton1->setText(QString());
729 #ifndef QT_NO_TOOLTIP
730         iButton0->setToolTip(QApplication::translate("MPerfWidget", "Tip: alt+click to set all
       colors at once", Q_NULLPTR));
731 #endif // QT_NO_TOOLTIP
732         iButton0->setText(QString());
733         label_3->setText(QApplication::translate("MPerfWidget", "You can assign other DPIs in the
       Binding tab.", Q_NULLPTR));
734         label_6->setText(QApplication::translate("MPerfWidget", "2:", Q_NULLPTR));
735         label_7->setText(QApplication::translate("MPerfWidget", "3:", Q_NULLPTR));
736         label_8->setText(QApplication::translate("MPerfWidget", "4:", Q_NULLPTR));
737         label_5->setText(QApplication::translate("MPerfWidget", "1:", Q_NULLPTR));
738         label_15->setText(QApplication::translate("MPerfWidget", "Other:", Q_NULLPTR));
739         label_13->setText(QApplication::translate("MPerfWidget", "5:", Q_NULLPTR));
740         label_14->setText(QApplication::translate("MPerfWidget", "Sniper:", Q_NULLPTR));
741 #ifndef QT_NO_TOOLTIP
742         eBox4->setToolTip(QApplication::translate("MPerfWidget", "Uncheck this box to skip the stage
       while moving up or down.", Q_NULLPTR));
```

```
743 #endif // QT_NO_TOOLTIP
744         eBox4->setText(QString());
745         label_11->setText(QApplication::translate("MPerfWidget", "DPI Stage:", Q_NULLPTR));
746 #ifndef QT_NO_TOOLTIP
747         eBox3->setToolTip(QApplication::translate("MPerfWidget", "Uncheck this box to skip the stage
      while moving up or down.", Q_NULLPTR));
748 #endif // QT_NO_TOOLTIP
749         eBox3->setText(QString());
750 #ifndef QT_NO_TOOLTIP
751         eBox1->setToolTip(QApplication::translate("MPerfWidget", "Uncheck this box to skip the stage
      while moving up or down.", Q_NULLPTR));
752 #endif // QT_NO_TOOLTIP
753         eBox1->setText(QString());
754         label_2->setText(QApplication::translate("MPerfWidget", "Movement", Q_NULLPTR));
755         aSnapBox->setText(QApplication::translate("MPerfWidget", "Angle snap", Q_NULLPTR));
756         indicBox->setText(QApplication::translate("MPerfWidget", "Use DPI indicator:", Q_NULLPTR));
757 #ifndef QT_NO_TOOLTIP
758         spinBox->setToolTip(QApplication::translate("MPerfWidget", "Indicator intensity", Q_NULLPTR)
    );
759 #endif // QT_NO_TOOLTIP
760         spinBox->setSuffix(QApplication::translate("MPerfWidget", "%", Q_NULLPTR));
761         xyBox->setText(QApplication::translate("MPerfWidget", "Independent X/Y states", Q_NULLPTR));
762         copyButton->setText(QApplication::translate("MPerfWidget", "Copy to mode...", Q_NULLPTR))
    ;
763     } // retranslateUi
```

Here is the caller graph for this function:



12.99.2.2 void Ui_MPerfWidget::setupUi (QWidget * *MPerfWidget*) [inline]

Definition at line 111 of file ui_mperfwidget.h.

References aSnapBox, copyButton, eBox1, eBox2, eBox3, eBox4, eBox5, gridLayout, horizontalLayout, horizontalLayout_2, horizontalLayout_4, horizontalSpacer, horizontalSpacer_2, horizontalSpacer_3, horizontalSpacer_4, horizontalSpacer_5, horizontalSpacer_6, horizontalSpacer_7, iButton0, iButton1, iButton2, iButton3, iButton4, iButton5, iButtonO, indicBox, label, label_11, label_12, label_13, label_14, label_15, label_2, label_25, label_3, label_4, label_5, label_6, label_7, label_8, lHeightBox, line, line_2, line_3, retranslateUi(), spinBox, verticalLayout, verticalSpacer, verticalSpacer_2, verticalSpacer_3, xBox0, xBox1, xBox2, xBox3, xBox4, xBox5, xSlider0, xSlider1, xSlider2, xSlider3, xSlider4, xSlider5, xyBox, yBox0, yBox1, yBox2, yBox3, yBox4, yBox5, ySlider0, ySlider1, ySlider2, ySlider3, ySlider4, and ySlider5.

Referenced by MPerfWidget::MPerfWidget().

```
112     {
113         if (MPerfWidget->objectName().isEmpty())
114             MPerfWidget->setObjectName(QStringLiteral("MPerfWidget"));
115         MPerfWidget->resize(932, 637);
116         verticalLayout = new QVBoxLayout(MPerfWidget);
117         verticalLayout->setObjectName(QStringLiteral("verticalLayout"));
118         label = new QLabel(MPerfWidget);
119         label->setObjectName(QStringLiteral("label"));
120         QFont font;
121         font.setBold(true);
122         font.setWeight(75);
123         label->setFont(font);
124 
125         verticalLayout->addWidget(label);
126 
127         line = new QFrame(MPerfWidget);
128         line->setObjectName(QStringLiteral("line"));
129         line->setFrameShape(QFrame::HLine);
130         line->setFrameShadow(QFrame::Sunken);
131 
132         verticalLayout->addWidget(line);
133 
134         gridLayout = new QGridLayout();
```

```
135         gridLayout->setObjectName(QStringLiteral("gridLayout"));
136         label_25 = new QLabel(MPerfWidget);
137         label_25->setObjectName(QStringLiteral("label_25"));
138         label_25->setAlignment(Qt::AlignRight|Qt::AlignTrailing|Qt::AlignVCenter);
139 
140         gridLayout->addWidget(label_25, 16, 5, 1, 2);
141 
142         horizontalLayout_2 = new QHBoxLayout();
143         horizontalLayout_2->setObjectName(QStringLiteral("horizontalLayout_2"));
144         lHeightBox = new QComboBox(MPerfWidget);
145         lHeightBox->setObjectName(QStringLiteral("lHeightBox"));
146 
147         horizontalLayout_2->addWidget(lHeightBox);
148 
149         horizontalSpacer_2 = new QSpacerItem(40, 20, QSizePolicy::Expanding,
    QSizePolicy::Minimum);
150 
151         horizontalLayout_2->addItem(horizontalSpacer_2);
152 
153 
154         gridLayout->addLayout(horizontalLayout_2, 16, 8, 1, 3);
155 
156         eBox2 = new QCheckBox(MPerfWidget);
157         eBox2->setObjectName(QStringLiteral("eBox2"));
158 
159         gridLayout->addWidget(eBox2, 7, 1, 1, 1);
160 
161         ySlider3 = new QSlider(MPerfWidget);
162         ySlider3->setObjectName(QStringLiteral("ySlider3"));
163         ySlider3->setMinimum(0);
164         ySlider3->setMaximum(1000);
165         ySlider3->setSingleStep(10);
166         ySlider3->setPageStep(50);
167         ySlider3->setValue(0);
168         ySlider3->setOrientation(Qt::Horizontal);
169 
170         gridLayout->addWidget(ySlider3, 8, 8, 1, 1);
171 
172         xSlider2 = new QSlider(MPerfWidget);
173         xSlider2->setObjectName(QStringLiteral("xSlider2"));
174         xSlider2->setMinimum(0);
175         xSlider2->setMaximum(1000);
176         xSlider2->setSingleStep(10);
177         xSlider2->setPageStep(50);
178         xSlider2->setValue(0);
179         xSlider2->setOrientation(Qt::Horizontal);
180 
181         gridLayout->addWidget(xSlider2, 7, 5, 1, 1);
182 
183         xBox3 = new QSpinBox(MPerfWidget);
184         xBox3->setObjectName(QStringLiteral("xBox3"));
185         xBox3->setMinimum(100);
186         xBox3->setMaximum(12000);
187         xBox3->setSingleStep(50);
188 
189         gridLayout->addWidget(xBox3, 8, 6, 1, 1);
190 
191         yBox0 = new QSpinBox(MPerfWidget);
192         yBox0->setObjectName(QStringLiteral("yBox0"));
193         yBox0->setMinimum(100);
194         yBox0->setMaximum(12000);
195         yBox0->setSingleStep(50);
196 
197         gridLayout->addWidget(yBox0, 11, 9, 1, 1);
198 
199         ySlider0 = new QSlider(MPerfWidget);
200         ySlider0->setObjectName(QStringLiteral("ySlider0"));
201         ySlider0->setMinimum(0);
202         ySlider0->setMaximum(1000);
203         ySlider0->setSingleStep(10);
204         ySlider0->setPageStep(50);
205         ySlider0->setValue(0);
206         ySlider0->setOrientation(Qt::Horizontal);
207 
208         gridLayout->addWidget(ySlider0, 11, 8, 1, 1);
209 
210         iButton4 = new ColorButton(MPerfWidget);
211         iButton4->setObjectName(QStringLiteral("iButton4"));
212         QSizePolicy sizePolicy(QSizePolicy::Fixed, QSizePolicy::Fixed);
213         sizePolicy.setHorizontalStretch(0);
214         sizePolicy.setVerticalStretch(0);
215         sizePolicy.setHeightForWidth(iButton4->sizePolicy().hasHeightForWidth());
216         iButton4->setSizePolicy(sizePolicy);
217         iButton4->setMinimumSize(QSize(0, 40));
218 
219         gridLayout->addWidget(iButton4, 9, 2, 1, 1);
220 
```

```
221         iButton5 = new ColorButton(MPerfWidget);
222         iButton5->setObjectName(QStringLiteral("iButton5"));
223         sizePolicy.setHeightForWidth(iButton5->sizePolicy().hasHeightForWidth());
224         iButton5->setSizePolicy(sizePolicy);
225         iButton5->setMinimumSize(QSize(0, 40));
226 
227         gridLayout->addWidget(iButton5, 10, 2, 1, 1);
228 
229         horizontalSpacer_4 = new QSpacerItem(5, 20, QSizePolicy::Fixed,
    QSizePolicy::Minimum);
230 
231         gridLayout->addItem(horizontalSpacer_4, 4, 4, 1, 1);
232 
233         verticalSpacer_3 = new QSpacerItem(20, 15, QSizePolicy::Minimum, QSizePolicy::Fixed
    );
234 
235         gridLayout->addItem(verticalSpacer_3, 3, 2, 1, 1);
236 
237         horizontalSpacer_5 = new QSpacerItem(5, 20, QSizePolicy::Fixed,
    QSizePolicy::Minimum);
238 
239         gridLayout->addItem(horizontalSpacer_5, 4, 7, 1, 1);
240 
241         iButton3 = new ColorButton(MPerfWidget);
242         iButton3->setObjectName(QStringLiteral("iButton3"));
243         sizePolicy.setHeightForWidth(iButton3->sizePolicy().hasHeightForWidth());
244         iButton3->setSizePolicy(sizePolicy);
245         iButton3->setMinimumSize(QSize(0, 40));
246 
247         gridLayout->addWidget(iButton3, 8, 2, 1, 1);
248 
249         label_12 = new QLabel(MPerfWidget);
250         label_12->setObjectName(QStringLiteral("label_12"));
251         QSizePolicy sizePolicy1(QSizePolicy::Expanding, QSizePolicy::Preferred);
252         sizePolicy1.setHorizontalStretch(0);
253         sizePolicy1.setVerticalStretch(0);
254         sizePolicy1.setHeightForWidth(label_12->sizePolicy().hasHeightForWidth());
255         label_12->setSizePolicy(sizePolicy1);
256 
257         gridLayout->addWidget(label_12, 4, 5, 1, 1);
258 
259         xBox4 = new QSpinBox(MPerfWidget);
260         xBox4->setObjectName(QStringLiteral("xBox4"));
261         xBox4->setMinimum(100);
262         xBox4->setMaximum(12000);
263         xBox4->setSingleStep(50);
264 
265         gridLayout->addWidget(xBox4, 9, 6, 1, 1);
266 
267         xBox0 = new QSpinBox(MPerfWidget);
268         xBox0->setObjectName(QStringLiteral("xBox0"));
269         xBox0->setMinimum(100);
270         xBox0->setMaximum(12000);
271         xBox0->setSingleStep(50);
272 
273         gridLayout->addWidget(xBox0, 11, 6, 1, 1);
274 
275         yBox5 = new QSpinBox(MPerfWidget);
276         yBox5->setObjectName(QStringLiteral("yBox5"));
277         yBox5->setMinimum(100);
278         yBox5->setMaximum(12000);
279         yBox5->setSingleStep(50);
280 
281         gridLayout->addWidget(yBox5, 10, 9, 1, 1);
282 
283         eBox5 = new QCheckBox(MPerfWidget);
284         eBox5->setObjectName(QStringLiteral("eBox5"));
285 
286         gridLayout->addWidget(eBox5, 10, 1, 1, 1);
287 
288         xSlider5 = new QSlider(MPerfWidget);
289         xSlider5->setObjectName(QStringLiteral("xSlider5"));
290         xSlider5->setMinimum(0);
291         xSlider5->setMaximum(1000);
292         xSlider5->setSingleStep(10);
293         xSlider5->setPageStep(50);
294         xSlider5->setValue(0);
295         xSlider5->setOrientation(Qt::Horizontal);
296 
297         gridLayout->addWidget(xSlider5, 10, 5, 1, 1);
298 
299         xBox1 = new QSpinBox(MPerfWidget);
300         xBox1->setObjectName(QStringLiteral("xBox1"));
301         sizePolicy.setHeightForWidth(xBox1->sizePolicy().hasHeightForWidth());
302         xBox1->setSizePolicy(sizePolicy);
303         xBox1->setMinimum(100);
304         xBox1->setMaximum(12000);
```

```
305         xBox1->setSingleStep(50);
306
307         gridLayout->addWidget(xBox1, 6, 6, 1, 1);
308
309         yBox2 = new QSpinBox(MPerfWidget);
310         yBox2->setObjectName(QStringLiteral("yBox2"));
311         yBox2->setMinimum(100);
312         yBox2->setMaximum(12000);
313         yBox2->setSingleStep(50);
314
315         gridLayout->addWidget(yBox2, 7, 9, 1, 1);
316
317         yBox3 = new QSpinBox(MPerfWidget);
318         yBox3->setObjectName(QStringLiteral("yBox3"));
319         yBox3->setMinimum(100);
320         yBox3->setMaximum(12000);
321         yBox3->setSingleStep(50);
322
323         gridLayout->addWidget(yBox3, 8, 9, 1, 1);
324
325         xSlider4 = new QSlider(MPerfWidget);
326         xSlider4->setObjectName(QStringLiteral("xSlider4"));
327         xSlider4->setMinimum(0);
328         xSlider4->setMaximum(1000);
329         xSlider4->setSingleStep(10);
330         xSlider4->setPageStep(50);
331         xSlider4->setValue(0);
332         xSlider4->setOrientation(Qt::Horizontal);
333
334         gridLayout->addWidget(xSlider4, 9, 5, 1, 1);
335
336         label_4 = new QLabel(MPerfWidget);
337         label_4->setObjectName(QStringLiteral("label_4"));
338         sizePolicy1.setHeightForWidth(label_4->sizePolicy().hasHeightForWidth());
339         label_4->setSizePolicy(sizePolicy1);
340
341         gridLayout->addWidget(label_4, 4, 8, 1, 1);
342
343         xSlider0 = new QSlider(MPerfWidget);
344         xSlider0->setObjectName(QStringLiteral("xSlider0"));
345         xSlider0->setMinimum(0);
346         xSlider0->setMaximum(1000);
347         xSlider0->setSingleStep(10);
348         xSlider0->setPageStep(50);
349         xSlider0->setValue(0);
350         xSlider0->setOrientation(Qt::Horizontal);
351
352         gridLayout->addWidget(xSlider0, 11, 5, 1, 1);
353
354         iButton2 = new ColorButton(MPerfWidget);
355         iButton2->setObjectName(QStringLiteral("iButton2"));
356         sizePolicy.setHeightForWidth(iButton2->sizePolicy().hasHeightForWidth());
357         iButton2->setSizePolicy(sizePolicy);
358         iButton2->setMinimumSize(QSize(0, 40));
359
360         gridLayout->addWidget(iButton2, 7, 2, 1, 1);
361
362         line_3 = new QFrame(MPerfWidget);
363         line_3->setObjectName(QStringLiteral("line_3"));
364         line_3->setFrameShape(QFrame::HLine);
365         line_3->setFrameShadow(QFrame::Sunken);
366
367         gridLayout->addWidget(line_3, 5, 0, 1, 11);
368
369         iButtonO = new ColorButton(MPerfWidget);
370         iButtonO->setObjectName(QStringLiteral("iButtonO"));
371         sizePolicy.setHeightForWidth(iButtonO->sizePolicy().hasHeightForWidth());
372         iButtonO->setSizePolicy(sizePolicy);
373         iButtonO->setMinimumSize(QSize(0, 40));
374
375         gridLayout->addWidget(iButtonO, 12, 2, 1, 1);
376
377         yBox1 = new QSpinBox(MPerfWidget);
378         yBox1->setObjectName(QStringLiteral("yBox1"));
379         sizePolicy.setHeightForWidth(yBox1->sizePolicy().hasHeightForWidth());
380         yBox1->setSizePolicy(sizePolicy);
381         yBox1->setMinimum(100);
382         yBox1->setMaximum(12000);
383         yBox1->setSingleStep(50);
384
385         gridLayout->addWidget(yBox1, 6, 9, 1, 1);
386
387         ySlider2 = new QSlider(MPerfWidget);
388         ySlider2->setObjectName(QStringLiteral("ySlider2"));
389         ySlider2->setMinimum(0);
390         ySlider2->setMaximum(1000);
391         ySlider2->setSingleStep(10);
```

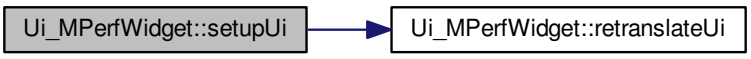
```
392         ySlider2->setPageStep(50);
393         ySlider2->setValue(0);
394         ySlider2->setOrientation(Qt::Horizontal);
395 
396         gridLayout->addWidget(ySlider2, 7, 8, 1, 1);
397 
398         xBox2 = new QSpinBox(MPerfWidget);
399         xBox2->setObjectName(QStringLiteral("xBox2"));
400         xBox2->setMinimum(100);
401         xBox2->setMaximum(12000);
402         xBox2->setSingleStep(50);
403 
404         gridLayout->addWidget(xBox2, 7, 6, 1, 1);
405 
406         xSlider1 = new QSlider(MPerfWidget);
407         xSlider1->setObjectName(QStringLiteral("xSlider1"));
408         QSizePolicy sizePolicy2(QSizePolicy::Expanding, QSizePolicy::Fixed);
409         sizePolicy2.setHorizontalStretch(0);
410         sizePolicy2.setVerticalStretch(0);
411         sizePolicy2.setHeightForWidth(xSlider1->sizePolicy().hasHeightForWidth());
412         xSlider1->setSizePolicy(sizePolicy2);
413         xSlider1->setMinimum(0);
414         xSlider1->setMaximum(1000);
415         xSlider1->setSingleStep(10);
416         xSlider1->setPageStep(50);
417         xSlider1->setValue(0);
418         xSlider1->setOrientation(Qt::Horizontal);
419 
420         gridLayout->addWidget(xSlider1, 6, 5, 1, 1);
421 
422         horizontalSpacer = new QSpacerItem(5, 20, QSizePolicy::Fixed, QSizePolicy::Minimum)
    ;
423 
424         gridLayout->addItem(horizontalSpacer, 0, 0, 1, 1);
425 
426         yBox4 = new QSpinBox(MPerfWidget);
427         yBox4->setObjectName(QStringLiteral("yBox4"));
428         yBox4->setMinimum(100);
429         yBox4->setMaximum(12000);
430         yBox4->setSingleStep(50);
431 
432         gridLayout->addWidget(yBox4, 9, 9, 1, 1);
433 
434         iButton1 = new ColorButton(MPerfWidget);
435         iButton1->setObjectName(QStringLiteral("iButton1"));
436         sizePolicy.setHeightForWidth(iButton1->sizePolicy().hasHeightForWidth());
437         iButton1->setSizePolicy(sizePolicy);
438         iButton1->setMinimumSize(QSize(0, 40));
439 
440         gridLayout->addWidget(iButton1, 6, 2, 1, 1);
441 
442         ySlider4 = new QSlider(MPerfWidget);
443         ySlider4->setObjectName(QStringLiteral("ySlider4"));
444         ySlider4->setMinimum(0);
445         ySlider4->setMaximum(1000);
446         ySlider4->setSingleStep(10);
447         ySlider4->setPageStep(50);
448         ySlider4->setValue(0);
449         ySlider4->setOrientation(Qt::Horizontal);
450 
451         gridLayout->addWidget(ySlider4, 9, 8, 1, 1);
452 
453         ySlider1 = new QSlider(MPerfWidget);
454         ySlider1->setObjectName(QStringLiteral("ySlider1"));
455         ySlider1->setMinimum(0);
456         ySlider1->setMaximum(1000);
457         ySlider1->setSingleStep(10);
458         ySlider1->setPageStep(50);
459         ySlider1->setValue(0);
460         ySlider1->setOrientation(Qt::Horizontal);
461 
462         gridLayout->addWidget(ySlider1, 6, 8, 1, 1);
463 
464         iButton0 = new ColorButton(MPerfWidget);
465         iButton0->setObjectName(QStringLiteral("iButton0"));
466         sizePolicy.setHeightForWidth(iButton0->sizePolicy().hasHeightForWidth());
467         iButton0->setSizePolicy(sizePolicy);
468         iButton0->setMinimumSize(QSize(0, 40));
469 
470         gridLayout->addWidget(iButton0, 11, 2, 1, 1);
471 
472         xSlider3 = new QSlider(MPerfWidget);
473         xSlider3->setObjectName(QStringLiteral("xSlider3"));
474         xSlider3->setMinimum(0);
475         xSlider3->setMaximum(1000);
476         xSlider3->setSingleStep(10);
477         xSlider3->setPageStep(50);
```

```
478         xSlider3->setValue(0);
479         xSlider3->setOrientation(Qt::Horizontal);
480 
481         gridLayout->addWidget(xSlider3, 8, 5, 1, 1);
482 
483         xBox5 = new QSpinBox(MPerfWidget);
484         xBox5->setObjectName(QStringLiteral("xBox5"));
485         xBox5->setMinimum(100);
486         xBox5->setMaximum(12000);
487         xBox5->setSingleStep(50);
488 
489         gridLayout->addWidget(xBox5, 10, 6, 1, 1);
490 
491         label_3 = new QLabel(MPerfWidget);
492         label_3->setObjectName(QStringLiteral("label_3"));
493 
494         gridLayout->addWidget(label_3, 12, 5, 1, 5);
495 
496         label_6 = new QLabel(MPerfWidget);
497         label_6->setObjectName(QStringLiteral("label_6"));
498 
499         gridLayout->addWidget(label_6, 7, 3, 1, 1);
500 
501         ySlider5 = new QSlider(MPerfWidget);
502         ySlider5->setObjectName(QStringLiteral("ySlider5"));
503         ySlider5->setMinimum(0);
504         ySlider5->setMaximum(1000);
505         ySlider5->setSingleStep(10);
506         ySlider5->setPageStep(50);
507         ySlider5->setValue(0);
508         ySlider5->setOrientation(Qt::Horizontal);
509 
510         gridLayout->addWidget(ySlider5, 10, 8, 1, 1);
511 
512         label_7 = new QLabel(MPerfWidget);
513         label_7->setObjectName(QStringLiteral("label_7"));
514 
515         gridLayout->addWidget(label_7, 8, 3, 1, 1);
516 
517         label_8 = new QLabel(MPerfWidget);
518         label_8->setObjectName(QStringLiteral("label_8"));
519 
520         gridLayout->addWidget(label_8, 9, 3, 1, 1);
521 
522         label_5 = new QLabel(MPerfWidget);
523         label_5->setObjectName(QStringLiteral("label_5"));
524 
525         gridLayout->addWidget(label_5, 6, 3, 1, 1);
526 
527         label_15 = new QLabel(MPerfWidget);
528         label_15->setObjectName(QStringLiteral("label_15"));
529 
530         gridLayout->addWidget(label_15, 12, 3, 1, 1);
531 
532         label_13 = new QLabel(MPerfWidget);
533         label_13->setObjectName(QStringLiteral("label_13"));
534 
535         gridLayout->addWidget(label_13, 10, 3, 1, 1);
536 
537         label_14 = new QLabel(MPerfWidget);
538         label_14->setObjectName(QStringLiteral("label_14"));
539 
540         gridLayout->addWidget(label_14, 11, 3, 1, 1);
541 
542         eBox4 = new QCheckBox(MPerfWidget);
543         eBox4->setObjectName(QStringLiteral("eBox4"));
544 
545         gridLayout->addWidget(eBox4, 9, 1, 1, 1);
546 
547         label_11 = new QLabel(MPerfWidget);
548         label_11->setObjectName(QStringLiteral("label_11"));
549 
550         gridLayout->addWidget(label_11, 4, 2, 1, 2);
551 
552         eBox3 = new QCheckBox(MPerfWidget);
553         eBox3->setObjectName(QStringLiteral("eBox3"));
554 
555         gridLayout->addWidget(eBox3, 8, 1, 1, 1);
556 
557         eBox1 = new QCheckBox(MPerfWidget);
558         eBox1->setObjectName(QStringLiteral("eBox1"));
559 
560         gridLayout->addWidget(eBox1, 6, 1, 1, 1);
561 
562         label_2 = new QLabel(MPerfWidget);
563         label_2->setObjectName(QStringLiteral("label_2"));
564         label_2->setFont(font);
```

```
565
566         gridLayout->addWidget(label_2, 14, 0, 1, 10);
567
568         verticalSpacer = new QSpacerItem(20, 30, QSizePolicy::Minimum, QSizePolicy::Fixed);
569
570         gridLayout->addItem(verticalSpacer, 13, 2, 1, 1);
571
572         aSnapBox = new QCheckBox(MPerfWidget);
573         aSnapBox->setObjectName(QStringLiteral("aSnapBox"));
574
575         gridLayout->addWidget(aSnapBox, 16, 1, 1, 4);
576
577         line_2 = new QFrame(MPerfWidget);
578         line_2->setObjectName(QStringLiteral("line_2"));
579         line_2->setFrameShape(QFrame::HLine);
580         line_2->setFrameShadow(QFrame::Sunken);
581
582         gridLayout->addWidget(line_2, 15, 0, 1, 10);
583
584         horizontalLayout_4 = new QHBoxLayout();
585         horizontalLayout_4->setObjectName(QStringLiteral("horizontalLayout_4"));
586         indicBox = new QCheckBox(MPerfWidget);
587         indicBox->setObjectName(QStringLiteral("indicBox"));
588
589         horizontalLayout_4->addWidget(indicBox);
590
591         horizontalSpacer_3 = new QSpacerItem(5, 20, QSizePolicy::Fixed,
    QSizePolicy::Minimum);
592
593         horizontalLayout_4->addItem(horizontalSpacer_3);
594
595         spinBox = new QSpinBox(MPerfWidget);
596         spinBox->setObjectName(QStringLiteral("spinBox"));
597         spinBox->setMaximum(100);
598         spinBox->setValue(100);
599
600         horizontalLayout_4->addWidget(spinBox);
601
602         horizontalSpacer_7 = new QSpacerItem(40, 20, QSizePolicy::Expanding,
    QSizePolicy::Minimum);
603
604         horizontalLayout_4->addItem(horizontalSpacer_7);
605
606
607         gridLayout->addLayout(horizontalLayout_4, 0, 1, 1, 9);
608
609         xyBox = new QCheckBox(MPerfWidget);
610         xyBox->setObjectName(QStringLiteral("xyBox"));
611
612         gridLayout->addWidget(xyBox, 2, 1, 1, 9);
613
614
615         verticalLayout->addLayout(gridLayout);
616
617         verticalSpacer_2 = new QSpacerItem(20, 40, QSizePolicy::Minimum,
    QSizePolicy::Expanding);
618
619         verticalLayout->addItem(verticalSpacer_2);
620
621         horizontalLayout = new QHBoxLayout();
622         horizontalLayout->setObjectName(QStringLiteral("horizontalLayout"));
623         copyButton = new QPushButton(MPerfWidget);
624         copyButton->setObjectName(QStringLiteral("copyButton"));
625
626         horizontalLayout->addWidget(copyButton);
627
628         horizontalSpacer_6 = new QSpacerItem(40, 20, QSizePolicy::Expanding,
    QSizePolicy::Minimum);
629
630         horizontalLayout->addItem(horizontalSpacer_6);
631
632
633         verticalLayout->addLayout(horizontalLayout);
634
635         QWidget::setTabOrder(indicBox, spinBox);
636         QWidget::setTabOrder(spinBox, xyBox);
637         QWidget::setTabOrder(xyBox, eBox1);
638         QWidget::setTabOrder(eBox1, eBox2);
639         QWidget::setTabOrder(eBox2, eBox3);
640         QWidget::setTabOrder(eBox3, eBox4);
641         QWidget::setTabOrder(eBox4, eBox5);
642         QWidget::setTabOrder(eBox5, iButton1);
643         QWidget::setTabOrder(iButton1, iButton2);
644         QWidget::setTabOrder(iButton2, iButton3);
645         QWidget::setTabOrder(iButton3, iButton4);
646         QWidget::setTabOrder(iButton4, iButton5);
647         QWidget::setTabOrder(iButton5, iButton0);
```

```
648         QWidget::setTabOrder(iButton0, iButtonO);
649         QWidget::setTabOrder(iButtonO, xSlider1);
650         QWidget::setTabOrder(xSlider1, xSlider2);
651         QWidget::setTabOrder(xSlider2, xSlider3);
652         QWidget::setTabOrder(xSlider3, xSlider4);
653         QWidget::setTabOrder(xSlider4, xSlider5);
654         QWidget::setTabOrder(xSlider5, xSlider0);
655         QWidget::setTabOrder(xSlider0, xBox1);
656         QWidget::setTabOrder(xBox1, xBox2);
657         QWidget::setTabOrder(xBox2, xBox3);
658         QWidget::setTabOrder(xBox3, xBox4);
659         QWidget::setTabOrder(xBox4, xBox5);
660         QWidget::setTabOrder(xBox5, xBox0);
661         QWidget::setTabOrder(xBox0, ySlider1);
662         QWidget::setTabOrder(ySlider1, ySlider2);
663         QWidget::setTabOrder(ySlider2, ySlider3);
664         QWidget::setTabOrder(ySlider3, ySlider4);
665         QWidget::setTabOrder(ySlider4, ySlider5);
666         QWidget::setTabOrder(ySlider5, ySlider0);
667         QWidget::setTabOrder(ySlider0, yBox1);
668         QWidget::setTabOrder(yBox1, yBox2);
669         QWidget::setTabOrder(yBox2, yBox3);
670         QWidget::setTabOrder(yBox3, yBox4);
671         QWidget::setTabOrder(yBox4, yBox5);
672         QWidget::setTabOrder(yBox5, yBox0);
673         QWidget::setTabOrder(yBox0, aSnapBox);
674         QWidget::setTabOrder(aSnapBox, lHeightBox);
675         QWidget::setTabOrder(lHeightBox, copyButton);
676 
677         retranslateUi(MPerfWidget);
678 
679         QMetaObject::connectSlotsByName(MPerfWidget);
680     } // setupUi
```

Here is the call graph for this function:



Here is the caller graph for this function:

12.99.3 Field Documentation

12.99.3.1 QCheckBox* Ui_MPerfWidget::aSnapBox

Definition at line 98 of file ui_mperfwidget.h.

Referenced by retranslateUi(), MPerfWidget::setPerf(), and setupUi().

12.99.3.2 QPushButton* Ui_MPerfWidget::copyButton

Definition at line 108 of file ui_mperfwidget.h.

Referenced by retranslateUi(), and setupUi().

12.99.3.3 QCheckBox* Ui_MPerfWidget::eBox1

Definition at line 95 of file ui_mperfwidget.h.

Referenced by MPerfWidget::MPerfWidget(), retranslateUi(), and setupUi().

12.99.3.4 QCheckBox* Ui_MPerfWidget::eBox2

Definition at line 44 of file ui_mperfwidget.h.

Referenced by MPerfWidget::MPerfWidget(), retranslateUi(), and setupUi().

12.99.3.5 QCheckBox* Ui_MPerfWidget::eBox3

Definition at line 94 of file ui_mperfwidget.h.

Referenced by MPerfWidget::MPerfWidget(), retranslateUi(), and setupUi().

12.99.3.6 QCheckBox* Ui_MPerfWidget::eBox4

Definition at line 92 of file ui_mperfwidget.h.

Referenced by MPerfWidget::MPerfWidget(), retranslateUi(), and setupUi().

12.99.3.7 QCheckBox* Ui_MPerfWidget::eBox5

Definition at line 60 of file ui_mperfwidget.h.

Referenced by MPerfWidget::MPerfWidget(), retranslateUi(), and setupUi().

12.99.3.8 QGridLayout* Ui_MPerfWidget::gridLayout

Definition at line 39 of file ui_mperfwidget.h.

Referenced by setupUi().

12.99.3.9 QHBoxLayout* Ui_MPerfWidget::horizontalLayout

Definition at line 107 of file ui_mperfwidget.h.

Referenced by setupUi().

12.99.3.10 QHBoxLayout* Ui_MPerfWidget::horizontalLayout_2

Definition at line 41 of file ui_mperfwidget.h.

Referenced by setupUi().

12.99.3.11 QHBoxLayout* Ui_MPerfWidget::horizontalLayout_4

Definition at line 100 of file ui_mperfwidget.h.

Referenced by setupUi().

12.99.3.12 QSpacerItem* Ui_MPerfWidget::horizontalSpacer

Definition at line 75 of file ui_mperfwidget.h.

Referenced by setupUi().

12.99.3.13 QSpacerItem* Ui_MPerfWidget::horizontalSpacer_2

Definition at line 43 of file ui_mperfwidget.h.

Referenced by setupUi().

12.99.3.14 QSpacerItem* Ui_MPerfWidget::horizontalSpacer_3

Definition at line 102 of file ui_mperfwidget.h.

Referenced by setupUi().

12.99.3.15 QSpacerItem* Ui_MPerfWidget::horizontalSpacer_4

Definition at line 52 of file ui_mperfwidget.h.

Referenced by setupUi().

12.99.3.16 QSpacerItem* Ui_MPerfWidget::horizontalSpacer_5

Definition at line 54 of file ui_mperfwidget.h.

Referenced by setupUi().

12.99.3.17 QSpacerItem* Ui_MPerfWidget::horizontalSpacer_6

Definition at line 109 of file ui_mperfwidget.h.

Referenced by setupUi().

12.99.3.18 QSpacerItem* Ui_MPerfWidget::horizontalSpacer_7

Definition at line 104 of file ui_mperfwidget.h.

Referenced by setupUi().

12.99.3.19 ColorButton* Ui_MPerfWidget::iButton0

Definition at line 80 of file ui_mperfwidget.h.

Referenced by MPerfWidget::MPerfWidget(), retranslateUi(), and setupUi().

12.99.3.20 ColorButton* Ui_MPerfWidget::iButton1

Definition at line 77 of file ui_mperfwidget.h.

Referenced by MPerfWidget::MPerfWidget(), retranslateUi(), and setupUi().

12.99.3.21 ColorButton* Ui_MPerfWidget::iButton2

Definition at line 68 of file ui_mperfwidget.h.

Referenced by MPerfWidget::MPerfWidget(), retranslateUi(), and setupUi().

12.99.3.22 ColorButton* Ui_MPerfWidget::iButton3

Definition at line 55 of file ui_mperfwidget.h.

Referenced by MPerfWidget::MPerfWidget(), retranslateUi(), and setupUi().

12.99.3.23 ColorButton* Ui_MPerfWidget::iButton4

Definition at line 50 of file ui_mperfwidget.h.

Referenced by MPerfWidget::MPerfWidget(), retranslateUi(), and setupUi().

12.99.3.24 ColorButton* Ui_MPerfWidget::iButton5

Definition at line 51 of file ui_mperfwidget.h.

Referenced by MPerfWidget::MPerfWidget(), retranslateUi(), and setupUi().

12.99.3.25 ColorButton* Ui_MPerfWidget::iButtonO

Definition at line 70 of file ui_mperfwidget.h.

Referenced by MPerfWidget::colorChanged(), MPerfWidget::MPerfWidget(), retranslateUi(), MPerfWidget::setPerf(), and setupUi().

12.99.3.26 QCheckBox* Ui_MPerfWidget::indicBox

Definition at line 101 of file ui_mperfwidget.h.

Referenced by retranslateUi(), MPerfWidget::setPerf(), and setupUi().

12.99.3.27 QLabel* Ui_MPerfWidget::label

Definition at line 37 of file ui_mperfwidget.h.

Referenced by retranslateUi(), and setupUi().

12.99.3.28 QLabel* Ui_MPerfWidget::label_11

Definition at line 93 of file ui_mperfwidget.h.

Referenced by retranslateUi(), and setupUi().

12.99.3.29 QLabel* Ui_MPerfWidget::label_12

Definition at line 56 of file ui_mperfwidget.h.

Referenced by retranslateUi(), and setupUi().

12.99.3.30 QLabel* Ui_MPerfWidget::label_13

Definition at line 90 of file ui_mperfwidget.h.

Referenced by retranslateUi(), and setupUi().

12.99.3.31 QLabel* Ui_MPerfWidget::label_14

Definition at line 91 of file ui_mperfwidget.h.

Referenced by retranslateUi(), and setupUi().

12.99.3.32 QLabel* Ui_MPerfWidget::label_15

Definition at line 89 of file ui_mperfwidget.h.

Referenced by retranslateUi(), and setupUi().

12.99.3.33 QLabel* Ui_MPerfWidget::label_2

Definition at line 96 of file ui_mperfwidget.h.

Referenced by retranslateUi(), and setupUi().

12.99.3.34 QLabel* Ui_MPerfWidget::label_25

Definition at line 40 of file ui_mperfwidget.h.

Referenced by retranslateUi(), and setupUi().

12.99.3.35 QLabel* Ui_MPerfWidget::label_3

Definition at line 83 of file ui_mperfwidget.h.

Referenced by retranslateUi(), and setupUi().

12.99.3.36 QLabel* Ui_MPerfWidget::label_4

Definition at line 66 of file ui_mperfwidget.h.

Referenced by retranslateUi(), and setupUi().

12.99.3.37 QLabel* Ui_MPerfWidget::label_5

Definition at line 88 of file ui_mperfwidget.h.

Referenced by retranslateUi(), and setupUi().

12.99.3.38 QLabel* Ui_MPerfWidget::label_6

Definition at line 84 of file ui_mperfwidget.h.

Referenced by retranslateUi(), and setupUi().

12.99.3.39 QLabel* Ui_MPerfWidget::label_7

Definition at line 86 of file ui_mperfwidget.h.

Referenced by retranslateUi(), and setupUi().

12.99.3.40 QLabel* Ui_MPerfWidget::label_8

Definition at line 87 of file ui_mperfwidget.h.

Referenced by retranslateUi(), and setupUi().

12.99.3.41 QComboBox* Ui_MPerfWidget::lHeightBox

Definition at line 42 of file ui_mperfwidget.h.

Referenced by retranslateUi(), MPerfWidget::setPerf(), and setupUi().

12.99.3.42 QFrame* Ui_MPerfWidget::line

Definition at line 38 of file ui_mperfwidget.h.

Referenced by setupUi().

12.99.3.43 QFrame* Ui_MPerfWidget::line_2

Definition at line 99 of file ui_mperfwidget.h.

Referenced by setupUi().

12.99.3.44 QFrame* Ui_MPerfWidget::line_3

Definition at line 69 of file ui_mperfwidget.h.

Referenced by setupUi().

12.99.3.45 QSpinBox* Ui_MPerfWidget::spinBox

Definition at line 103 of file ui_mperfwidget.h.

Referenced by retranslateUi(), MPerfWidget::setPerf(), and setupUi().

12.99.3.46 QVBoxLayout* Ui_MPerfWidget::verticalLayout

Definition at line 36 of file ui_mperfwidget.h.

Referenced by setupUi().

12.99.3.47 QSpacerItem* Ui_MPerfWidget::verticalSpacer

Definition at line 97 of file ui_mperfwidget.h.

Referenced by setupUi().

12.99.3.48 QSpacerItem* Ui_MPerfWidget::verticalSpacer_2

Definition at line 106 of file ui_mperfwidget.h.

Referenced by setupUi().

12.99.3.49 QSpacerItem* Ui_MPerfWidget::verticalSpacer_3

Definition at line 53 of file ui_mperfwidget.h.

Referenced by setupUi().

12.99.3.50 QSpinBox* Ui_MPerfWidget::xBox0

Definition at line 58 of file ui_mperfwidget.h.

Referenced by MPerfWidget::MPerfWidget(), and setupUi().

12.99.3.51 QSpinBox* Ui_MPerfWidget::xBox1

Definition at line 62 of file ui_mperfwidget.h.

Referenced by MPerfWidget::MPerfWidget(), and setupUi().

12.99.3.52 QSpinBox* Ui_MPerfWidget::xBox2

Definition at line 73 of file ui_mperfwidget.h.

Referenced by MPerfWidget::MPerfWidget(), and setupUi().

12.99.3.53 QSpinBox* Ui_MPerfWidget::xBox3

Definition at line 47 of file ui_mperfwidget.h.

Referenced by MPerfWidget::MPerfWidget(), and setupUi().

12.99.3.54 QSpinBox* Ui_MPerfWidget::xBox4

Definition at line 57 of file ui_mperfwidget.h.

Referenced by MPerfWidget::MPerfWidget(), and setupUi().

12.99.3.55 QSpinBox* Ui_MPerfWidget::xBox5

Definition at line 82 of file ui_mperfwidget.h.

Referenced by MPerfWidget::MPerfWidget(), and setupUi().

12.99.3.56 QSlider* Ui_MPerfWidget::xSlider0

Definition at line 67 of file ui_mperfwidget.h.

Referenced by MPerfWidget::MPerfWidget(), and setupUi().

12.99.3.57 QSlider* Ui_MPerfWidget::xSlider1

Definition at line 74 of file ui_mperfwidget.h.

Referenced by MPerfWidget::MPerfWidget(), and setupUi().

12.99.3.58 QSlider* Ui_MPerfWidget::xSlider2

Definition at line 46 of file ui_mperfwidget.h.

Referenced by MPerfWidget::MPerfWidget(), and setupUi().

12.99.3.59 QSlider* Ui_MPerfWidget::xSlider3

Definition at line 81 of file ui_mperfwidget.h.

Referenced by MPerfWidget::MPerfWidget(), and setupUi().

12.99.3.60 QSlider* Ui_MPerfWidget::xSlider4

Definition at line 65 of file ui_mperfwidget.h.

Referenced by MPerfWidget::MPerfWidget(), and setupUi().

12.99.3.61 QSlider* Ui_MPerfWidget::xSlider5

Definition at line 61 of file ui_mperfwidget.h.

Referenced by MPerfWidget::MPerfWidget(), and setupUi().

12.99.3.62 QCheckBox* Ui_MPerfWidget::xyBox

Definition at line 105 of file ui_mperfwidget.h.

Referenced by MPerfWidget::MPerfWidget(), retranslateUi(), setupUi(), and MPerfWidget::sliderYMoved().

12.99.3.63 QSpinBox* Ui_MPerfWidget::yBox0

Definition at line 48 of file ui_mperfwidget.h.

Referenced by MPerfWidget::MPerfWidget(), and setupUi().

12.99.3.64 QSpinBox* Ui_MPerfWidget::yBox1

Definition at line 71 of file ui_mperfwidget.h.

Referenced by MPerfWidget::MPerfWidget(), and setupUi().

12.99.3.65 QSpinBox* Ui_MPerfWidget::yBox2

Definition at line 63 of file ui_mperfwidget.h.

Referenced by MPerfWidget::MPerfWidget(), and setupUi().

12.99.3.66 QSpinBox* Ui_MPerfWidget::yBox3

Definition at line 64 of file ui_mperfwidget.h.

Referenced by MPerfWidget::MPerfWidget(), and setupUi().

12.99.3.67 QSpinBox* Ui_MPerfWidget::yBox4

Definition at line 76 of file ui_mperfwidget.h.

Referenced by MPerfWidget::MPerfWidget(), and setupUi().

12.99.3.68 QSpinBox* Ui_MPerfWidget::yBox5

Definition at line 59 of file ui_mperfwidget.h.

Referenced by MPerfWidget::MPerfWidget(), and setupUi().

12.99.3.69 QSlider* Ui_MPerfWidget::ySlider0

Definition at line 49 of file ui_mperfwidget.h.

Referenced by MPerfWidget::MPerfWidget(), and setupUi().

12.99.3.70 QSlider* Ui_MPerfWidget::ySlider1

Definition at line 79 of file ui_mperfwidget.h.

Referenced by MPerfWidget::MPerfWidget(), and setupUi().

12.99.3.71 QSlider* Ui_MPerfWidget::ySlider2

Definition at line 72 of file ui_mperfwidget.h.

Referenced by MPerfWidget::MPerfWidget(), and setupUi().

12.99.3.72 QSlider* Ui_MPerfWidget::ySlider3

Definition at line 45 of file ui_mperfwidget.h.

Referenced by MPerfWidget::MPerfWidget(), and setupUi().

12.99.3.73 QSlider* Ui_MPerfWidget::ySlider4

Definition at line 78 of file ui_mperfwidget.h.

Referenced by MPerfWidget::MPerfWidget(), and setupUi().

12.99.3.74 QSlider* Ui_MPerfWidget::ySlider5

Definition at line 85 of file ui_mperfwidget.h.

Referenced by MPerfWidget::MPerfWidget(), and setupUi().

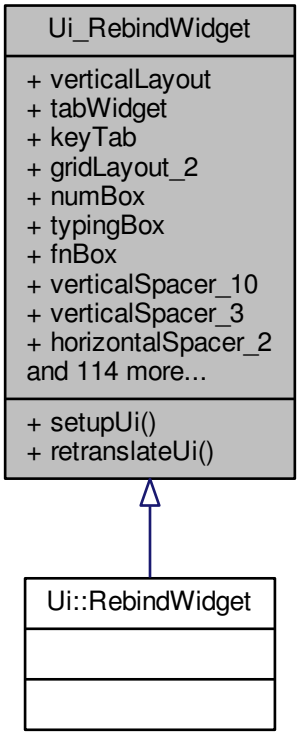
The documentation for this class was generated from the following file:

- src/ckb/ui_mperfwidget.h

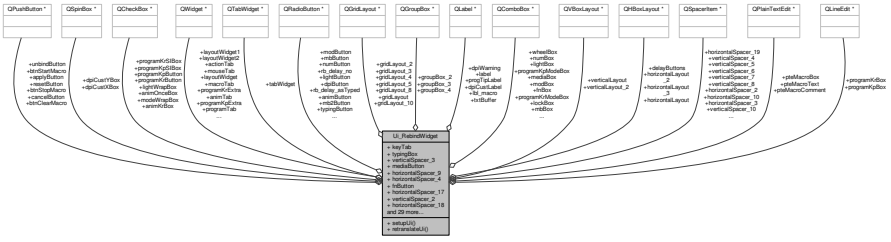
12.100 Ui_RebindWidget Class Reference

`#include <src/ckb/ui_rebindwidget.h>`

Inheritance diagram for Ui_RebindWidget:



Collaboration diagram for Ui_RebindWidget:



Public Member Functions

- void setupUi (QWidget ∗RebindWidget)
- void retranslateUi (QWidget ∗RebindWidget)

Data Fields

- QVBoxLayout * verticalLayout
- QTabWidget * tabWidget
- QWidget * keyTab
- QGridLayout * gridLayout_2
- QComboBox * numBox
- QComboBox * typingBox
- QComboBox * fnBox
- QSpacerItem * verticalSpacer_10
- QSpacerItem * verticalSpacer_3
- QSpacerItem * horizontalSpacer_2
- QRadioButton * typingButton
- QComboBox * modBox
- QSpacerItem * horizontalSpacer_3
- QRadioButton * mediaButton
- QSpacerItem * verticalSpacer_4
- QSpacerItem * horizontalSpacer_9
- QSpacerItem * horizontalSpacer_4
- QRadioButton * modButton
- QRadioButton * fnButton
- QComboBox * mediaBox
- QRadioButton * numButton
- QSpacerItem * horizontalSpacer_17
- QWidget * mouseTab
- QGridLayout * gridLayout
- QRadioButton * dpiButton
- QSpacerItem * verticalSpacer_2
- QSpacerItem * verticalSpacer_8
- QSpacerItem * horizontalSpacer_18
- QSpacerItem * horizontalSpacer_20
- QComboBox * mb2Box
- QComboBox * dpiBox
- QSpinBox * dpiCustYBox
- QComboBox * mbBox
- QComboBox * wheelBox
- QRadioButton * mbButton
- QSpinBox * dpiCustXBox
- QLabel * dpiWarning
- QRadioButton * wheelButton
- QSpacerItem * horizontalSpacer_22
- QRadioButton * mb2Button
- QLabel * dpiCustLabel
- QSpacerItem * horizontalSpacer_19
- QSpacerItem * horizontalSpacer_21
- QWidget * animTab
- QGridLayout * gridLayout_5
- QSpacerItem * verticalSpacer_11
- QComboBox * animBox
- QSpacerItem * horizontalSpacer_24
- QSpacerItem * horizontalSpacer_25
- QCheckBox * animKrBox
- QCheckBox * animOnceBox
- QSpacerItem * horizontalSpacer_26
- QRadioButton * animButton

- QSpacerItem * horizontalSpacer_23
- QWidget * actionTab
- QGridLayout * gridLayout_3
- QSpacerItem * horizontalSpacer_5
- QRadioButton * modeButton
- QCheckBox * modeWrapBox
- QCheckBox * lightWrapBox
- QSpacerItem * verticalSpacer_7
- QSpacerItem * verticalSpacer_6
- QRadioButton * lockButton
- QSpacerItem * verticalSpacer_5
- QComboBox * lightBox
- QComboBox * lockBox
- QRadioButton * lightButton
- QComboBox * modeBox
- QSpacerItem * horizontalSpacer_8
- QSpacerItem * horizontalSpacer_10
- QSpacerItem * horizontalSpacer_7
- QSpacerItem * horizontalSpacer_6
- QWidget * programTab
- QGridLayout * gridLayout_4
- QSpacerItem * verticalSpacer
- QSpacerItem * horizontalSpacer
- QCheckBox * programKpButton
- QLineEdit * programKpBox
- QSpacerItem * horizontalSpacer_11
- QSpacerItem * horizontalSpacer_13
- QLineEdit * programKrBox
- QCheckBox * programKrButton
- QSpacerItem * horizontalSpacer_12
- QWidget * programKpExtra
- QHBoxLayout * horizontalLayout_2
- QCheckBox * programKpSIBox
- QComboBox * programKpModeBox
- QSpacerItem * horizontalSpacer_15
- QWidget * programKrExtra
- QHBoxLayout * horizontalLayout_3
- QCheckBox * programKrSIBox
- QComboBox * programKrModeBox
- QSpacerItem * horizontalSpacer_16
- QLabel * progTipLabel
- QWidget * macroTab
- QWidget * layoutWidget1
- QGridLayout * gridLayout_8
- QGroupBox * groupBox_2
- QPlainTextEdit * pteMacroBox
- QPlainTextEdit * pteMacroComment
- QPlainTextEdit * pteMacroText
- QWidget * layoutWidget
- QVBoxLayout * verticalLayout_2
- QGroupBox * groupBox_3
- QGroupBox * groupBox_4
- QWidget * layoutWidget_2
- QGridLayout * gridLayout_10
- QPushButton * btnStopMacro

- QPushButton * btnClearMacro
- QPushButton * btnStartMacro
- QLabel * label
- QLabel * lbl_macro
- QWidget * layoutWidget2
- QHBoxLayout * delayButtons
- QRadioButton * rb_delay_no
- QRadioButton * rb_delay_default
- QRadioButton * rb_delay_asTyped
- QLabel * txtBuffer
- QHBoxLayout * horizontalLayout
- QSpacerItem * horizontalSpacer_14
- QPushButton * unbindButton
- QPushButton * resetButton
- QPushButton * cancelButton
- QPushButton * applyButton

12.100.1 Detailed Description

Definition at line 35 of file ui_rebindwidget.h.

12.100.2 Member Function Documentation

12.100.2.1 void Ui_RebindWidget::retranslateUi (QWidget * *RebindWidget*) [inline]

Definition at line 901 of file ui_rebindwidget.h.

References actionTab, animBox, animButton, animKrBox, animOnceBox, animTab, applyButton, btnClearMacro, btnStartMacro, btnStopMacro, cancelButton, dpiBox, dpiButton, dpiCustLabel, dpiWarning, fnBox, fnButton, groupBox_2, groupBox_3, groupBox_4, keyTab, label, lbl_macro, lightBox, lightButton, lightWrapBox, lockBox, lockButton, macroTab, mb2Box, mb2Button, mbBox, mbButton, mediaBox, mediaButton, modBox, modButton, modeBox, modeButton, modeWrapBox, mouseTab, numBox, numButton, programKpButton, programKpModeBox, programKpSlBox, programKrButton, programKrModeBox, programKrSlBox, programTab, progTipLabel, pteMacroComment, pteMacroText, rb_delay_asTyped, rb_delay_default, rb_delay_no, resetButton, tabWidget, txtBuffer, typingBox, typingButton, unbindButton, wheelBox, and wheelButton.

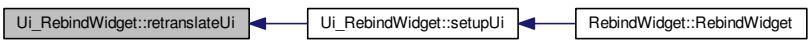
Referenced by setupUi().

```
902     {
903         numBox->clear();
904         numBox->insertItems(0, QStringList()
905          << QString()
906         );
907         typingBox->clear();
908         typingBox->insertItems(0, QStringList()
909          << QString()
910         );
911         fnBox->clear();
912         fnBox->insertItems(0, QStringList()
913          << QString()
914         );
915         typingButton->setText(QApplication::translate("RebindWidget", "Typing:", Q_NULLPTR));
916         modBox->clear();
917         modBox->insertItems(0, QStringList()
918          << QString()
919         );
920         mediaButton->setText(QApplication::translate("RebindWidget", "Media:", Q_NULLPTR));
921         modButton->setText(QApplication::translate("RebindWidget", "Modifier:", Q_NULLPTR));
922         fnButton->setText(QApplication::translate("RebindWidget", "Miscellaneous:", Q_NULLPTR));
923         mediaBox->clear();
924         mediaBox->insertItems(0, QStringList()
925          << QString()
926         );
927         numButton->setText(QApplication::translate("RebindWidget", "Number pad:", Q_NULLPTR));
928         tabWidget->setTabText(tabWidget->indexOf(keyTab), QApplication::translate("
```

```
      RebindWidget", "Keyboard", Q_NULLPTR));
929         dpiButton->setText(QApplication::translate("RebindWidget", "Change DPI:", Q_NULLPTR));
930         mb2Box->clear();
931         mb2Box->insertItems(0, QStringList()
932          << QString()
933          << QApplication::translate("RebindWidget", "4 (Back)", Q_NULLPTR)
934          << QApplication::translate("RebindWidget", "5 (Forward)", Q_NULLPTR)
935          << QApplication::translate("RebindWidget", "6", Q_NULLPTR)
936          << QApplication::translate("RebindWidget", "7", Q_NULLPTR)
937          << QApplication::translate("RebindWidget", "8", Q_NULLPTR)
938         );
939         dpiBox->clear();
940         dpiBox->insertItems(0, QStringList()
941          << QString()
942          << QApplication::translate("RebindWidget", "(Cycle Up)", Q_NULLPTR)
943          << QApplication::translate("RebindWidget", "(Cycle Down)", Q_NULLPTR)
944          << QApplication::translate("RebindWidget", "(Up)", Q_NULLPTR)
945          << QApplication::translate("RebindWidget", "(Down)", Q_NULLPTR)
946          << QApplication::translate("RebindWidget", "Sniper", Q_NULLPTR)
947          << QApplication::translate("RebindWidget", "1:", Q_NULLPTR)
948          << QApplication::translate("RebindWidget", "2:", Q_NULLPTR)
949          << QApplication::translate("RebindWidget", "3:", Q_NULLPTR)
950          << QApplication::translate("RebindWidget", "4:", Q_NULLPTR)
951          << QApplication::translate("RebindWidget", "5:", Q_NULLPTR)
952          << QApplication::translate("RebindWidget", "Custom:", Q_NULLPTR)
953         );
954         mbBox->clear();
955         mbBox->insertItems(0, QStringList()
956          << QString()
957          << QApplication::translate("RebindWidget", "Left", Q_NULLPTR)
958          << QApplication::translate("RebindWidget", "Right", Q_NULLPTR)
959          << QApplication::translate("RebindWidget", "Middle", Q_NULLPTR)
960         );
961         wheelBox->clear();
962         wheelBox->insertItems(0, QStringList()
963          << QString()
964          << QApplication::translate("RebindWidget", "Up", Q_NULLPTR)
965          << QApplication::translate("RebindWidget", "Down", Q_NULLPTR)
966         );
967         mbButton->setText(QApplication::translate("RebindWidget", "Standard button:", Q_NULLPTR));
968         dpiWarning->setText(QApplication::translate("RebindWidget", "DPI is only available for
       mice.", Q_NULLPTR));
969         wheelButton->setText(QApplication::translate("RebindWidget", "Wheel:", Q_NULLPTR));
970         mb2Button->setText(QApplication::translate("RebindWidget", "Special button:", Q_NULLPTR));
971         dpiCustLabel->setText(QApplication::translate("RebindWidget", "x", Q_NULLPTR));
972         tabWidget->setTabText(tabWidget->indexOf(mouseTab),
      QApplication::translate("RebindWidget", "Mouse", Q_NULLPTR));
973         animBox->clear();
974         animBox->insertItems(0, QStringList()
975          << QString()
976         );
977         animKrBox->setText(QApplication::translate("RebindWidget", "Stop on key release",
      Q_NULLPTR));
978         animOnceBox->setText(QApplication::translate("RebindWidget", "Restart every time",
      Q_NULLPTR));
979         animButton->setText(QApplication::translate("RebindWidget", "Start animation on keypress:
      ", Q_NULLPTR));
980         tabWidget->setTabText(tabWidget->indexOf(animTab), QApplication::translate
      ("RebindWidget", "Animation", Q_NULLPTR));
981         modeButton->setText(QApplication::translate("RebindWidget", "Switch to mode:", Q_NULLPTR)
      );
982         modeWrapBox->setText(QApplication::translate("RebindWidget", "Wrap-around", Q_NULLPTR));
983         lightWrapBox->setText(QApplication::translate("RebindWidget", "Wrap-around", Q_NULLPTR)
      );
984         lockButton->setText(QApplication::translate("RebindWidget", "Windows lock:", Q_NULLPTR));
985         lightBox->clear();
986         lightBox->insertItems(0, QStringList()
987          << QString()
988          << QApplication::translate("RebindWidget", "Up", Q_NULLPTR)
989          << QApplication::translate("RebindWidget", "Down", Q_NULLPTR)
990         );
991         lockBox->clear();
992         lockBox->insertItems(0, QStringList()
993          << QString()
994          << QApplication::translate("RebindWidget", "Toggle", Q_NULLPTR)
995          << QApplication::translate("RebindWidget", "On", Q_NULLPTR)
996          << QApplication::translate("RebindWidget", "Off", Q_NULLPTR)
997         );
998         lightButton->setText(QApplication::translate("RebindWidget", "Brightness:", Q_NULLPTR));
999         modeBox->clear();
1000         modeBox->insertItems(0, QStringList()
1001          << QString()
1002          << QApplication::translate("RebindWidget", "(Previous)", Q_NULLPTR)
1003          << QApplication::translate("RebindWidget", "(Next)", Q_NULLPTR)
1004         );
1005         tabWidget->setTabText(tabWidget->indexOf(actionTab),
      QApplication::translate("RebindWidget", "Special", Q_NULLPTR));
```

```
1006         programKpButton->setText(QApplication::translate("RebindWidget", "Launch program on
      key press:", Q_NULLPTR));
1007         programKrButton->setText(QApplication::translate("RebindWidget", "Launch program on
      key release:", Q_NULLPTR));
1008         programKpSIBox->setText(QApplication::translate("RebindWidget", "Single instance",
     Q_NULLPTR));
1009         programKpModeBox->clear();
1010         programKpModeBox->insertItems(0, QStringList()
1011          << QApplication::translate("RebindWidget", "Run indefinitely", Q_NULLPTR)
1012          << QApplication::translate("RebindWidget", "Stop on release", Q_NULLPTR)
1013          << QApplication::translate("RebindWidget", "Stop on re-press", Q_NULLPTR)
1014         );
1015         programKrSIBox->setText(QApplication::translate("RebindWidget", "Single instance",
     Q_NULLPTR));
1016         programKrModeBox->clear();
1017         programKrModeBox->insertItems(0, QStringList()
1018          << QApplication::translate("RebindWidget", "Run indefinitely", Q_NULLPTR)
1019          << QApplication::translate("RebindWidget", "Stop on re-press", Q_NULLPTR)
1020         );
1021         progTipLabel->setText(QApplication::translate("RebindWidget", "Tip", Q_NULLPTR));
1022         tabWidget->setTabText(tabWidget->indexOf(programTab),
     QApplication::translate("RebindWidget", "Program", Q_NULLPTR));
1023         groupBox_2->setTitle(QApplication::translate("RebindWidget", "Macro Key Actions",
     Q_NULLPTR));
1024 #ifndef QT_NO_TOOLTIP
1025         pteMacroComment->setToolTip(QApplication::translate("RebindWidget", "What is this
      macro for?", Q_NULLPTR));
1026 #endif // QT_NO_TOOLTIP
1027         pteMacroComment->setDocumentTitle(QString());
1028         pteMacroComment->setPlainText(QString());
1029         pteMacroText->setDocumentTitle(QString());
1030         pteMacroText->setPlainText(QString());
1031         groupBox_3->setTitle(QApplication::translate("RebindWidget", "Macro Comment", Q_NULLPTR))
     ;
1032         groupBox_4->setTitle(QApplication::translate("RebindWidget", "Macro Text", Q_NULLPTR));
1033         btnStopMacro->setText(QApplication::translate("RebindWidget", "Stop", Q_NULLPTR));
1034         btnClearMacro->setText(QApplication::translate("RebindWidget", "Clear", Q_NULLPTR));
1035         btnStartMacro->setText(QApplication::translate("RebindWidget", "Start", Q_NULLPTR));
1036         label->setText(QApplication::translate("RebindWidget", "Record macro ", Q_NULLPTR));
1037         lbl_macro->setText(QApplication::translate("RebindWidget", "Comment label for help",
     Q_NULLPTR));
1038 #ifndef QT_NO_WHATSTHIS
1039         rb_delay_no->setWhatsThis(QApplication::translate("RebindWidget", "configure delays
      between keystrokes: Disable delay", Q_NULLPTR));
1040 #endif // QT_NO_WHATSTHIS
1041         rb_delay_no->setText(QApplication::translate("RebindWidget", "No delay", Q_NULLPTR));
1042 #ifndef QT_NO_WHATSTHIS
1043         rb_delay_default->setWhatsThis(QApplication::translate("RebindWidget", "configure
      delays between keystrokes: Set delay to default values: 20us up to 15 chars, 200us above", Q_NULLPTR));
1044 #endif // QT_NO_WHATSTHIS
1045         rb_delay_default->setText(QApplication::translate("RebindWidget", "default",
     Q_NULLPTR));
1046 #ifndef QT_NO_WHATSTHIS
1047         rb_delay_asTyped->setWhatsThis(QApplication::translate("RebindWidget", "configure
      delays between keystrokes: Remember delays as they had been typed while the macro definition process",
     Q_NULLPTR));
1048 #endif // QT_NO_WHATSTHIS
1049         rb_delay_asTyped->setText(QApplication::translate("RebindWidget", "as typed",
     Q_NULLPTR));
1050         txtBuffer->setText(QString());
1051         tabWidget->setTabText(tabWidget->indexOf(macroTab),
     QApplication::translate("RebindWidget", "Macro", Q_NULLPTR));
1052         unbindButton->setText(QApplication::translate("RebindWidget", "Unbind", Q_NULLPTR));
1053         resetButton->setText(QApplication::translate("RebindWidget", "Reset to Default",
     Q_NULLPTR));
1054         cancelButton->setText(QApplication::translate("RebindWidget", "Cancel", Q_NULLPTR));
1055         applyButton->setText(QApplication::translate("RebindWidget", "Apply", Q_NULLPTR));
1056         Q_UNUSED(RebindWidget);
1057     } // retranslateUi
```

Here is the caller graph for this function:



12.100.2.2 void Ui_RebindWidget::setupUi (QWidget * *RebindWidget*) [inline]

Definition at line 163 of file ui_rebindwidget.h.

References actionTab, animBox, animButton, animKrBox, animOnceBox, animTab, applyButton, btnClearMacro, btnStartMacro, btnStopMacro, cancelButton, delayButtons, dpiBox, dpiButton, dpiCustLabel, dpiCustXBox, dpiCustYBox, dpiWarning, fnBox, fnButton, gridLayout, gridLayout_10, gridLayout_2, gridLayout_3, gridLayout_4, gridLayout_5, gridLayout_8, groupBox_2, groupBox_3, groupBox_4, horizontalLayout, horizontalLayout_2, horizontalLayout_3, horizontalSpacer, horizontalSpacer_10, horizontalSpacer_11, horizontalSpacer_12, horizontalSpacer_13, horizontalSpacer_14, horizontalSpacer_15, horizontalSpacer_16, horizontalSpacer_17, horizontalSpacer_18, horizontalSpacer_19, horizontalSpacer_2, horizontalSpacer_20, horizontalSpacer_21, horizontalSpacer_22, horizontalSpacer_23, horizontalSpacer_24, horizontalSpacer_25, horizontalSpacer_26, horizontalSpacer_3, horizontalSpacer_4, horizontalSpacer_5, horizontalSpacer_6, horizontalSpacer_7, horizontalSpacer_8, horizontalSpacer_9, keyTab, label, layoutWidget, layoutWidget1, layoutWidget2, layoutWidget_2, lbl_macro, lightBox, lightButton, lightWrapBox, lockBox, lockButton, macroTab, mb2Box, mb2Button, mbBox, mbButton, mediaBox, mediaButton, modBox, modButton, modeBox, modeButton, modeWrapBox, mouseTab, numBox, numButton, programKpBox, programKpButton, programKpExtra, programKpModeBox, programKpSIBox, programKrBox, programKrButton, programKrExtra, programKrModeBox, programKrSIBox, programTab, progTipLabel, pteMacroBox, pteMacroComment, pteMacroText, rb_delay_asTyped, rb_delay_default, rb_delay_no, resetButton, retranslateUi(), tabWidget, txtBuffer, typingBox, typingButton, unbindButton, verticalLayout, verticalLayout_2, verticalSpacer, verticalSpacer_10, verticalSpacer_11, verticalSpacer_2, verticalSpacer_3, verticalSpacer_4, verticalSpacer_5, verticalSpacer_6, verticalSpacer_7, verticalSpacer_8, wheelBox, and wheelButton.

Referenced by RebindWidget::RebindWidget().

```
164     {
165         if (RebindWidget->objectName().isEmpty())
166             RebindWidget->setObjectName(QStringLiteral("RebindWidget"));
167         RebindWidget->resize(687, 342);
168         QSizePolicy sizePolicy(QSizePolicy::Expanding, QSizePolicy::Preferred);
169         sizePolicy.setHorizontalStretch(0);
170         sizePolicy.setVerticalStretch(0);
171         sizePolicy.setHeightForWidth(RebindWidget->sizePolicy().hasHeightForWidth());
172         RebindWidget->setSizePolicy(sizePolicy);
173         RebindWidget->setStyleSheet(QStringLiteral("font-size: 12px ;"));
174         verticalLayout = new QVBoxLayout(RebindWidget);
175         verticalLayout->setObjectName(QStringLiteral("verticalLayout"));
176         tabWidget = new QTabWidget(RebindWidget);
177         tabWidget->setObjectName(QStringLiteral("tabWidget"));
178         tabWidget->setStyleSheet(QStringLiteral(""));
179         tabWidget->setUsesScrollButtons(false);
180         tabWidget->setDocumentMode(false);
181         keyTab = new QWidget();
182         keyTab->setObjectName(QStringLiteral("keyTab"));
183         sizePolicy.setHeightForWidth(keyTab->sizePolicy().hasHeightForWidth());
184         keyTab->setSizePolicy(sizePolicy);
185         gridLayout_2 = new QGridLayout(keyTab);
186         gridLayout_2->setObjectName(QStringLiteral("gridLayout_2"));
187         gridLayout_2->setContentsMargins(-1, -1, -1, 0);
188         numBox = new QComboBox(keyTab);
189         numBox->setObjectName(QStringLiteral("numBox"));
190         QSizePolicy sizePolicy1(QSizePolicy::Expanding, QSizePolicy::Fixed);
191         sizePolicy1.setHorizontalStretch(0);
192         sizePolicy1.setVerticalStretch(0);
193         sizePolicy1.setHeightForWidth(numBox->sizePolicy().hasHeightForWidth());
194         numBox->setSizePolicy(sizePolicy1);
195 
196         gridLayout_2->addWidget(numBox, 6, 4, 1, 1);
197 
198         typingBox = new QComboBox(keyTab);
199         typingBox->setObjectName(QStringLiteral("typingBox"));
200         sizePolicy1.setHeightForWidth(typingBox->sizePolicy().hasHeightForWidth());
201         typingBox->setSizePolicy(sizePolicy1);
202 
203         gridLayout_2->addWidget(typingBox, 1, 1, 1, 1);
204 
205         fnBox = new QComboBox(keyTab);
206         fnBox->setObjectName(QStringLiteral("fnBox"));
207         sizePolicy1.setHeightForWidth(fnBox->sizePolicy().hasHeightForWidth());
208         fnBox->setSizePolicy(sizePolicy1);
209 
210         gridLayout_2->addWidget(fnBox, 6, 1, 1, 1);
211 
212         verticalSpacer_10 = new QSpacerItem(0, 5, QSizePolicy::Minimum, QSizePolicy::Fixed
    );
213 
214         gridLayout_2->addItem(verticalSpacer_10, 9, 1, 1, 1);
```

```
215
216         verticalSpacer_3 = new QSpacerItem(20, 40, QSizePolicy::Minimum,
    QSizePolicy::Expanding);
217
218         gridLayout_2->addItem(verticalSpacer_3, 15, 2, 1, 1);
219
220         horizontalSpacer_2 = new QSpacerItem(5, 20, QSizePolicy::Fixed,
    QSizePolicy::Minimum);
221
222         gridLayout_2->addItem(horizontalSpacer_2, 1, 3, 1, 1);
223
224         typingButton = new QRadioButton(keyTab);
225         typingButton->setObjectName(QStringLiteral("typingButton"));
226         typingButton->setAutoExclusive(false);
227
228         gridLayout_2->addWidget(typingButton, 0, 0, 1, 2);
229
230         modBox = new QComboBox(keyTab);
231         modBox->setObjectName(QStringLiteral("modBox"));
232         sizePolicy1.setHeightForWidth(modBox->sizePolicy().hasHeightForWidth());
233         modBox->setSizePolicy(sizePolicy1);
234
235         gridLayout_2->addWidget(modBox, 1, 4, 1, 1);
236
237         horizontalSpacer_3 = new QSpacerItem(5, 20, QSizePolicy::Fixed,
    QSizePolicy::Minimum);
238
239         gridLayout_2->addItem(horizontalSpacer_3, 1, 0, 1, 1);
240
241         mediaButton = new QRadioButton(keyTab);
242         mediaButton->setObjectName(QStringLiteral("mediaButton"));
243         mediaButton->setAutoExclusive(false);
244
245         gridLayout_2->addWidget(mediaButton, 10, 0, 1, 2);
246
247         verticalSpacer_4 = new QSpacerItem(20, 5, QSizePolicy::Minimum, QSizePolicy::Fixed)
    ;
248
249         gridLayout_2->addItem(verticalSpacer_4, 2, 1, 1, 1);
250
251         horizontalSpacer_9 = new QSpacerItem(0, 20, QSizePolicy::Expanding,
    QSizePolicy::Minimum);
252
253         gridLayout_2->addItem(horizontalSpacer_9, 1, 5, 1, 2);
254
255         horizontalSpacer_4 = new QSpacerItem(0, 20, QSizePolicy::Expanding,
    QSizePolicy::Minimum);
256
257         gridLayout_2->addItem(horizontalSpacer_4, 1, 2, 1, 1);
258
259         modButton = new QRadioButton(keyTab);
260         modButton->setObjectName(QStringLiteral("modButton"));
261
262         gridLayout_2->addWidget(modButton, 0, 3, 1, 2);
263
264         fnButton = new QRadioButton(keyTab);
265         fnButton->setObjectName(QStringLiteral("fnButton"));
266         fnButton->setAutoExclusive(false);
267
268         gridLayout_2->addWidget(fnButton, 5, 0, 1, 2);
269
270         mediaBox = new QComboBox(keyTab);
271         mediaBox->setObjectName(QStringLiteral("mediaBox"));
272         sizePolicy1.setHeightForWidth(mediaBox->sizePolicy().hasHeightForWidth());
273         mediaBox->setSizePolicy(sizePolicy1);
274
275         gridLayout_2->addWidget(mediaBox, 11, 1, 1, 1);
276
277         numButton = new QRadioButton(keyTab);
278         numButton->setObjectName(QStringLiteral("numButton"));
279         numButton->setAutoExclusive(false);
280
281         gridLayout_2->addWidget(numButton, 5, 3, 1, 2);
282
283         horizontalSpacer_17 = new QSpacerItem(40, 20, QSizePolicy::Expanding,
    QSizePolicy::Minimum);
284
285         gridLayout_2->addItem(horizontalSpacer_17, 1, 7, 1, 1);
286
287         tabWidget->addTab(keyTab, QString());
288         mouseTab = new QWidget();
289         mouseTab->setObjectName(QStringLiteral("mouseTab"));
290         gridLayout = new QGridLayout(mouseTab);
291         gridLayout->setObjectName(QStringLiteral("gridLayout"));
292         dpiButton = new QRadioButton(mouseTab);
293         dpiButton->setObjectName(QStringLiteral("dpiButton"));
294         dpiButton->setAutoExclusive(false);
```

```
295
296         gridLayout->addWidget(dpiButton, 3, 3, 1, 5);
297
298         verticalSpacer_2 = new QSpacerItem(20, 40, QSizePolicy::Minimum,
    QSizePolicy::Expanding);
299
300         gridLayout->addItem(verticalSpacer_2, 5, 1, 1, 1);
301
302         verticalSpacer_8 = new QSpacerItem(20, 5, QSizePolicy::Minimum, QSizePolicy::Fixed)
    ;
303
304         gridLayout->addItem(verticalSpacer_8, 2, 1, 1, 1);
305
306         horizontalSpacer_18 = new QSpacerItem(5, 20, QSizePolicy::Fixed,
    QSizePolicy::Minimum);
307
308         gridLayout->addItem(horizontalSpacer_18, 1, 0, 1, 1);
309
310         horizontalSpacer_20 = new QSpacerItem(5, 20, QSizePolicy::Fixed,
    QSizePolicy::Minimum);
311
312         gridLayout->addItem(horizontalSpacer_20, 1, 3, 1, 1);
313
314         mb2Box = new QComboBox(mouseTab);
315         mb2Box->setObjectName(QStringLiteral("mb2Box"));
316         sizePolicy1.setHeightForWidth(mb2Box->sizePolicy().hasHeightForWidth());
317         mb2Box->setSizePolicy(sizePolicy1);
318
319         gridLayout->addWidget(mb2Box, 1, 4, 1, 4);
320
321         dpiBox = new QComboBox(mouseTab);
322         dpiBox->setObjectName(QStringLiteral("dpiBox"));
323
324         gridLayout->addWidget(dpiBox, 4, 4, 1, 1);
325
326         dpiCustYBox = new QSpinBox(mouseTab);
327         dpiCustYBox->setObjectName(QStringLiteral("dpiCustYBox"));
328         dpiCustYBox->setMinimum(400);
329         dpiCustYBox->setMaximum(8500);
330         dpiCustYBox->setSingleStep(50);
331
332         gridLayout->addWidget(dpiCustYBox, 4, 7, 1, 1);
333
334         mbBox = new QComboBox(mouseTab);
335         mbBox->setObjectName(QStringLiteral("mbBox"));
336         sizePolicy1.setHeightForWidth(mbBox->sizePolicy().hasHeightForWidth());
337         mbBox->setSizePolicy(sizePolicy1);
338
339         gridLayout->addWidget(mbBox, 1, 1, 1, 1);
340
341         wheelBox = new QComboBox(mouseTab);
342         wheelBox->setObjectName(QStringLiteral("wheelBox"));
343
344         gridLayout->addWidget(wheelBox, 4, 1, 1, 1);
345
346         mbButton = new QRadioButton(mouseTab);
347         mbButton->setObjectName(QStringLiteral("mbButton"));
348         mbButton->setAutoExclusive(false);
349
350         gridLayout->addWidget(mbButton, 0, 0, 1, 2);
351
352         dpiCustXBox = new QSpinBox(mouseTab);
353         dpiCustXBox->setObjectName(QStringLiteral("dpiCustXBox"));
354         dpiCustXBox->setMinimum(400);
355         dpiCustXBox->setMaximum(8500);
356         dpiCustXBox->setSingleStep(50);
357
358         gridLayout->addWidget(dpiCustXBox, 4, 5, 1, 1);
359
360         dpiWarning = new QLabel(mouseTab);
361         dpiWarning->setObjectName(QStringLiteral("dpiWarning"));
362         QSizePolicy sizePolicy2(QSizePolicy::Preferred, QSizePolicy::Fixed);
363         sizePolicy2.setHorizontalStretch(0);
364         sizePolicy2.setVerticalStretch(0);
365         sizePolicy2.setHeightForWidth(dpiWarning->sizePolicy().hasHeightForWidth());
366         dpiWarning->setSizePolicy(sizePolicy2);
367
368         gridLayout->addWidget(dpiWarning, 6, 0, 1, 10);
369
370         wheelButton = new QRadioButton(mouseTab);
371         wheelButton->setObjectName(QStringLiteral("wheelButton"));
372
373         gridLayout->addWidget(wheelButton, 3, 0, 1, 2);
374
375         horizontalSpacer_22 = new QSpacerItem(5, 20, QSizePolicy::Fixed,
    QSizePolicy::Minimum);
376
```

```
377         gridLayout->addItem(horizontalSpacer_22, 1, 8, 1, 1);
378
379         mb2Button = new QRadioButton(mouseTab);
380         mb2Button->setObjectName(QStringLiteral("mb2Button"));
381         mb2Button->setAutoExclusive(false);
382
383         gridLayout->addWidget(mb2Button, 0, 3, 1, 5);
384
385         dpiCustLabel = new QLabel(mouseTab);
386         dpiCustLabel->setObjectName(QStringLiteral("dpiCustLabel"));
387
388         gridLayout->addWidget(dpiCustLabel, 4, 6, 1, 1);
389
390         horizontalSpacer_19 = new QSpacerItem(0, 0, QSizePolicy::Expanding,
    QSizePolicy::Minimum);
391
392         gridLayout->addItem(horizontalSpacer_19, 1, 2, 1, 1);
393
394         horizontalSpacer_21 = new QSpacerItem(0, 20, QSizePolicy::Expanding,
    QSizePolicy::Minimum);
395
396         gridLayout->addItem(horizontalSpacer_21, 1, 9, 1, 1);
397
398         tabWidget->addTab(mouseTab, QString());
399         animTab = new QWidget();
400         animTab->setObjectName(QStringLiteral("animTab"));
401         gridLayout_5 = new QGridLayout(animTab);
402         gridLayout_5->setObjectName(QStringLiteral("gridLayout_5"));
403         verticalSpacer_11 = new QSpacerItem(20, 40, QSizePolicy::Minimum,
    QSizePolicy::Expanding);
404
405         gridLayout_5->addItem(verticalSpacer_11, 6, 2, 1, 1);
406
407         animBox = new QComboBox(animTab);
408         animBox->setObjectName(QStringLiteral("animBox"));
409         sizePolicy1.setHeightForWidth(animBox->sizePolicy().hasHeightForWidth());
410         animBox->setSizePolicy(sizePolicy1);
411
412         gridLayout_5->addWidget(animBox, 4, 2, 1, 1);
413
414         horizontalSpacer_24 = new QSpacerItem(20, 20, QSizePolicy::Fixed,
    QSizePolicy::Minimum);
415
416         gridLayout_5->addItem(horizontalSpacer_24, 4, 3, 1, 1);
417
418         horizontalSpacer_25 = new QSpacerItem(438, 20, QSizePolicy::Expanding,
    QSizePolicy::Minimum);
419
420         gridLayout_5->addItem(horizontalSpacer_25, 4, 7, 1, 1);
421
422         animKrBox = new QCheckBox(animTab);
423         animKrBox->setObjectName(QStringLiteral("animKrBox"));
424
425         gridLayout_5->addWidget(animKrBox, 4, 6, 1, 1);
426
427         animOnceBox = new QCheckBox(animTab);
428         animOnceBox->setObjectName(QStringLiteral("animOnceBox"));
429
430         gridLayout_5->addWidget(animOnceBox, 4, 4, 1, 1);
431
432         horizontalSpacer_26 = new QSpacerItem(20, 20, QSizePolicy::Fixed,
    QSizePolicy::Minimum);
433
434         gridLayout_5->addItem(horizontalSpacer_26, 4, 5, 1, 1);
435
436         animButton = new QRadioButton(animTab);
437         animButton->setObjectName(QStringLiteral("animButton"));
438         sizePolicy1.setHeightForWidth(animButton->sizePolicy().hasHeightForWidth());
439         animButton->setSizePolicy(sizePolicy1);
440
441         gridLayout_5->addWidget(animButton, 0, 1, 1, 2);
442
443         horizontalSpacer_23 = new QSpacerItem(5, 20, QSizePolicy::Fixed,
    QSizePolicy::Minimum);
444
445         gridLayout_5->addItem(horizontalSpacer_23, 4, 1, 1, 1);
446
447         tabWidget->addTab(animTab, QString());
448         actionTab = new QWidget();
449         actionTab->setObjectName(QStringLiteral("actionTab"));
450         sizePolicy.setHeightForWidth(actionTab->sizePolicy().hasHeightForWidth());
451         actionTab->setSizePolicy(sizePolicy);
452         gridLayout_3 = new QGridLayout(actionTab);
453         gridLayout_3->setObjectName(QStringLiteral("gridLayout_3"));
454         gridLayout_3->setContentsMargins(-1, -1, -1, 0);
455         horizontalSpacer_5 = new QSpacerItem(5, 20, QSizePolicy::Fixed,
    QSizePolicy::Minimum);
```

```
456
457         gridLayout_3->addItem(horizontalSpacer_5, 1, 0, 1, 1);
458
459         modeButton = new QRadioButton(actionTab);
460         modeButton->setObjectName(QStringLiteral("modeButton"));
461         modeButton->setAutoExclusive(false);
462
463         gridLayout_3->addWidget(modeButton, 0, 0, 1, 2);
464
465         modeWrapBox = new QCheckBox(actionTab);
466         modeWrapBox->setObjectName(QStringLiteral("modeWrapBox"));
467         modeWrapBox->setChecked(true);
468
469         gridLayout_3->addWidget(modeWrapBox, 1, 3, 1, 1);
470
471         lightWrapBox = new QCheckBox(actionTab);
472         lightWrapBox->setObjectName(QStringLiteral("lightWrapBox"));
473         lightWrapBox->setChecked(true);
474
475         gridLayout_3->addWidget(lightWrapBox, 4, 3, 1, 1);
476
477         verticalSpacer_7 = new QSpacerItem(20, 5, QSizePolicy::Minimum, QSizePolicy::Fixed)
    ;
478
479         gridLayout_3->addItem(verticalSpacer_7, 2, 1, 1, 1);
480
481         verticalSpacer_6 = new QSpacerItem(20, 5, QSizePolicy::Minimum, QSizePolicy::Fixed)
    ;
482
483         gridLayout_3->addItem(verticalSpacer_6, 5, 1, 1, 1);
484
485         lockButton = new QRadioButton(actionTab);
486         lockButton->setObjectName(QStringLiteral("lockButton"));
487         lockButton->setAutoExclusive(false);
488
489         gridLayout_3->addWidget(lockButton, 6, 0, 1, 2);
490
491         verticalSpacer_5 = new QSpacerItem(20, 40, QSizePolicy::Minimum,
    QSizePolicy::Expanding);
492
493         gridLayout_3->addItem(verticalSpacer_5, 8, 1, 1, 1);
494
495         lightBox = new QComboBox(actionTab);
496         lightBox->setObjectName(QStringLiteral("lightBox"));
497         sizePolicy1.setHeightForWidth(lightBox->sizePolicy().hasHeightForWidth());
498         lightBox->setSizePolicy(sizePolicy1);
499
500         gridLayout_3->addWidget(lightBox, 4, 1, 1, 1);
501
502         lockBox = new QComboBox(actionTab);
503         lockBox->setObjectName(QStringLiteral("lockBox"));
504         sizePolicy1.setHeightForWidth(lockBox->sizePolicy().hasHeightForWidth());
505         lockBox->setSizePolicy(sizePolicy1);
506
507         gridLayout_3->addWidget(lockBox, 7, 1, 1, 1);
508
509         lightButton = new QRadioButton(actionTab);
510         lightButton->setObjectName(QStringLiteral("lightButton"));
511         lightButton->setAutoExclusive(false);
512
513         gridLayout_3->addWidget(lightButton, 3, 0, 1, 2);
514
515         modeBox = new QComboBox(actionTab);
516         modeBox->setObjectName(QStringLiteral("modeBox"));
517         sizePolicy1.setHeightForWidth(modeBox->sizePolicy().hasHeightForWidth());
518         modeBox->setSizePolicy(sizePolicy1);
519
520         gridLayout_3->addWidget(modeBox, 1, 1, 1, 1);
521
522         horizontalSpacer_8 = new QSpacerItem(5, 20, QSizePolicy::Fixed,
    QSizePolicy::Minimum);
523
524         gridLayout_3->addItem(horizontalSpacer_8, 1, 2, 1, 1);
525
526         horizontalSpacer_10 = new QSpacerItem(40, 20, QSizePolicy::Expanding,
    QSizePolicy::Minimum);
527
528         gridLayout_3->addItem(horizontalSpacer_10, 7, 3, 1, 1);
529
530         horizontalSpacer_7 = new QSpacerItem(40, 20, QSizePolicy::Expanding,
    QSizePolicy::Minimum);
531
532         gridLayout_3->addItem(horizontalSpacer_7, 1, 4, 1, 1);
533
534         horizontalSpacer_6 = new QSpacerItem(40, 20, QSizePolicy::Expanding,
    QSizePolicy::Minimum);
535
```

```
536         gridLayout_3->addItem(horizontalSpacer_6, 1, 5, 1, 1);
537 
538         tabWidget->addTab(actionTab, QString());
539         programTab = new QWidget();
540         programTab->setObjectName(QStringLiteral("programTab"));
541         gridLayout_4 = new QGridLayout(programTab);
542         gridLayout_4->setObjectName(QStringLiteral("gridLayout_4"));
543         verticalSpacer = new QSpacerItem(20, 5, QSizePolicy::Minimum, QSizePolicy::Fixed);
544 
545         gridLayout_4->addItem(verticalSpacer, 2, 1, 1, 1);
546 
547         horizontalSpacer = new QSpacerItem(5, 20, QSizePolicy::Fixed, QSizePolicy::Minimum)
    ;
548 
549         gridLayout_4->addItem(horizontalSpacer, 1, 0, 1, 1);
550 
551         programKpButton = new QCheckBox(programTab);
552         programKpButton->setObjectName(QStringLiteral("programKpButton"));
553 
554         gridLayout_4->addWidget(programKpButton, 0, 0, 1, 2);
555 
556         programKpBox = new QLineEdit(programTab);
557         programKpBox->setObjectName(QStringLiteral("programKpBox"));
558 
559         gridLayout_4->addWidget(programKpBox, 1, 1, 1, 1);
560 
561         horizontalSpacer_11 = new QSpacerItem(40, 20, QSizePolicy::Expanding,
    QSizePolicy::Minimum);
562 
563         gridLayout_4->addItem(horizontalSpacer_11, 0, 2, 1, 2);
564 
565         horizontalSpacer_13 = new QSpacerItem(20, 20, QSizePolicy::Fixed,
    QSizePolicy::Minimum);
566 
567         gridLayout_4->addItem(horizontalSpacer_13, 1, 2, 1, 1);
568 
569         programKrBox = new QLineEdit(programTab);
570         programKrBox->setObjectName(QStringLiteral("programKrBox"));
571 
572         gridLayout_4->addWidget(programKrBox, 4, 1, 1, 1);
573 
574         programKrButton = new QCheckBox(programTab);
575         programKrButton->setObjectName(QStringLiteral("programKrButton"));
576 
577         gridLayout_4->addWidget(programKrButton, 3, 0, 1, 2);
578 
579         horizontalSpacer_12 = new QSpacerItem(40, 20, QSizePolicy::Expanding,
    QSizePolicy::Minimum);
580 
581         gridLayout_4->addItem(horizontalSpacer_12, 0, 4, 1, 1);
582 
583         programKpExtra = new QWidget(programTab);
584         programKpExtra->setObjectName(QStringLiteral("programKpExtra"));
585         horizontalLayout_2 = new QHBoxLayout(programKpExtra);
586         horizontalLayout_2->setSpacing(12);
587         horizontalLayout_2->setObjectName(QStringLiteral("horizontalLayout_2"));
588         horizontalLayout_2->setContentsMargins(0, 0, 0, 0);
589         programKpSIBox = new QCheckBox(programKpExtra);
590         programKpSIBox->setObjectName(QStringLiteral("programKpSIBox"));
591 
592         horizontalLayout_2->addWidget(programKpSIBox);
593 
594         programKpModeBox = new QComboBox(programKpExtra);
595         programKpModeBox->setObjectName(QStringLiteral("programKpModeBox"));
596 
597         horizontalLayout_2->addWidget(programKpModeBox);
598 
599         horizontalSpacer_15 = new QSpacerItem(40, 20, QSizePolicy::Expanding,
    QSizePolicy::Minimum);
600 
601         horizontalLayout_2->addItem(horizontalSpacer_15);
602 
603 
604         gridLayout_4->addWidget(programKpExtra, 1, 3, 1, 2);
605 
606         programKrExtra = new QWidget(programTab);
607         programKrExtra->setObjectName(QStringLiteral("programKrExtra"));
608         horizontalLayout_3 = new QHBoxLayout(programKrExtra);
609         horizontalLayout_3->setSpacing(12);
610         horizontalLayout_3->setObjectName(QStringLiteral("horizontalLayout_3"));
611         horizontalLayout_3->setContentsMargins(0, 0, 0, 0);
612         programKrSIBox = new QCheckBox(programKrExtra);
613         programKrSIBox->setObjectName(QStringLiteral("programKrSIBox"));
614 
615         horizontalLayout_3->addWidget(programKrSIBox);
616 
617         programKrModeBox = new QComboBox(programKrExtra);
```

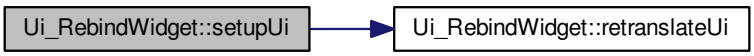
```
618         programKrModeBox->setObjectName(QStringLiteral("programKrModeBox"));
619
620         horizontalLayout_3->addWidget(programKrModeBox);
621
622         horizontalSpacer_16 = new QSpacerItem(40, 20, QSizePolicy::Expanding,
    QSizePolicy::Minimum);
623
624         horizontalLayout_3->addItem(horizontalSpacer_16);
625
626
627         gridLayout_4->addWidget(programKrExtra, 4, 3, 1, 2);
628
629         progTipLabel = new QLabel(programTab);
630         progTipLabel->setObjectName(QStringLiteral("progTipLabel"));
631         QSizePolicy sizePolicy3(QSizePolicy::Preferred, QSizePolicy::Expanding);
632         sizePolicy3.setHorizontalStretch(0);
633         sizePolicy3.setVerticalStretch(0);
634         sizePolicy3.setHeightForWidth(progTipLabel->sizePolicy().hasHeightForWidth());
635         progTipLabel->setSizePolicy(sizePolicy3);
636         progTipLabel->setTextInteractionFlags(Qt::LinksAccessibleByMouse|
    Qt::TextSelectableByMouse);
637
638         gridLayout_4->addWidget(progTipLabel, 5, 1, 1, 4);
639
640         tabWidget->addTab(programTab, QString());
641         macroTab = new QWidget();
642         macroTab->setObjectName(QStringLiteral("macroTab"));
643         layoutWidget1 = new QWidget(macroTab);
644         layoutWidget1->setObjectName(QStringLiteral("layoutWidget1"));
645         layoutWidget1->setGeometry(QRect(40, 12, 241, 161));
646         gridLayout_8 = new QGridLayout(layoutWidget1);
647         gridLayout_8->setObjectName(QStringLiteral("gridLayout_8"));
648         gridLayout_8->setSizeConstraint(QLayout::SetNoConstraint);
649         gridLayout_8->setContentsMargins(0, 0, 0, 0);
650         groupBox_2 = new QGroupBox(layoutWidget1);
651         groupBox_2->setObjectName(QStringLiteral("groupBox_2"));
652         pteMacroBox = new QPlainTextEdit(groupBox_2);
653         pteMacroBox->setObjectName(QStringLiteral("pteMacroBox"));
654         pteMacroBox->setGeometry(QRect(0, 20, 241, 131));
655         pteMacroBox->setFocusPolicy(Qt::WheelFocus);
656         pteMacroBox->setTabChangesFocus(true);
657
658         gridLayout_8->addWidget(groupBox_2, 0, 0, 1, 1);
659
660         pteMacroComment = new QPlainTextEdit(macroTab);
661         pteMacroComment->setObjectName(QStringLiteral("pteMacroComment"));
662         pteMacroComment->setEnabled(true);
663         pteMacroComment->setGeometry(QRect(410, 10, 311, 31));
664         pteMacroComment->setFocusPolicy(Qt::WheelFocus);
665         pteMacroComment->setContextMenuPolicy(Qt::NoContextMenu);
666         pteMacroComment->setAcceptDrops(true);
667         pteMacroComment->setToolTipDuration(1);
668         pteMacroComment->setInputMethodHints(Qt::ImhNone);
669         pteMacroComment->setVerticalScrollBarPolicy(Qt::ScrollBarAlwaysOff);
670         pteMacroComment->setTabChangesFocus(true);
671         pteMacroComment->setUndoRedoEnabled(true);
672         pteMacroComment->setReadOnly(false);
673         pteMacroComment->setTabStopWidth(1);
674         pteMacroText = new QPlainTextEdit(macroTab);
675         pteMacroText->setObjectName(QStringLiteral("pteMacroText"));
676         pteMacroText->setGeometry(QRect(410, 50, 311, 66));
677         pteMacroText->setFocusPolicy(Qt::StrongFocus);
678         pteMacroText->setContextMenuPolicy(Qt::NoContextMenu);
679         pteMacroText->setAcceptDrops(false);
680         pteMacroText->setInputMethodHints(Qt::ImhNoAutoUppercase);
681         pteMacroText->setUndoRedoEnabled(false);
682         pteMacroText->setReadOnly(false);
683         pteMacroText->setTabStopWidth(1);
684         layoutWidget = new QWidget(macroTab);
685         layoutWidget->setObjectName(QStringLiteral("layoutWidget"));
686         layoutWidget->setGeometry(QRect(300, 10, 131, 74));
687         verticalLayout_2 = new QVBoxLayout(layoutWidget);
688         verticalLayout_2->setObjectName(QStringLiteral("verticalLayout_2"));
689         verticalLayout_2->setSizeConstraint(QLayout::SetNoConstraint);
690         verticalLayout_2->setContentsMargins(0, 0, 0, 0);
691         groupBox_3 = new QGroupBox(layoutWidget);
692         groupBox_3->setObjectName(QStringLiteral("groupBox_3"));
693
694         verticalLayout_2->addWidget(groupBox_3);
695
696         groupBox_4 = new QGroupBox(layoutWidget);
697         groupBox_4->setObjectName(QStringLiteral("groupBox_4"));
698
699         verticalLayout_2->addWidget(groupBox_4);
700
701         layoutWidget_2 = new QWidget(macroTab);
702         layoutWidget_2->setObjectName(QStringLiteral("layoutWidget_2"));
```

```
703         layoutWidget_2->setGeometry(QRect(300, 160, 407, 36));
704         gridLayout_10 = new QGridLayout(layoutWidget_2);
705         gridLayout_10->setObjectName(QStringLiteral("gridLayout_10"));
706         gridLayout_10->setContentsMargins(0, 0, 0, 0);
707         btnStopMacro = new QPushButton(layoutWidget_2);
708         btnStopMacro->setObjectName(QStringLiteral("btnStopMacro"));
709         btnStopMacro->setEnabled(false);
710 
711         gridLayout_10->addWidget(btnStopMacro, 0, 2, 1, 1);
712 
713         btnClearMacro = new QPushButton(layoutWidget_2);
714         btnClearMacro->setObjectName(QStringLiteral("btnClearMacro"));
715 
716         gridLayout_10->addWidget(btnClearMacro, 0, 3, 1, 1);
717 
718         btnStartMacro = new QPushButton(layoutWidget_2);
719         btnStartMacro->setObjectName(QStringLiteral("btnStartMacro"));
720 
721         gridLayout_10->addWidget(btnStartMacro, 0, 1, 1, 1);
722 
723         label = new QLabel(layoutWidget_2);
724         label->setObjectName(QStringLiteral("label"));
725         label->setStyleSheet(QLatin1String("font: 75 10pt ; \n"
726 ""));
727 
728         gridLayout_10->addWidget(label, 0, 0, 1, 1);
729 
730         lbl_macro = new QLabel(macroTab);
731         lbl_macro->setObjectName(QStringLiteral("lbl_macro"));
732         lbl_macro->setGeometry(QRect(300, 130, 431, 27));
733         lbl_macro->setStyleSheet(QStringLiteral("font: 75 11pt ; color: darkblue;"));
734         layoutWidget2 = new QWidget(macroTab);
735         layoutWidget2->setObjectName(QStringLiteral("layoutWidget2"));
736         layoutWidget2->setGeometry(QRect(40, 170, 238, 28));
737         delayButtons = new QHBoxLayout(layoutWidget2);
738         delayButtons->setSpacing(0);
739         delayButtons->setObjectName(QStringLiteral("delayButtons"));
740         delayButtons->setSizeConstraint(QLayout::SetMinimumSize);
741         delayButtons->setContentsMargins(0, 0, 0, 0);
742         rb_delay_no = new QRadioButton(layoutWidget2);
743         rb_delay_no->setObjectName(QStringLiteral("rb_delay_no"));
744         rb_delay_no->setStyleSheet(QStringLiteral("font: 10pt ; color:  green;"));
745 
746         delayButtons->addWidget(rb_delay_no);
747 
748         rb_delay_default = new QRadioButton(layoutWidget2);
749         rb_delay_default->setObjectName(QStringLiteral("rb_delay_default"));
750         rb_delay_default->setStyleSheet(QStringLiteral("font: 10pt ; color:  green;"));
751         rb_delay_default->setChecked(false);
752         rb_delay_default->setAutoExclusive(true);
753 
754         delayButtons->addWidget(rb_delay_default);
755 
756         rb_delay_asTyped = new QRadioButton(layoutWidget2);
757         rb_delay_asTyped->setObjectName(QStringLiteral("rb_delay_asTyped"));
758         rb_delay_asTyped->setStyleSheet(QStringLiteral("font: 10pt ; color:  green;"));
759         rb_delay_asTyped->setChecked(true);
760 
761         delayButtons->addWidget(rb_delay_asTyped);
762 
763         txtBuffer = new QLabel(macroTab);
764         txtBuffer->setObjectName(QStringLiteral("txtBuffer"));
765         txtBuffer->setEnabled(false);
766         txtBuffer->setGeometry(QRect(300, 100, 0, 0));
767         QSizePolicy sizePolicy4(QSizePolicy::Fixed, QSizePolicy::Fixed);
768         sizePolicy4.setHorizontalStretch(0);
769         sizePolicy4.setVerticalStretch(0);
770         sizePolicy4.setHeightForWidth(txtBuffer->sizePolicy().hasHeightForWidth());
771         txtBuffer->setSizePolicy(sizePolicy4);
772         txtBuffer->setMaximumSize(QSize(0, 0));
773         txtBuffer->setContextMenuPolicy(Qt::NoContextMenu);
774 #ifndef QT_NO_TOOLTIP
775         txtBuffer->setToolTip(QStringLiteral(""));
776 #endif // QT_NO_TOOLTIP
777         txtBuffer->setToolTipDuration(0);
778 #ifndef QT_NO_STATUSTIP
779         txtBuffer->setStatusTip(QStringLiteral(""));
780 #endif // QT_NO_STATUSTIP
781 #ifndef QT_NO_WHATSTHIS
782         txtBuffer->setWhatsThis(QStringLiteral(""));
783 #endif // QT_NO_WHATSTHIS
784         txtBuffer->setFrameShadow(QFrame::Plain);
785         txtBuffer->setLineWidth(0);
786         txtBuffer->setTextFormat(Qt::PlainText);
787         txtBuffer->setIndent(0);
788         txtBuffer->setTextInteractionFlags(Qt::NoTextInteraction);
789         tabWidget->addTab(macroTab, QString());
```

```
790         pteMacroComment->raise();
791         layoutWidget->raise();
792         layoutWidget1->raise();
793         layoutWidget->raise();
794         pteMacroText->raise();
795         layoutWidget_2->raise();
796         lbl_macro->raise();
797         txtBuffer->raise();
798 
799         verticalLayout->addWidget(tabWidget);
800 
801         horizontalLayout = new QHBoxLayout();
802         horizontalLayout->setObjectName(QStringLiteral("horizontalLayout"));
803         horizontalSpacer_14 = new QSpacerItem(40, 20, QSizePolicy::Expanding,
    QSizePolicy::Minimum);
804 
805         horizontalLayout->addItem(horizontalSpacer_14);
806 
807         unbindButton = new QPushButton(RebindWidget);
808         unbindButton->setObjectName(QStringLiteral("unbindButton"));
809 
810         horizontalLayout->addWidget(unbindButton);
811 
812         resetButton = new QPushButton(RebindWidget);
813         resetButton->setObjectName(QStringLiteral("resetButton"));
814 
815         horizontalLayout->addWidget(resetButton);
816 
817         cancelButton = new QPushButton(RebindWidget);
818         cancelButton->setObjectName(QStringLiteral("cancelButton"));
819 
820         horizontalLayout->addWidget(cancelButton);
821 
822         applyButton = new QPushButton(RebindWidget);
823         applyButton->setObjectName(QStringLiteral("applyButton"));
824 
825         horizontalLayout->addWidget(applyButton);
826 
827 
828         verticalLayout->addLayout(horizontalLayout);
829 
830         QWidget::setTabOrder(tabWidget, typingButton);
831         QWidget::setTabOrder(typingButton, typingBox);
832         QWidget::setTabOrder(typingBox, modButton);
833         QWidget::setTabOrder(modButton, modBox);
834         QWidget::setTabOrder(modBox, fnButton);
835         QWidget::setTabOrder(fnButton, fnBox);
836         QWidget::setTabOrder(fnBox, numButton);
837         QWidget::setTabOrder(numButton, numBox);
838         QWidget::setTabOrder(numBox, mediaButton);
839         QWidget::setTabOrder(mediaButton, mediaBox);
840         QWidget::setTabOrder(mediaBox, mbButton);
841         QWidget::setTabOrder(mbButton, mbBox);
842         QWidget::setTabOrder(mbBox, mb2Button);
843         QWidget::setTabOrder(mb2Button, mb2Box);
844         QWidget::setTabOrder(mb2Box, wheelButton);
845         QWidget::setTabOrder(wheelButton, wheelBox);
846         QWidget::setTabOrder(wheelBox, dpiButton);
847         QWidget::setTabOrder(dpiButton, dpiBox);
848         QWidget::setTabOrder(dpiBox, dpiCustXBox);
849         QWidget::setTabOrder(dpiCustXBox, dpiCustYBox);
850         QWidget::setTabOrder(dpiCustYBox, animButton);
851         QWidget::setTabOrder(animButton, animBox);
852         QWidget::setTabOrder(animBox, animOnceBox);
853         QWidget::setTabOrder(animOnceBox, animKrBox);
854         QWidget::setTabOrder(animKrBox, modeButton);
855         QWidget::setTabOrder(modeButton, modeBox);
856         QWidget::setTabOrder(modeBox, modeWrapBox);
857         QWidget::setTabOrder(modeWrapBox, lightButton);
858         QWidget::setTabOrder(lightButton, lightBox);
859         QWidget::setTabOrder(lightBox, lightWrapBox);
860         QWidget::setTabOrder(lightWrapBox, lockButton);
861         QWidget::setTabOrder(lockButton, lockBox);
862         QWidget::setTabOrder(lockBox, programKpButton);
863         QWidget::setTabOrder(programKpButton, programKpBox);
864         QWidget::setTabOrder(programKpBox, programKpSIBox);
865         QWidget::setTabOrder(programKpSIBox, programKpModeBox);
866         QWidget::setTabOrder(programKpModeBox, programKrButton);
867         QWidget::setTabOrder(programKrButton, programKrBox);
868         QWidget::setTabOrder(programKrBox, programKrSIBox);
869         QWidget::setTabOrder(programKrSIBox, programKrModeBox);
870         QWidget::setTabOrder(programKrModeBox, pteMacroBox);
871         QWidget::setTabOrder(pteMacroBox, rb_delay_no);
872         QWidget::setTabOrder(rb_delay_no, rb_delay_default);
873         QWidget::setTabOrder(rb_delay_default, rb_delay_asTyped);
874         QWidget::setTabOrder(rb_delay_asTyped, pteMacroComment);
875         QWidget::setTabOrder(pteMacroComment, pteMacroText);
```

```
876         QWidget::setTabOrder(pteMacroText, btnStartMacro);
877         QWidget::setTabOrder(btnStartMacro, btnStopMacro);
878         QWidget::setTabOrder(btnStopMacro, btnClearMacro);
879         QWidget::setTabOrder(btnClearMacro, unbindButton);
880         QWidget::setTabOrder(unbindButton, resetButton);
881         QWidget::setTabOrder(resetButton, cancelButton);
882         QWidget::setTabOrder(cancelButton, applyButton);
883
884         retranslateUi(RebindWidget);
885         QObject::connect(btnClearMacro, SIGNAL(clicked()),
    pteMacroText, SLOT(clear()));
886         QObject::connect(btnClearMacro, SIGNAL(clicked()),
    pteMacroComment, SLOT(clear()));
887         QObject::connect(btnClearMacro, SIGNAL(clicked()),
    btnStopMacro, SLOT(animateClick()));
888         QObject::connect(cancelButton, SIGNAL(clicked()), txtBuffer, SLOT(clear()));
889         QObject::connect(btnClearMacro, SIGNAL(clicked()),
    txtBuffer, SLOT(clear()));
890         QObject::connect(btnStartMacro, SIGNAL(clicked()),
    txtBuffer, SLOT(clear()));
891         QObject::connect(unbindButton, SIGNAL(clicked()), txtBuffer, SLOT(clear()));
892         QObject::connect(resetButton, SIGNAL(clicked()), txtBuffer, SLOT(clear()));
893
894         tabWidget->setCurrentIndex(5);
895         applyButton->setDefault(true);
896
897
898         QMetaObject::connectSlotsByName(RebindWidget);
899     } // setupUi
```

Here is the call graph for this function:



Here is the caller graph for this function:

12.100.3 Field Documentation

12.100.3.1 QWidget* Ui_RebindWidget::actionTab

Definition at line 92 of file ui_rebindwidget.h.

Referenced by retranslateUi(), and setupUi().

12.100.3.2 QComboBox* Ui_RebindWidget::animBox

Definition at line 84 of file ui_rebindwidget.h.

Referenced by RebindWidget::applyChanges(), RebindWidget::on_animBox_currentIndexChanged(), RebindWidget::on_animButton_clicked(), retranslateUi(), RebindWidget::setBox(), RebindWidget::setSelection(), and

setupUi().

12.100.3.3 QRadioButton* Ui_RebindWidget::animButton

Definition at line 90 of file ui_rebindwidget.h.

Referenced by RebindWidget::on_animBox_currentIndexChanged(), retranslateUi(), and setupUi().

12.100.3.4 QCheckBox* Ui_RebindWidget::animKrBox

Definition at line 87 of file ui_rebindwidget.h.

Referenced by RebindWidget::applyChanges(), retranslateUi(), RebindWidget::setSelection(), and setupUi().

12.100.3.5 QCheckBox* Ui_RebindWidget::animOnceBox

Definition at line 88 of file ui_rebindwidget.h.

Referenced by RebindWidget::applyChanges(), retranslateUi(), RebindWidget::setSelection(), and setupUi().

12.100.3.6 QWidget* Ui_RebindWidget::animTab

Definition at line 81 of file ui_rebindwidget.h.

Referenced by retranslateUi(), and setupUi().

12.100.3.7 QPushButton* Ui_RebindWidget::applyButton

Definition at line 161 of file ui_rebindwidget.h.

Referenced by RebindWidget::on_btnStartMacro_clicked(), RebindWidget::on_btnStopMacro_clicked(), retranslateUi(), and setupUi().

12.100.3.8 QPushButton* Ui_RebindWidget::btnClearMacro

Definition at line 146 of file ui_rebindwidget.h.

Referenced by retranslateUi(), and setupUi().

12.100.3.9 QPushButton* Ui_RebindWidget::btnStartMacro

Definition at line 147 of file ui_rebindwidget.h.

Referenced by RebindWidget::on_btnStartMacro_clicked(), RebindWidget::on_btnStopMacro_clicked(), retranslateUi(), and setupUi().

12.100.3.10 QPushButton* Ui_RebindWidget::btnStopMacro

Definition at line 145 of file ui_rebindwidget.h.

Referenced by RebindWidget::on_btnStartMacro_clicked(), RebindWidget::on_btnStopMacro_clicked(), retranslateUi(), and setupUi().

12.100.3.11 QPushButton* Ui_RebindWidget::cancelButton

Definition at line 160 of file ui_rebindwidget.h.

Referenced by retranslateUi(), and setupUi().

12.100.3.12 QHBoxLayout* Ui_RebindWidget::delayButtons

Definition at line 151 of file ui_rebindwidget.h.

Referenced by setupUi().

12.100.3.13 QComboBox* Ui_RebindWidget::dpiBox

Definition at line 68 of file ui_rebindwidget.h.

Referenced by RebindWidget::applyChanges(), RebindWidget::on_dpiBox_currentIndexChanged(), RebindWidget::on_dpiButton_clicked(), retranslateUi(), RebindWidget::setBind(), RebindWidget::setBox(), RebindWidget::setSelection(), and setupUi().

12.100.3.14 QRadioButton* Ui_RebindWidget::dpiButton

Definition at line 62 of file ui_rebindwidget.h.

Referenced by RebindWidget::on_dpiBox_currentIndexChanged(), retranslateUi(), RebindWidget::setBind(), and setupUi().

12.100.3.15 QLabel* Ui_RebindWidget::dpiCustLabel

Definition at line 78 of file ui_rebindwidget.h.

Referenced by RebindWidget::on_dpiBox_currentIndexChanged(), retranslateUi(), RebindWidget::setBind(), and setupUi().

12.100.3.16 QSpinBox* Ui_RebindWidget::dpiCustXBox

Definition at line 73 of file ui_rebindwidget.h.

Referenced by RebindWidget::applyChanges(), RebindWidget::on_dpiBox_currentIndexChanged(), RebindWidget::setBind(), RebindWidget::setSelection(), and setupUi().

12.100.3.17 QSpinBox* Ui_RebindWidget::dpiCustYBox

Definition at line 69 of file ui_rebindwidget.h.

Referenced by RebindWidget::applyChanges(), RebindWidget::on_dpiBox_currentIndexChanged(), RebindWidget::setBind(), RebindWidget::setSelection(), and setupUi().

12.100.3.18 QLabel* Ui_RebindWidget::dpiWarning

Definition at line 74 of file ui_rebindwidget.h.

Referenced by retranslateUi(), RebindWidget::setBind(), and setupUi().

12.100.3.19 QComboBox* Ui_RebindWidget::fnBox

Definition at line 44 of file ui_rebindwidget.h.

Referenced by RebindWidget::applyChanges(), RebindWidget::on_fnBox_currentIndexChanged(), RebindWidget::on_fnButton_clicked(), RebindWidget::RebindWidget(), retranslateUi(), RebindWidget::setBox(), RebindWidget::setSelection(), and setupUi().

12.100.3.20 QRadioButton* Ui_RebindWidget::fnButton

Definition at line 56 of file ui_rebindwidget.h.

Referenced by RebindWidget::on_fnBox_currentIndexChanged(), retranslateUi(), and setupUi().

12.100.3.21 QGridLayout* Ui_RebindWidget::gridLayout

Definition at line 61 of file ui_rebindwidget.h.

Referenced by setupUi().

12.100.3.22 QGridLayout* Ui_RebindWidget::gridLayout_10

Definition at line 144 of file ui_rebindwidget.h.

Referenced by setupUi().

12.100.3.23 QGridLayout* Ui_RebindWidget::gridLayout_2

Definition at line 41 of file ui_rebindwidget.h.

Referenced by setupUi().

12.100.3.24 QGridLayout* Ui_RebindWidget::gridLayout_3

Definition at line 93 of file ui_rebindwidget.h.

Referenced by setupUi().

12.100.3.25 QGridLayout* Ui_RebindWidget::gridLayout_4

Definition at line 111 of file ui_rebindwidget.h.

Referenced by setupUi().

12.100.3.26 QGridLayout* Ui_RebindWidget::gridLayout_5

Definition at line 82 of file ui_rebindwidget.h.

Referenced by setupUi().

12.100.3.27 QGridLayout* Ui_RebindWidget::gridLayout_8

Definition at line 134 of file ui_rebindwidget.h.

Referenced by setupUi().

12.100.3.28 QGroupBox* Ui_RebindWidget::groupBox_2

Definition at line 135 of file ui_rebindwidget.h.

Referenced by retranslateUi(), and setupUi().

12.100.3.29 QGroupBox* Ui_RebindWidget::groupBox_3

Definition at line 141 of file ui_rebindwidget.h.

Referenced by retranslateUi(), and setupUi().

12.100.3.30 QGroupBox* Ui_RebindWidget::groupBox_4

Definition at line 142 of file ui_rebindwidget.h.

Referenced by retranslateUi(), and setupUi().

12.100.3.31 QHBoxLayout* Ui_RebindWidget::horizontalLayout

Definition at line 156 of file ui_rebindwidget.h.

Referenced by setupUi().

12.100.3.32 QHBoxLayout* Ui_RebindWidget::horizontalLayout_2

Definition at line 122 of file ui_rebindwidget.h.

Referenced by setupUi().

12.100.3.33 QHBoxLayout* Ui_RebindWidget::horizontalLayout_3

Definition at line 127 of file ui_rebindwidget.h.

Referenced by setupUi().

12.100.3.34 QSpacerItem* Ui_RebindWidget::horizontalSpacer

Definition at line 113 of file ui_rebindwidget.h.

Referenced by setupUi().

12.100.3.35 QSpacerItem* Ui_RebindWidget::horizontalSpacer_10

Definition at line 107 of file ui_rebindwidget.h.

Referenced by setupUi().

12.100.3.36 QSpacerItem* Ui_RebindWidget::horizontalSpacer_11

Definition at line 116 of file ui_rebindwidget.h.

Referenced by setupUi().

12.100.3.37 QSpacerItem* Ui_RebindWidget::horizontalSpacer_12

Definition at line 120 of file ui_rebindwidget.h.

Referenced by setupUi().

12.100.3.38 QSpacerItem* Ui_RebindWidget::horizontalSpacer_13

Definition at line 117 of file ui_rebindwidget.h.

Referenced by setupUi().

12.100.3.39 QSpacerItem* Ui_RebindWidget::horizontalSpacer_14

Definition at line 157 of file ui_rebindwidget.h.

Referenced by setupUi().

12.100.3.40 QSpacerItem* Ui_RebindWidget::horizontalSpacer_15

Definition at line 125 of file ui_rebindwidget.h.

Referenced by setupUi().

12.100.3.41 QSpacerItem* Ui_RebindWidget::horizontalSpacer_16

Definition at line 130 of file ui_rebindwidget.h.

Referenced by setupUi().

12.100.3.42 QSpacerItem* Ui_RebindWidget::horizontalSpacer_17

Definition at line 59 of file ui_rebindwidget.h.

Referenced by setupUi().

12.100.3.43 QSpacerItem* Ui_RebindWidget::horizontalSpacer_18

Definition at line 65 of file ui_rebindwidget.h.

Referenced by setupUi().

12.100.3.44 QSpacerItem* Ui_RebindWidget::horizontalSpacer_19

Definition at line 79 of file ui_rebindwidget.h.

Referenced by setupUi().

12.100.3.45 QSpacerItem* Ui_RebindWidget::horizontalSpacer_2

Definition at line 47 of file ui_rebindwidget.h.

Referenced by setupUi().

12.100.3.46 QSpacerItem* Ui_RebindWidget::horizontalSpacer_20

Definition at line 66 of file ui_rebindwidget.h.

Referenced by setupUi().

12.100.3.47 QSpacerItem* Ui_RebindWidget::horizontalSpacer_21

Definition at line 80 of file ui_rebindwidget.h.

Referenced by setupUi().

12.100.3.48 QSpacerItem* Ui_RebindWidget::horizontalSpacer_22

Definition at line 76 of file ui_rebindwidget.h.

Referenced by setupUi().

12.100.3.49 QSpacerItem* Ui_RebindWidget::horizontalSpacer_23

Definition at line 91 of file ui_rebindwidget.h.

Referenced by setupUi().

12.100.3.50 QSpacerItem* Ui_RebindWidget::horizontalSpacer_24

Definition at line 85 of file ui_rebindwidget.h.

Referenced by setupUi().

12.100.3.51 QSpacerItem* Ui_RebindWidget::horizontalSpacer_25

Definition at line 86 of file ui_rebindwidget.h.

Referenced by setupUi().

12.100.3.52 QSpacerItem* Ui_RebindWidget::horizontalSpacer_26

Definition at line 89 of file ui_rebindwidget.h.

Referenced by setupUi().

12.100.3.53 QSpacerItem* Ui_RebindWidget::horizontalSpacer_3

Definition at line 50 of file ui_rebindwidget.h.

Referenced by setupUi().

12.100.3.54 QSpacerItem* Ui_RebindWidget::horizontalSpacer_4

Definition at line 54 of file ui_rebindwidget.h.

Referenced by setupUi().

12.100.3.55 QSpacerItem* Ui_RebindWidget::horizontalSpacer_5

Definition at line 94 of file ui_rebindwidget.h.

Referenced by setupUi().

12.100.3.56 QSpacerItem* Ui_RebindWidget::horizontalSpacer_6

Definition at line 109 of file ui_rebindwidget.h.

Referenced by setupUi().

12.100.3.57 QSpacerItem* Ui_RebindWidget::horizontalSpacer_7

Definition at line 108 of file ui_rebindwidget.h.

Referenced by setupUi().

12.100.3.58 QSpacerItem* Ui_RebindWidget::horizontalSpacer_8

Definition at line 106 of file ui_rebindwidget.h.

Referenced by setupUi().

12.100.3.59 QSpacerItem* Ui_RebindWidget::horizontalSpacer_9

Definition at line 53 of file ui_rebindwidget.h.

Referenced by setupUi().

12.100.3.60 QWidget* Ui_RebindWidget::keyTab

Definition at line 40 of file ui_rebindwidget.h.

Referenced by retranslateUi(), and setupUi().

12.100.3.61 QLabel* Ui_RebindWidget::label

Definition at line 148 of file ui_rebindwidget.h.

Referenced by retranslateUi(), and setupUi().

12.100.3.62 QWidget* Ui_RebindWidget::layoutWidget

Definition at line 139 of file ui_rebindwidget.h.

Referenced by setupUi().

12.100.3.63 QWidget* Ui_RebindWidget::layoutWidget1

Definition at line 133 of file ui_rebindwidget.h.

Referenced by setupUi().

12.100.3.64 QWidget* Ui_RebindWidget::layoutWidget2

Definition at line 150 of file ui_rebindwidget.h.

Referenced by setupUi().

12.100.3.65 QWidget* Ui_RebindWidget::layoutWidget_2

Definition at line 143 of file ui_rebindwidget.h.

Referenced by setupUi().

12.100.3.66 QLabel* Ui_RebindWidget::lbl_macro

Definition at line 149 of file ui_rebindwidget.h.

Referenced by RebindWidget::helpStatus(), retranslateUi(), and setupUi().

12.100.3.67 QComboBox* Ui_RebindWidget::lightBox

Definition at line 102 of file ui_rebindwidget.h.

Referenced by RebindWidget::applyChanges(), RebindWidget::on_lightBox_currentIndexChanged(), RebindWidget::on_lightButton_clicked(), retranslateUi(), RebindWidget::setBox(), RebindWidget::setSelection(), and setupUi().

12.100.3.68 QRadioButton* Ui_RebindWidget::lightButton

Definition at line 104 of file ui_rebindwidget.h.

Referenced by RebindWidget::on_lightBox_currentIndexChanged(), retranslateUi(), and setupUi().

12.100.3.69 QCheckBox* Ui_RebindWidget::lightWrapBox

Definition at line 97 of file ui_rebindwidget.h.

Referenced by RebindWidget::applyChanges(), RebindWidget::on_lightBox_currentIndexChanged(), RebindWidget::RebindWidget(), retranslateUi(), RebindWidget::setSelection(), and setupUi().

12.100.3.70 QComboBox* Ui_RebindWidget::lockBox

Definition at line 103 of file ui_rebindwidget.h.

Referenced by RebindWidget::applyChanges(), RebindWidget::on_lockBox_currentIndexChanged(), RebindWidget::on_lockButton_clicked(), retranslateUi(), RebindWidget::setBox(), RebindWidget::setSelection(), and setupUi().

12.100.3.71 QRadioButton* Ui_RebindWidget::lockButton

Definition at line 100 of file ui_rebindwidget.h.

Referenced by RebindWidget::on_lockBox_currentIndexChanged(), retranslateUi(), and setupUi().

12.100.3.72 QWidget* Ui_RebindWidget::macroTab

Definition at line 132 of file ui_rebindwidget.h.

Referenced by retranslateUi(), and setupUi().

12.100.3.73 QComboBox* Ui_RebindWidget::mb2Box

Definition at line 67 of file ui_rebindwidget.h.

Referenced by RebindWidget::applyChanges(), RebindWidget::on_mb2Box_currentIndexChanged(), RebindWidget::on_mb2Button_clicked(), retranslateUi(), RebindWidget::setBox(), RebindWidget::setSelection(), and setupUi().

12.100.3.74 QRadioButton* Ui_RebindWidget::mb2Button

Definition at line 77 of file ui_rebindwidget.h.

Referenced by RebindWidget::on_mb2Box_currentIndexChanged(), retranslateUi(), and setupUi().

12.100.3.75 QComboBox* Ui_RebindWidget::mbBox

Definition at line 70 of file ui_rebindwidget.h.

Referenced by RebindWidget::applyChanges(), RebindWidget::on_mbBox_currentIndexChanged(), RebindWidget::on_mbButton_clicked(), retranslateUi(), RebindWidget::setBox(), RebindWidget::setSelection(), and setupUi().

12.100.3.76 QRadioButton* Ui_RebindWidget::mbButton

Definition at line 72 of file ui_rebindwidget.h.

Referenced by RebindWidget::on_mbBox_currentIndexChanged(), retranslateUi(), and setupUi().

12.100.3.77 QComboBox* Ui_RebindWidget::mediaBox

Definition at line 57 of file ui_rebindwidget.h.

Referenced by RebindWidget::applyChanges(), RebindWidget::on_mediaBox_currentIndexChanged(), RebindWidget::on_mediaButton_clicked(), RebindWidget::RebindWidget(), retranslateUi(), RebindWidget::setBox(), RebindWidget::setSelection(), and setupUi().

12.100.3.78 QRadioButton* Ui_RebindWidget::mediaButton

Definition at line 51 of file ui_rebindwidget.h.

Referenced by RebindWidget::on_mediaBox_currentIndexChanged(), retranslateUi(), and setupUi().

12.100.3.79 QComboBox* Ui_RebindWidget::modBox

Definition at line 49 of file ui_rebindwidget.h.

Referenced by RebindWidget::applyChanges(), RebindWidget::on_modBox_currentIndexChanged(), RebindWidget::on_modButton_clicked(), RebindWidget::RebindWidget(), retranslateUi(), RebindWidget::setBox(), RebindWidget::setSelection(), and setupUi().

12.100.3.80 QRadioButton* Ui_RebindWidget::modButton

Definition at line 55 of file ui_rebindwidget.h.

Referenced by RebindWidget::on_modBox_currentIndexChanged(), retranslateUi(), and setupUi().

12.100.3.81 QComboBox* Ui_RebindWidget::modeBox

Definition at line 105 of file ui_rebindwidget.h.

Referenced by RebindWidget::applyChanges(), RebindWidget::on_modeBox_currentIndexChanged(), RebindWidget::on_modeButton_clicked(), retranslateUi(), RebindWidget::setBind(), RebindWidget::setBox(), RebindWidget::setSelection(), and setupUi().

12.100.3.82 QRadioButton* Ui_RebindWidget::modeButton

Definition at line 95 of file ui_rebindwidget.h.

Referenced by RebindWidget::on_modeBox_currentIndexChanged(), retranslateUi(), and setupUi().

12.100.3.83 QCheckBox* Ui_RebindWidget::modeWrapBox

Definition at line 96 of file ui_rebindwidget.h.

Referenced by RebindWidget::applyChanges(), RebindWidget::on_modeBox_currentIndexChanged(), RebindWidget::RebindWidget(), retranslateUi(), RebindWidget::setSelection(), and setupUi().

12.100.3.84 QWidget* Ui_RebindWidget::mouseTab

Definition at line 60 of file ui_rebindwidget.h.

Referenced by retranslateUi(), and setupUi().

12.100.3.85 QComboBox* Ui_RebindWidget::numBox

Definition at line 42 of file ui_rebindwidget.h.

Referenced by RebindWidget::applyChanges(), RebindWidget::on_numBox_currentIndexChanged(), RebindWidget::on_numButton_clicked(), RebindWidget::RebindWidget(), retranslateUi(), RebindWidget::setBox(), RebindWidget::setSelection(), and setupUi().

12.100.3.86 QRadioButton* Ui_RebindWidget::numButton

Definition at line 58 of file ui_rebindwidget.h.

Referenced by RebindWidget::on_numBox_currentIndexChanged(), retranslateUi(), and setupUi().

12.100.3.87 QLineEdit* Ui_RebindWidget::programKpBox

Definition at line 115 of file ui_rebindwidget.h.

Referenced by RebindWidget::applyChanges(), RebindWidget::on_programKpBox_textChanged(), RebindWidget::on_programKpButton_clicked(), RebindWidget::setBox(), RebindWidget::setSelection(), and setupUi().

12.100.3.88 QCheckBox* Ui_RebindWidget::programKpButton

Definition at line 114 of file ui_rebindwidget.h.

Referenced by RebindWidget::on_programKpBox_textChanged(), retranslateUi(), RebindWidget::setBox(), and setupUi().

12.100.3.89 QWidget* Ui_RebindWidget::programKpExtra

Definition at line 121 of file ui_rebindwidget.h.

Referenced by RebindWidget::on_programKpBox_textChanged(), RebindWidget::RebindWidget(), and setupUi().

12.100.3.90 QComboBox* Ui_RebindWidget::programKpModeBox

Definition at line 124 of file ui_rebindwidget.h.

Referenced by RebindWidget::applyChanges(), RebindWidget::on_programKpSIBox_clicked(), retranslateUi(), RebindWidget::setSelection(), and setupUi().

12.100.3.91 QCheckBox* Ui_RebindWidget::programKpSIBox

Definition at line 123 of file ui_rebindwidget.h.

Referenced by RebindWidget::applyChanges(), retranslateUi(), RebindWidget::setSelection(), and setupUi().

12.100.3.92 QLineEdit* Ui_RebindWidget::programKrBox

Definition at line 118 of file ui_rebindwidget.h.

Referenced by RebindWidget::applyChanges(), RebindWidget::on_programKrBox_textChanged(), RebindWidget::on_programKrButton_clicked(), RebindWidget::setBox(), RebindWidget::setSelection(), and setupUi().

12.100.3.93 QCheckBox* Ui_RebindWidget::programKrButton

Definition at line 119 of file ui_rebindwidget.h.

Referenced by RebindWidget::on_programKrBox_textChanged(), retranslateUi(), RebindWidget::setBox(), and setupUi().

12.100.3.94 QWidget* Ui_RebindWidget::programKrExtra

Definition at line 126 of file ui_rebindwidget.h.

Referenced by RebindWidget::on_programKrBox_textChanged(), RebindWidget::RebindWidget(), and setupUi().

12.100.3.95 QComboBox* Ui_RebindWidget::programKrModeBox

Definition at line 129 of file ui_rebindwidget.h.

Referenced by RebindWidget::applyChanges(), RebindWidget::on_programKrSIBox_clicked(), retranslateUi(), RebindWidget::setSelection(), and setupUi().

12.100.3.96 QCheckBox* Ui_RebindWidget::programKrSIBox

Definition at line 128 of file ui_rebindwidget.h.

Referenced by RebindWidget::applyChanges(), retranslateUi(), RebindWidget::setSelection(), and setupUi().

12.100.3.97 QWidget* Ui_RebindWidget::programTab

Definition at line 110 of file ui_rebindwidget.h.

Referenced by retranslateUi(), and setupUi().

12.100.3.98 QLabel* Ui_RebindWidget::progTipLabel

Definition at line 131 of file ui_rebindwidget.h.

Referenced by RebindWidget::RebindWidget(), retranslateUi(), and setupUi().

12.100.3.99 QPlainTextEdit* Ui_RebindWidget::pteMacroBox

Definition at line 136 of file ui_rebindwidget.h.

Referenced by RebindWidget::applyChanges(), RebindWidget::convertMacroBox(), RebindWidget::on_btnStartMacro_clicked(), RebindWidget::on_pteMacroBox_textChanged(), RebindWidget::setBox(), RebindWidget::setSelection(), and setupUi().

12.100.3.100 QPlainTextEdit* Ui_RebindWidget::pteMacroComment

Definition at line 137 of file ui_rebindwidget.h.

Referenced by RebindWidget::applyChanges(), retranslateUi(), RebindWidget::setSelection(), and setupUi().

12.100.3.101 QPlainTextEdit* Ui_RebindWidget::pteMacroText

Definition at line 138 of file ui_rebindwidget.h.

Referenced by RebindWidget::applyChanges(), RebindWidget::on_btnStartMacro_clicked(), retranslateUi(), RebindWidget::setSelection(), and setupUi().

12.100.3.102 QRadioButton* Ui_RebindWidget::rb_delay_asTyped

Definition at line 154 of file ui_rebindwidget.h.

Referenced by RebindWidget::applyChanges(), RebindWidget::on_btnStartMacro_clicked(), RebindWidget::on_btnStopMacro_clicked(), retranslateUi(), RebindWidget::setCorrectRadioButton(), and setupUi().

12.100.3.103 QRadioButton* Ui_RebindWidget::rb_delay_default

Definition at line 153 of file ui_rebindwidget.h.

Referenced by RebindWidget::convertMacroBox(), RebindWidget::on_btnStartMacro_clicked(), RebindWidget::on_btnStopMacro_clicked(), retranslateUi(), RebindWidget::setCorrectRadioButton(), and setupUi().

12.100.3.104 QRadioButton* Ui_RebindWidget::rb_delay_no

Definition at line 152 of file ui_rebindwidget.h.

Referenced by RebindWidget::convertMacroBox(), RebindWidget::on_btnStartMacro_clicked(), RebindWidget::on_btnStopMacro_clicked(), retranslateUi(), RebindWidget::setCorrectRadioButton(), and setupUi().

12.100.3.105 QPushButton* Ui_RebindWidget::resetButton

Definition at line 159 of file ui_rebindwidget.h.

Referenced by RebindWidget::on_btnStartMacro_clicked(), RebindWidget::on_btnStopMacro_clicked(), retranslateUi(), and setupUi().

12.100.3.106 QTabWidget* Ui_RebindWidget::tabWidget

Definition at line 39 of file ui_rebindwidget.h.

Referenced by retranslateUi(), RebindWidget::setSelection(), and setupUi().

12.100.3.107 QLabel* Ui_RebindWidget::txtBuffer

Definition at line 155 of file ui_rebindwidget.h.

Referenced by RebindWidget::applyChanges(), RebindWidget::convertMacroBox(), retranslateUi(), RebindWidget::setBox(), RebindWidget::setSelection(), and setupUi().

12.100.3.108 QComboBox* Ui_RebindWidget::typingBox

Definition at line 43 of file ui_rebindwidget.h.

Referenced by RebindWidget::applyChanges(), RebindWidget::on_typingBox_currentIndexChanged(), RebindWidget::on_typingButton_clicked(), retranslateUi(), RebindWidget::setBind(), RebindWidget::setBox(), RebindWidget::setSelection(), and setupUi().

12.100.3.109 QRadioButton* Ui_RebindWidget::typingButton

Definition at line 48 of file ui_rebindwidget.h.

Referenced by RebindWidget::on_typingBox_currentIndexChanged(), retranslateUi(), and setupUi().

12.100.3.110 QPushButton* Ui_RebindWidget::unbindButton

Definition at line 158 of file ui_rebindwidget.h.

Referenced by RebindWidget::on_btnStartMacro_clicked(), RebindWidget::on_btnStopMacro_clicked(), retranslateUi(), and setupUi().

12.100.3.111 QVBoxLayout* Ui_RebindWidget::verticalLayout

Definition at line 38 of file ui_rebindwidget.h.

Referenced by setupUi().

12.100.3.112 QVBoxLayout* Ui_RebindWidget::verticalLayout_2

Definition at line 140 of file ui_rebindwidget.h.

Referenced by setupUi().

12.100.3.113 QSpacerItem* Ui_RebindWidget::verticalSpacer

Definition at line 112 of file ui_rebindwidget.h.

Referenced by setupUi().

12.100.3.114 QSpacerItem* Ui_RebindWidget::verticalSpacer_10

Definition at line 45 of file ui_rebindwidget.h.

Referenced by setupUi().

12.100.3.115 QSpacerItem* Ui_RebindWidget::verticalSpacer_11

Definition at line 83 of file ui_rebindwidget.h.

Referenced by setupUi().

12.100.3.116 QSpacerItem* Ui_RebindWidget::verticalSpacer_2

Definition at line 63 of file ui_rebindwidget.h.

Referenced by setupUi().

12.100.3.117 QSpacerItem* Ui_RebindWidget::verticalSpacer_3

Definition at line 46 of file ui_rebindwidget.h.

Referenced by setupUi().

12.100.3.118 QSpacerItem* Ui_RebindWidget::verticalSpacer_4

Definition at line 52 of file ui_rebindwidget.h.

Referenced by setupUi().

12.100.3.119 QSpacerItem* Ui_RebindWidget::verticalSpacer_5

Definition at line 101 of file ui_rebindwidget.h.

Referenced by setupUi().

12.100.3.120 QSpacerItem* Ui_RebindWidget::verticalSpacer_6

Definition at line 99 of file ui_rebindwidget.h.

Referenced by setupUi().

12.100.3.121 QSpacerItem* Ui_RebindWidget::verticalSpacer_7

Definition at line 98 of file ui_rebindwidget.h.

Referenced by setupUi().

12.100.3.122 QSpacerItem* Ui_RebindWidget::verticalSpacer_8

Definition at line 64 of file ui_rebindwidget.h.

Referenced by setupUi().

12.100.3.123 QComboBox* Ui_RebindWidget::wheelBox

Definition at line 71 of file ui_rebindwidget.h.

Referenced by RebindWidget::applyChanges(), RebindWidget::on_wheelBox_currentIndexChanged(), RebindWidget::on_wheelButton_clicked(), retranslateUi(), RebindWidget::setBox(), RebindWidget::setSelection(), and setupUi().

12.100.3.124 QRadioButton* Ui_RebindWidget::wheelButton

Definition at line 75 of file ui_rebindwidget.h.

Referenced by RebindWidget::on_wheelBox_currentIndexChanged(), retranslateUi(), and setupUi().

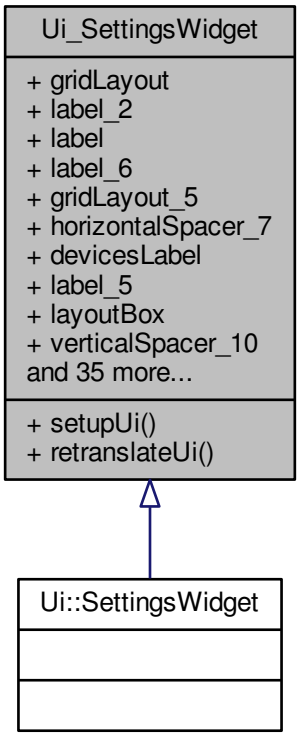
The documentation for this class was generated from the following file:

- src/ckb/ui_rebindwidget.h

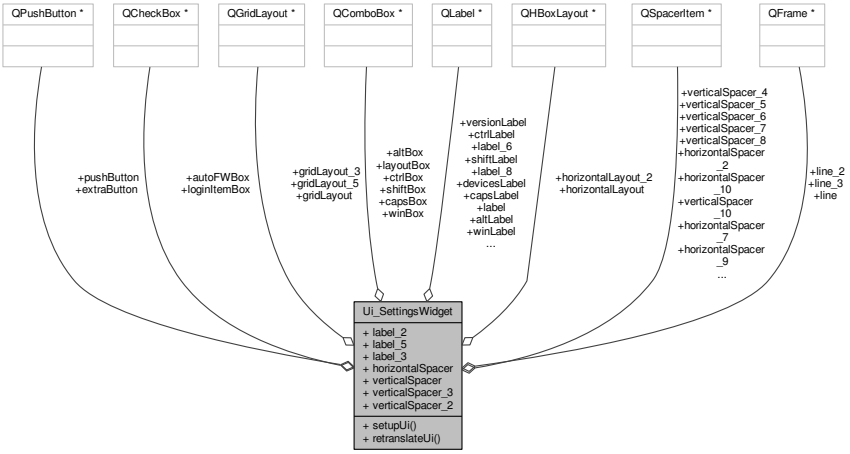
12.101 Ui_SettingsWidget Class Reference

`#include <src/ckb/ui_settingswidget.h>`

Inheritance diagram for Ui_SettingsWidget:



Collaboration diagram for Ui_SettingsWidget:



Public Member Functions

- void setupUi (QWidget *SettingsWidget)
- void retranslateUi (QWidget *SettingsWidget)

Data Fields

- QGridLayout * gridLayout
- QLabel * label_2
- QLabel * label
- QLabel * label_6
- QGridLayout * gridLayout_5
- QSpacerItem * horizontalSpacer_7
- QLabel * devicesLabel
- QLabel * label_5
- QComboBox * layoutBox
- QSpacerItem * verticalSpacer_10
- QHBoxLayout * horizontalLayout_2
- QLabel * label_3
- QSpacerItem * horizontalSpacer
- QPushButton * pushButton
- QSpacerItem * verticalSpacer
- QSpacerItem * verticalSpacer_8
- QCheckBox * loginItemBox
- QSpacerItem * verticalSpacer_6
- QCheckBox * autoFWBox
- QFrame * line_2
- QLabel * label_8
- QGridLayout * gridLayout_3
- QLabel * shiftLabel
- QLabel * capsLabel
- QSpacerItem * horizontalSpacer_10
- QComboBox * capsBox
- QLabel * ctrlLabel

- QSpacerItem * horizontalSpacer_9
- QComboBox * ctrlBox
- QComboBox * shiftBox
- QLabel * winLabel
- QComboBox * winBox
- QComboBox * altBox
- QLabel * altLabel
- QFrame * line
- QSpacerItem * verticalSpacer_3
- QSpacerItem * verticalSpacer_5
- QFrame * line_3
- QSpacerItem * verticalSpacer_4
- QSpacerItem * verticalSpacer_2
- QLabel * versionLabel
- QHBoxLayout * horizontalLayout
- QPushButton * extraButton
- QSpacerItem * horizontalSpacer_2
- QSpacerItem * verticalSpacer_7

12.101.1 Detailed Description

Definition at line 29 of file ui_settingswidget.h.

12.101.2 Member Function Documentation

12.101.2.1 void Ui_SettingsWidget::retranslateUi (QWidget * *SettingsWidget*) [inline]

Definition at line 356 of file ui_settingswidget.h.

References altLabel, autoFWBox, capsLabel, ctrlLabel, devicesLabel, extraButton, label, label_2, label_3, label_5, label_6, label_8, loginItemBox, pushButton, shiftLabel, versionLabel, and winLabel.

Referenced by setupUi().

```
357     {
358         SettingsWidget->setWindowTitle(QApplication::translate("SettingsWidget", "Form", Q_NULLPTR));
359         label_2->setText(QApplication::translate("SettingsWidget", "\302\251 2014-2016 <a href=\"
      https://github.com/ccMSC/\" style=\"text-decoration:none;\">ccMSC</a>.<br/>\302\251 2017 <a href=\"
      https://github.com/mattanger/ckb-next/graphs/contributors\" style=\"text-decoration:none;\">The ckb-next development
       team</a>.", Q_NULLPTR));
360         label->setText(QApplication::translate("SettingsWidget", "Modifier keys", Q_NULLPTR));
361         label_6->setText(QApplication::translate("SettingsWidget", "Application", Q_NULLPTR));
362         devicesLabel->setText(QApplication::translate("SettingsWidget", "No devices connected",
      Q_NULLPTR));
363         label_5->setText(QApplication::translate("SettingsWidget", "Keyboard layout:", Q_NULLPTR));
364         label_3->setText(QApplication::translate("SettingsWidget", "<a href=\"
      https://github.com/mattanger/ckb-next\" style=\"text-decoration:none;\">https://github.com/mattanger/ckb-next</a>", Q_NU
365         pushButton->setText(QApplication::translate("SettingsWidget", "Quit", Q_NULLPTR));
366 #ifndef QT_NO_TOOLTIP
367         loginItemBox->setToolTip(QApplication::translate("SettingsWidget", "ckb-next will be
      started when you log in to your computer.", Q_NULLPTR));
368 #endif // QT_NO_TOOLTIP
369         loginItemBox->setText(QApplication::translate("SettingsWidget", "Start ckb-next at
      login", Q_NULLPTR));
370 #ifndef QT_NO_TOOLTIP
371         autoFWBox->setToolTip(QApplication::translate("SettingsWidget", "You will be notified when
      new firmware versions are available. You'll have the option to install them immediately or wait until
      later.", Q_NULLPTR));
372 #endif // QT_NO_TOOLTIP
373         autoFWBox->setText(QApplication::translate("SettingsWidget", "Check for new firmware
      automatically", Q_NULLPTR));
374         label_8->setText(QApplication::translate("SettingsWidget", "These will override the keyboard
      profile. See \"Binding\" tab for more settings.", Q_NULLPTR));
375         shiftLabel->setText(QApplication::translate("SettingsWidget", "Shift:", Q_NULLPTR));
376         capsLabel->setText(QApplication::translate("SettingsWidget", "Caps Lock:", Q_NULLPTR));
377         ctrlLabel->setText(QApplication::translate("SettingsWidget", "Ctrl:", Q_NULLPTR));
378         winLabel->setText(QApplication::translate("SettingsWidget", "Windows:", Q_NULLPTR));
```

```
379         altLabel->setText(QApplication::translate("SettingsWidget", "Alt:", Q_NULLPTR));
380         versionLabel->setText(QApplication::translate("SettingsWidget", "ckb-next", Q_NULLPTR))
    ;
381         extraButton->setText(QApplication::translate("SettingsWidget", "More settings",
    Q_NULLPTR));
382     } // retranslateUi
```

Here is the caller graph for this function:

12.101.2.2 void Ui_SettingsWidget::setupUi (QWidget * *SettingsWidget*) [inline]

Definition at line 78 of file ui_settingswidget.h.

References altBox, altLabel, autoFWBox, capsBox, capsLabel, ctrlBox, ctrlLabel, devicesLabel, extraButton, gridLayout, gridLayout_3, gridLayout_5, horizontalLayout, horizontalLayout_2, horizontalSpacer, horizontalSpacer_10, horizontalSpacer_2, horizontalSpacer_7, horizontalSpacer_9, label, label_2, label_3, label_5, label_6, label_8, layoutBox, line, line_2, line_3, loginItemBox, pushButton, retranslateUi(), shiftBox, shiftLabel, versionLabel, verticalSpacer, verticalSpacer_10, verticalSpacer_2, verticalSpacer_3, verticalSpacer_4, verticalSpacer_5, verticalSpacer_6, verticalSpacer_7, verticalSpacer_8, winBox, and winLabel.

Referenced by SettingsWidget::SettingsWidget().

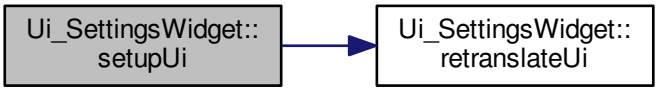
```
79     {
80         if (SettingsWidget->objectName().isEmpty())
81             SettingsWidget->setObjectName(QStringLiteral("SettingsWidget"));
82         SettingsWidget->resize(827, 654);
83         gridLayout = new QGridLayout(SettingsWidget);
84         gridLayout->setObjectName(QStringLiteral("gridLayout"));
85         label_2 = new QLabel(SettingsWidget);
86         label_2->setObjectName(QStringLiteral("label_2"));
87         label_2->setAlignment(Qt::AlignLeading|Qt::AlignLeft|Qt::AlignVCenter);
88         label_2->setOpenExternalLinks(true);
89 
90         gridLayout->addWidget(label_2, 20, 0, 1, 2);
91 
92         label = new QLabel(SettingsWidget);
93         label->setObjectName(QStringLiteral("label"));
94         QFont font;
95         font.setBold(true);
96         font.setWeight(75);
97         label->setFont(font);
98 
99         gridLayout->addWidget(label, 6, 0, 1, 2);
100 
101          label_6 = new QLabel(SettingsWidget);
102          label_6->setObjectName(QStringLiteral("label_6"));
103          label_6->setFont(font);
104 
105          gridLayout->addWidget(label_6, 12, 0, 1, 2);
106 
107          gridLayout_5 = new QGridLayout();
108          gridLayout_5->setObjectName(QStringLiteral("gridLayout_5"));
109          horizontalSpacer_7 = new QSpacerItem(40, 20, QSizePolicy::Expanding,
    QSizePolicy::Minimum);
110 
111          gridLayout_5->addItem(horizontalSpacer_7, 2, 2, 1, 1);
112 
113          devicesLabel = new QLabel(SettingsWidget);
114          devicesLabel->setObjectName(QStringLiteral("devicesLabel"));
115 
116          gridLayout_5->addWidget(devicesLabel, 0, 0, 1, 3);
117 
118          label_5 = new QLabel(SettingsWidget);
119          label_5->setObjectName(QStringLiteral("label_5"));
120 
```

```
121         gridLayout_5->addWidget(label_5, 2, 0, 1, 1);
122
123         layoutBox = new QComboBox(SettingsWidget);
124         layoutBox->setObjectName(QStringLiteral("layoutBox"));
125
126         gridLayout_5->addWidget(layoutBox, 2, 1, 1, 1);
127
128         verticalSpacer_10 = new QSpacerItem(20, 6, QSizePolicy::Minimum,
    QSizePolicy::Fixed);
129
130         gridLayout_5->addItem(verticalSpacer_10, 1, 0, 1, 1);
131
132
133         gridLayout->addLayout(gridLayout_5, 4, 1, 1, 1);
134
135         horizontalLayout_2 = new QHBoxLayout();
136         horizontalLayout_2->setObjectName(QStringLiteral("horizontalLayout_2"));
137         label_3 = new QLabel(SettingsWidget);
138         label_3->setObjectName(QStringLiteral("label_3"));
139         label_3->setOpenExternalLinks(true);
140
141         horizontalLayout_2->addWidget(label_3);
142
143         horizontalSpacer = new QSpacerItem(40, 20, QSizePolicy::Expanding,
    QSizePolicy::Minimum);
144
145         horizontalLayout_2->addItem(horizontalSpacer);
146
147         pushButton = new QPushButton(SettingsWidget);
148         pushButton->setObjectName(QStringLiteral("pushButton"));
149         pushButton->setFlat(false);
150
151         horizontalLayout_2->addWidget(pushButton);
152
153
154         gridLayout->addLayout(horizontalLayout_2, 21, 0, 1, 2);
155
156         verticalSpacer = new QSpacerItem(20, 40, QSizePolicy::Minimum, QSizePolicy::Expanding
    );
157
158         gridLayout->addItem(verticalSpacer, 19, 0, 1, 2);
159
160         verticalSpacer_8 = new QSpacerItem(0, 28, QSizePolicy::Minimum, QSizePolicy::Fixed)
    ;
161
162         gridLayout->addItem(verticalSpacer_8, 16, 0, 1, 1);
163
164         loginItemBox = new QCheckBox(SettingsWidget);
165         loginItemBox->setObjectName(QStringLiteral("loginItemBox"));
166
167         gridLayout->addWidget(loginItemBox, 16, 1, 1, 1);
168
169         verticalSpacer_6 = new QSpacerItem(0, 28, QSizePolicy::Minimum, QSizePolicy::Fixed)
    ;
170
171         gridLayout->addItem(verticalSpacer_6, 15, 0, 1, 1);
172
173         autoFWBox = new QCheckBox(SettingsWidget);
174         autoFWBox->setObjectName(QStringLiteral("autoFWBox"));
175         autoFWBox->setChecked(true);
176
177         gridLayout->addWidget(autoFWBox, 15, 1, 1, 1);
178
179         line_2 = new QFrame(SettingsWidget);
180         line_2->setObjectName(QStringLiteral("line_2"));
181         line_2->setFrameShape(QFrame::HLine);
182         line_2->setFrameShadow(QFrame::Sunken);
183
184         gridLayout->addWidget(line_2, 13, 0, 1, 2);
185
186         label_8 = new QLabel(SettingsWidget);
187         label_8->setObjectName(QStringLiteral("label_8"));
188         QFont font1;
189         font1.setItalic(true);
190         label_8->setFont(font1);
191
192         gridLayout->addWidget(label_8, 8, 1, 1, 1);
193
194         gridLayout_3 = new QGridLayout();
195         gridLayout_3->setObjectName(QStringLiteral("gridLayout_3"));
196         shiftLabel = new QLabel(SettingsWidget);
197         shiftLabel->setObjectName(QStringLiteral("shiftLabel"));
198         QSizePolicy sizePolicy(QSizePolicy::Expanding, QSizePolicy::Preferred);
199         sizePolicy.setHorizontalStretch(0);
200         sizePolicy.setVerticalStretch(0);
201         sizePolicy.setHeightForWidth(shiftLabel->sizePolicy().hasHeightForWidth());
202         shiftLabel->setSizePolicy(sizePolicy);
```

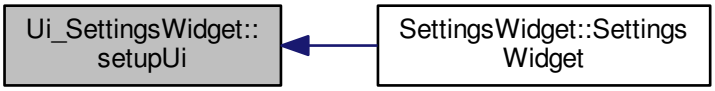
```
203         shiftLabel->setAlignment(Qt::AlignRight|Qt::AlignTrailing|Qt::AlignVCenter);
204
205         gridLayout_3->addWidget(shiftLabel, 0, 3, 1, 1);
206
207         capsLabel = new QLabel(SettingsWidget);
208         capsLabel->setObjectName(QStringLiteral("capsLabel"));
209         capsLabel->setAlignment(Qt::AlignRight|Qt::AlignTrailing|Qt::AlignVCenter);
210
211         gridLayout_3->addWidget(capsLabel, 0, 1, 1, 1);
212
213         horizontalSpacer_10 = new QSpacerItem(0, 0, QSizePolicy::Fixed,
    QSizePolicy::Minimum);
214
215         gridLayout_3->addItem(horizontalSpacer_10, 0, 0, 1, 1);
216
217         capsBox = new QComboBox(SettingsWidget);
218         capsBox->setObjectName(QStringLiteral("capsBox"));
219         QSizePolicy sizePolicy1(QSizePolicy::Expanding, QSizePolicy::Fixed);
220         sizePolicy1.setHorizontalStretch(0);
221         sizePolicy1.setVerticalStretch(0);
222         sizePolicy1.setHeightForWidth(capsBox->sizePolicy().hasHeightForWidth());
223         capsBox->setSizePolicy(sizePolicy1);
224
225         gridLayout_3->addWidget(capsBox, 0, 2, 1, 1);
226
227         ctrlLabel = new QLabel(SettingsWidget);
228         ctrlLabel->setObjectName(QStringLiteral("ctrlLabel"));
229         ctrlLabel->setAlignment(Qt::AlignRight|Qt::AlignTrailing|Qt::AlignVCenter);
230
231         gridLayout_3->addWidget(ctrlLabel, 1, 1, 1, 1);
232
233         horizontalSpacer_9 = new QSpacerItem(40, 20, QSizePolicy::Expanding,
    QSizePolicy::Minimum);
234
235         gridLayout_3->addItem(horizontalSpacer_9, 1, 7, 1, 1);
236
237         ctrlBox = new QComboBox(SettingsWidget);
238         ctrlBox->setObjectName(QStringLiteral("ctrlBox"));
239
240         gridLayout_3->addWidget(ctrlBox, 1, 2, 1, 1);
241
242         shiftBox = new QComboBox(SettingsWidget);
243         shiftBox->setObjectName(QStringLiteral("shiftBox"));
244         sizePolicy1.setHeightForWidth(shiftBox->sizePolicy().hasHeightForWidth());
245         shiftBox->setSizePolicy(sizePolicy1);
246
247         gridLayout_3->addWidget(shiftBox, 0, 4, 1, 1);
248
249         winLabel = new QLabel(SettingsWidget);
250         winLabel->setObjectName(QStringLiteral("winLabel"));
251         sizePolicy.setHeightForWidth(winLabel->sizePolicy().hasHeightForWidth());
252         winLabel->setSizePolicy(sizePolicy);
253         winLabel->setAlignment(Qt::AlignRight|Qt::AlignTrailing|Qt::AlignVCenter);
254
255         gridLayout_3->addWidget(winLabel, 1, 3, 1, 1);
256
257         winBox = new QComboBox(SettingsWidget);
258         winBox->setObjectName(QStringLiteral("winBox"));
259         sizePolicy1.setHeightForWidth(winBox->sizePolicy().hasHeightForWidth());
260         winBox->setSizePolicy(sizePolicy1);
261
262         gridLayout_3->addWidget(winBox, 1, 4, 1, 1);
263
264         altBox = new QComboBox(SettingsWidget);
265         altBox->setObjectName(QStringLiteral("altBox"));
266         sizePolicy1.setHeightForWidth(altBox->sizePolicy().hasHeightForWidth());
267         altBox->setSizePolicy(sizePolicy1);
268
269         gridLayout_3->addWidget(altBox, 1, 6, 1, 1);
270
271         altLabel = new QLabel(SettingsWidget);
272         altLabel->setObjectName(QStringLiteral("altLabel"));
273         sizePolicy.setHeightForWidth(altLabel->sizePolicy().hasHeightForWidth());
274         altLabel->setSizePolicy(sizePolicy);
275         altLabel->setAlignment(Qt::AlignRight|Qt::AlignTrailing|Qt::AlignVCenter);
276
277         gridLayout_3->addWidget(altLabel, 1, 5, 1, 1);
278
279
280         gridLayout->addLayout(gridLayout_3, 10, 1, 1, 1);
281
282         line = new QFrame(SettingsWidget);
283         line->setObjectName(QStringLiteral("line"));
284         line->setFrameShape(QFrame::HLine);
285         line->setFrameShadow(QFrame::Sunken);
286
287         gridLayout->addWidget(line, 1, 0, 1, 2);
```

```
288
289         verticalSpacer_3 = new QSpacerItem(0, 5, QSizePolicy::Minimum, QSizePolicy::Fixed);
290
291         gridLayout->addItem(verticalSpacer_3, 2, 1, 1, 1);
292
293         verticalSpacer_5 = new QSpacerItem(20, 12, QSizePolicy::Minimum, QSizePolicy::Fixed
    );
294
295         gridLayout->addItem(verticalSpacer_5, 11, 1, 1, 1);
296
297         line_3 = new QFrame(SettingsWidget);
298         line_3->setObjectName(QStringLiteral("line_3"));
299         QSizePolicy sizePolicy2(QSizePolicy::Minimum, QSizePolicy::Fixed);
300         sizePolicy2.setHorizontalStretch(0);
301         sizePolicy2.setVerticalStretch(0);
302         sizePolicy2.setHeightForWidth(line_3->sizePolicy().hasHeightForWidth());
303         line_3->setSizePolicy(sizePolicy2);
304         line_3->setFrameShape(QFrame::HLine);
305         line_3->setFrameShadow(QFrame::Sunken);
306
307         gridLayout->addWidget(line_3, 7, 0, 1, 2);
308
309         verticalSpacer_4 = new QSpacerItem(20, 6, QSizePolicy::Minimum, QSizePolicy::Fixed)
    ;
310
311         gridLayout->addItem(verticalSpacer_4, 9, 1, 1, 1);
312
313         verticalSpacer_2 = new QSpacerItem(20, 12, QSizePolicy::Minimum, QSizePolicy::Fixed
    );
314
315         gridLayout->addItem(verticalSpacer_2, 5, 1, 1, 1);
316
317         versionLabel = new QLabel(SettingsWidget);
318         versionLabel->setObjectName(QStringLiteral("versionLabel"));
319         versionLabel->setFont(font);
320
321         gridLayout->addWidget(versionLabel, 0, 0, 1, 2);
322
323         horizontalLayout = new QHBoxLayout();
324         horizontalLayout->setObjectName(QStringLiteral("horizontalLayout"));
325         extraButton = new QPushButton(SettingsWidget);
326         extraButton->setObjectName(QStringLiteral("extraButton"));
327
328         horizontalLayout->addWidget(extraButton);
329
330         horizontalSpacer_2 = new QSpacerItem(40, 20, QSizePolicy::Expanding,
    QSizePolicy::Minimum);
331
332         horizontalLayout->addItem(horizontalSpacer_2);
333
334
335         gridLayout->addLayout(horizontalLayout, 17, 1, 1, 1);
336
337         verticalSpacer_7 = new QSpacerItem(0, 28, QSizePolicy::Minimum, QSizePolicy::Fixed)
    ;
338
339         gridLayout->addItem(verticalSpacer_7, 17, 0, 1, 1);
340
341         QWidget::setTabOrder(layoutBox, capsBox);
342         QWidget::setTabOrder(capsBox, shiftBox);
343         QWidget::setTabOrder(shiftBox, ctrlBox);
344         QWidget::setTabOrder(ctrlBox, winBox);
345         QWidget::setTabOrder(winBox, altBox);
346         QWidget::setTabOrder(altBox, autoFWBox);
347         QWidget::setTabOrder(autoFWBox, loginItemBox);
348         QWidget::setTabOrder(loginItemBox, extraButton);
349         QWidget::setTabOrder(extraButton, pushButton);
350
351         retranslateUi(SettingsWidget);
352
353         QMetaObject::connectSlotsByName(SettingsWidget);
354     } // setupUi
```

Here is the call graph for this function:



Here is the caller graph for this function:



12.101.3 Field Documentation

12.101.3.1 QComboBox* Ui_SettingsWidget::altBox

Definition at line 64 of file ui_settingswidget.h.

Referenced by SettingsWidget::SettingsWidget(), setupUi(), and SettingsWidget::updateModifiers().

12.101.3.2 QLabel* Ui_SettingsWidget::altLabel

Definition at line 65 of file ui_settingswidget.h.

Referenced by retranslateUi(), and setupUi().

12.101.3.3 QCheckBox* Ui_SettingsWidget::autoFWBox

Definition at line 50 of file ui_settingswidget.h.

Referenced by retranslateUi(), SettingsWidget::SettingsWidget(), and setupUi().

12.101.3.4 QComboBox* Ui_SettingsWidget::capsBox

Definition at line 57 of file ui_settingswidget.h.

Referenced by SettingsWidget::SettingsWidget(), setupUi(), and SettingsWidget::updateModifiers().

12.101.3.5 QLabel* Ui_SettingsWidget::capsLabel

Definition at line 55 of file ui_settingswidget.h.

Referenced by retranslateUi(), and setupUi().

12.101.3.6 QComboBox* Ui_SettingsWidget::ctrlBox

Definition at line 60 of file ui_settingswidget.h.

Referenced by SettingsWidget::SettingsWidget(), setupUi(), and SettingsWidget::updateModifiers().

12.101.3.7 QLabel* Ui_SettingsWidget::ctrlLabel

Definition at line 58 of file ui_settingswidget.h.

Referenced by retranslateUi(), and setupUi().

12.101.3.8 QLabel* Ui_SettingsWidget::devicesLabel

Definition at line 38 of file ui_settingswidget.h.

Referenced by retranslateUi(), SettingsWidget::setStatus(), and setupUi().

12.101.3.9 QPushButton* Ui_SettingsWidget::extraButton

Definition at line 74 of file ui_settingswidget.h.

Referenced by retranslateUi(), and setupUi().

12.101.3.10 QGridLayout* Ui_SettingsWidget::gridLayout

Definition at line 32 of file ui_settingswidget.h.

Referenced by setupUi().

12.101.3.11 QGridLayout* Ui_SettingsWidget::gridLayout_3

Definition at line 53 of file ui_settingswidget.h.

Referenced by setupUi().

12.101.3.12 QGridLayout* Ui_SettingsWidget::gridLayout_5

Definition at line 36 of file ui_settingswidget.h.

Referenced by setupUi().

12.101.3.13 QHBoxLayout* Ui_SettingsWidget::horizontalLayout

Definition at line 73 of file ui_settingswidget.h.

Referenced by setupUi().

12.101.3.14 QHBoxLayout* Ui_SettingsWidget::horizontalLayout_2

Definition at line 42 of file ui_settingswidget.h.

Referenced by setupUi().

12.101.3.15 QSpacerItem* Ui_SettingsWidget::horizontalSpacer

Definition at line 44 of file ui_settingswidget.h.

Referenced by setupUi().

12.101.3.16 QSpacerItem* Ui_SettingsWidget::horizontalSpacer_10

Definition at line 56 of file ui_settingswidget.h.

Referenced by setupUi().

12.101.3.17 QSpacerItem* Ui_SettingsWidget::horizontalSpacer_2

Definition at line 75 of file ui_settingswidget.h.

Referenced by setupUi().

12.101.3.18 QSpacerItem* Ui_SettingsWidget::horizontalSpacer_7

Definition at line 37 of file ui_settingswidget.h.

Referenced by setupUi().

12.101.3.19 QSpacerItem* Ui_SettingsWidget::horizontalSpacer_9

Definition at line 59 of file ui_settingswidget.h.

Referenced by setupUi().

12.101.3.20 QLabel* Ui_SettingsWidget::label

Definition at line 34 of file ui_settingswidget.h.

Referenced by retranslateUi(), and setupUi().

12.101.3.21 QLabel* Ui_SettingsWidget::label_2

Definition at line 33 of file ui_settingswidget.h.

Referenced by retranslateUi(), and setupUi().

12.101.3.22 QLabel* Ui_SettingsWidget::label_3

Definition at line 43 of file ui_settingswidget.h.

Referenced by retranslateUi(), and setupUi().

12.101.3.23 QLabel* Ui_SettingsWidget::label_5

Definition at line 39 of file ui_settingswidget.h.

Referenced by retranslateUi(), and setupUi().

12.101.3.24 QLabel* Ui_SettingsWidget::label_6

Definition at line 35 of file ui_settingswidget.h.

Referenced by retranslateUi(), and setupUi().

12.101.3.25 QLabel* Ui_SettingsWidget::label_8

Definition at line 52 of file ui_settingswidget.h.

Referenced by retranslateUi(), and setupUi().

12.101.3.26 QComboBox* Ui_SettingsWidget::layoutBox

Definition at line 40 of file ui_settingswidget.h.

Referenced by SettingsWidget::SettingsWidget(), setupUi(), and SettingsWidget::showLayoutDialog().

12.101.3.27 QFrame* Ui_SettingsWidget::line

Definition at line 66 of file ui_settingswidget.h.

Referenced by setupUi().

12.101.3.28 QFrame* Ui_SettingsWidget::line_2

Definition at line 51 of file ui_settingswidget.h.

Referenced by setupUi().

12.101.3.29 QFrame* Ui_SettingsWidget::line_3

Definition at line 69 of file ui_settingswidget.h.

Referenced by setupUi().

12.101.3.30 QCheckBox* Ui_SettingsWidget::loginItemBox

Definition at line 48 of file ui_settingswidget.h.

Referenced by retranslateUi(), SettingsWidget::SettingsWidget(), and setupUi().

12.101.3.31 QPushButton* Ui_SettingsWidget::pushButton

Definition at line 45 of file ui_settingswidget.h.

Referenced by retranslateUi(), and setupUi().

12.101.3.32 QComboBox* Ui_SettingsWidget::shiftBox

Definition at line 61 of file ui_settingswidget.h.

Referenced by SettingsWidget::SettingsWidget(), setupUi(), and SettingsWidget::updateModifiers().

12.101.3.33 QLabel* Ui_SettingsWidget::shiftLabel

Definition at line 54 of file ui_settingswidget.h.

Referenced by retranslateUi(), and setupUi().

12.101.3.34 QLabel* Ui_SettingsWidget::versionLabel

Definition at line 72 of file ui_settingswidget.h.

Referenced by retranslateUi(), setupUi(), and SettingsWidget::setVersion().

12.101.3.35 QSpacerItem* Ui_SettingsWidget::verticalSpacer

Definition at line 46 of file ui_settingswidget.h.

Referenced by setupUi().

12.101.3.36 QSpacerItem* Ui_SettingsWidget::verticalSpacer_10

Definition at line 41 of file ui_settingswidget.h.

Referenced by setupUi().

12.101.3.37 QSpacerItem* Ui_SettingsWidget::verticalSpacer_2

Definition at line 71 of file ui_settingswidget.h.

Referenced by setupUi().

12.101.3.38 QSpacerItem* Ui_SettingsWidget::verticalSpacer_3

Definition at line 67 of file ui_settingswidget.h.

Referenced by setupUi().

12.101.3.39 QSpacerItem* Ui_SettingsWidget::verticalSpacer_4

Definition at line 70 of file ui_settingswidget.h.

Referenced by setupUi().

12.101.3.40 QSpacerItem* Ui_SettingsWidget::verticalSpacer_5

Definition at line 68 of file ui_settingswidget.h.

Referenced by setupUi().

12.101.3.41 QSpacerItem* Ui_SettingsWidget::verticalSpacer_6

Definition at line 49 of file ui_settingswidget.h.

Referenced by setupUi().

12.101.3.42 QSpacerItem* Ui_SettingsWidget::verticalSpacer_7

Definition at line 76 of file ui_settingswidget.h.

Referenced by setupUi().

12.101.3.43 QSpacerItem* Ui_SettingsWidget::verticalSpacer_8

Definition at line 47 of file ui_settingswidget.h.

Referenced by setupUi().

12.101.3.44 QComboBox* Ui_SettingsWidget::winBox

Definition at line 63 of file ui_settingswidget.h.

Referenced by SettingsWidget::SettingsWidget(), setupUi(), and SettingsWidget::updateModifiers().

12.101.3.45 QLabel* Ui_SettingsWidget::winLabel

Definition at line 62 of file ui_settingswidget.h.

Referenced by retranslateUi(), and setupUi().

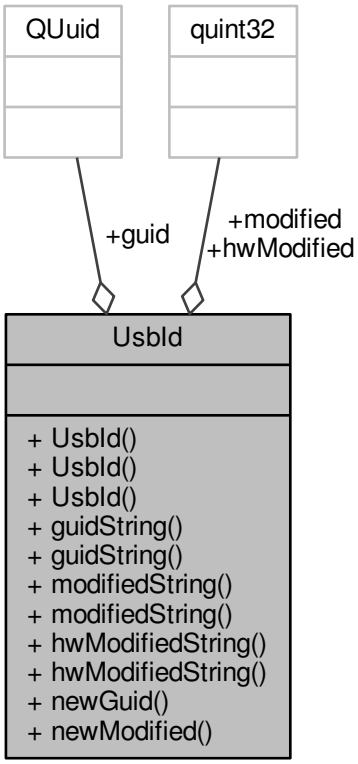
The documentation for this class was generated from the following file:

- src/ckb/ui_settingswidget.h

12.102 UsbId Struct Reference

`#include <src/ckb/kbmode.h>`

Collaboration diagram for UsbId:



Public Member Functions

- UsbId (const QString &_guid, quint32 _modified)
- UsbId (const QString &_guid, const QString &_modified)
- UsbId ()
- QString guidString () const
- void guidString (const QString &newGuid)
- QString modifiedString () const
- void modifiedString (const QString &newModified)
- QString hwModifiedString () const
- void hwModifiedString (const QString &newModified)
- void newGuid ()
- void newModified ()

Data Fields

- QUuid guid
- quint32 modified
- quint32 hwModified

12.102.1 Detailed Description

Definition at line 11 of file kbmode.h.

12.102.2 Constructor & Destructor Documentation

12.102.2.1 UsbId::UsbId (const QString & *_guid,* quint32 *_modified*) [inline]

Definition at line 16 of file kbmode.h.

```
16 : guid(_guid), modified(_modified), hwModified(_modified) {}
```

12.102.2.2 UsbId::UsbId (const QString & *_guid,* const QString & *_modified*) [inline]

Definition at line 17 of file kbmode.h.

```
17 : guid(_guid), modified(_modified.toUInt(0, 16)), hwModified(
      modified) {}
```

12.102.2.3 UsbId::UsbId ( ) [inline]

Definition at line 18 of file kbmode.h.

```
18 : guid(QUuid::createUuid()),modified(0) {}
```

12.102.3 Member Function Documentation

12.102.3.1 QString UsbId::guidString ( ) const [inline]

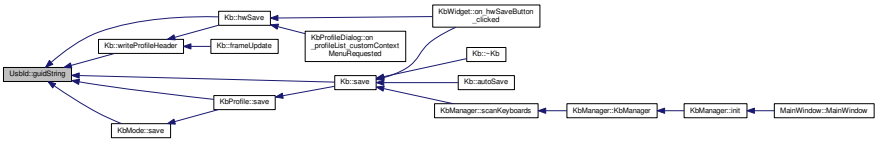
Definition at line 20 of file kbmode.h.

References guid.

Referenced by Kb::hwSave(), KbProfile::save(), KbMode::save(), Kb::save(), and Kb::writeProfileHeader().

```
20 { return guid.toString().toUpper(); }
```

Here is the caller graph for this function:



12.102.3.2 void UsbId::guidString (const QString & *newGuid*) [inline]

Definition at line 21 of file kbmode.h.

References guid, and newGuid().

```
21 { guid = newGuid; }
```

Here is the call graph for this function:

12.102.3.3 QString UsbId::hwModifiedString ( ) const [inline]

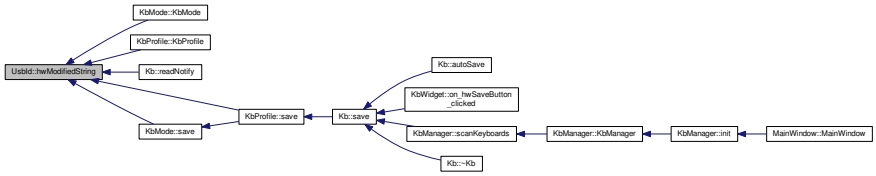
Definition at line 24 of file kbmode.h.

References hwModified.

Referenced by KbMode::KbMode(), KbProfile::KbProfile(), Kb::readNotify(), KbProfile::save(), and KbMode::save().

```
24 { return QString::number(hwModified, 16); }
```

Here is the caller graph for this function:



12.102.3.4 void UsbId::hwModifiedString (const QString & *newModified*) [inline]

Definition at line 25 of file kbmode.h.

References hwModified.

```
25 { hwModified = newModified.toUInt(0, 16); }
```

12.102.3.5 QString UsbId::modifiedString ( ) const [inline]

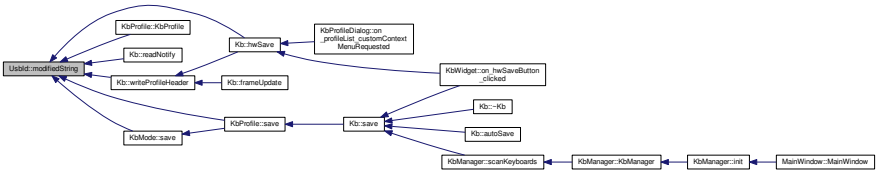
Definition at line 22 of file kbmode.h.

References modified.

Referenced by Kb::hwSave(), KbProfile::KbProfile(), Kb::readNotify(), KbProfile::save(), KbMode::save(), and Kb::writeProfileHeader().

```
22 { return QString::number(modified, 16); }
```

Here is the caller graph for this function:



12.102.3.6 void UsbId::modifiedString (const QString & *newModified*) [inline]

Definition at line 23 of file kbmode.h.

References modified.

```
23 { modified = newModified.toUInt(0, 16); }
```

12.102.3.7 void UsbId::newGuid ( ) [inline]

Definition at line 28 of file kbmode.h.

References guid.

Referenced by guidString().

```
28 { guid = QUuid::createUuid(); }
```

Here is the caller graph for this function:

12.102.3.8 void UsbId::newModified ( ) [inline]

Definition at line 29 of file kbmode.h.

References modified.

Referenced by KbProfile::save(), and KbMode::save().

```
29 { quint32 newMod; do { newMod = qrand(); } while(newMod == modified);
      modified = newMod; }
```

Here is the caller graph for this function:

12.102.4 Field Documentation

12.102.4.1 QUuid UsbId::guid

Definition at line 12 of file kbmode.h.

Referenced by KbProfile::find(), Kb::find(), guidString(), KbMode::KbMode(), KbProfile::KbProfile(), newGuid(), KbWidget::on_modesList_customContextMenuRequested(), KbWidget::on_modesList_itemChanged(), KbWidget::on_modesList_itemClicked(), KbProfileDialog::on_profileList_customContextMenuRequested(), KbProfileDialog::on_profileList_itemChanged(), KbProfileDialog::on_profileList_itemClicked(), KbWidget::profileChanged(), Kb::readNotify(), and KbProfileDialog::repopulate().

12.102.4.2 quint32 UsbId::hwModified

Definition at line 14 of file kbmode.h.

Referenced by hwModifiedString(), Kb::hwSave(), KbMode::KbMode(), and KbProfile::KbProfile().

12.102.4.3 quint32 UsbId::modified

Definition at line 13 of file kbmode.h.

Referenced by Kb::hwSave(), KbMode::KbMode(), KbProfile::KbProfile(), modifiedString(), and newModified().

The documentation for this struct was generated from the following file:

- src/ckb/kbmode.h

Chapter 13

File Documentation

13.1 BUILD.md File Reference

13.2 DAEMON.md File Reference

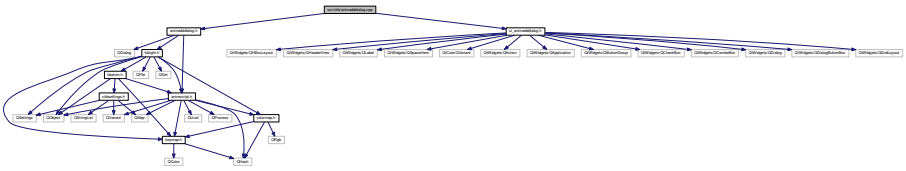
13.3 README.md File Reference

13.4 ROADMAP.md File Reference

13.5 src/ckb/animadddialog.cpp File Reference

```
#include "animadddialog.h"
#include "ui_animadddialog.h"
```

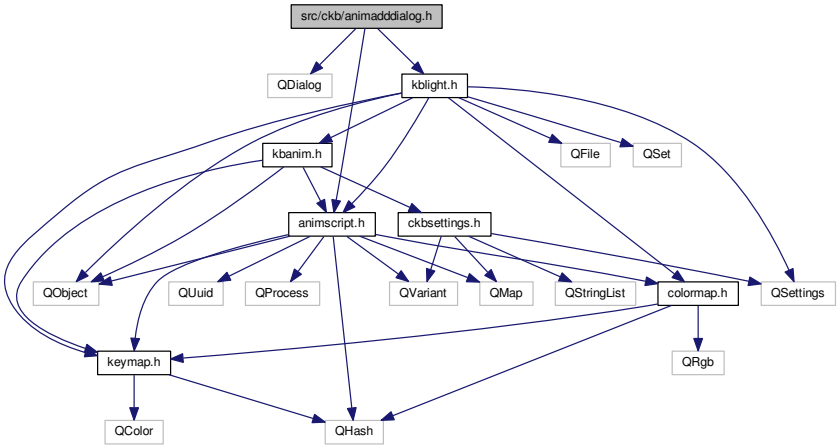
Include dependency graph for animadddialog.cpp:



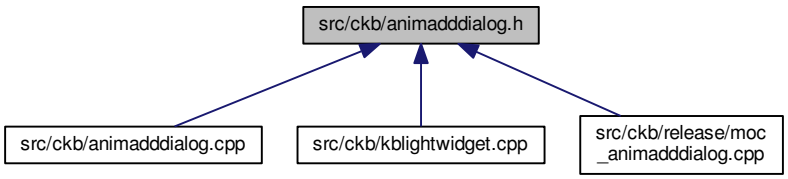
13.6 src/ckb/animadddialog.h File Reference

```
#include <QDialog>
#include "animscript.h"
#include "kblight.h"
```

Include dependency graph for animadddialog.h:



This graph shows which files directly or indirectly include this file:



Data Structures

- class AnimAddDialog

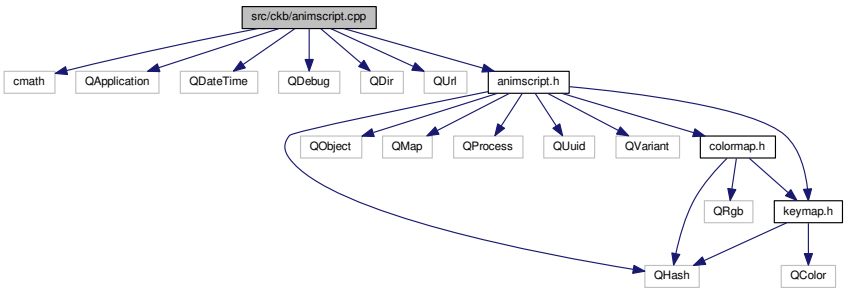
Namespaces

- Ui

13.7 src/ckb/animscript.cpp File Reference

```
#include <cmath>
#include <QApplication>
#include <QDateTime>
#include <QDebug>
#include <QDir>
#include <QUrl>
#include "animscript.h"
```

Include dependency graph for animscript.cpp:



Functions

- QString urlParam (const QString &param)

Variables

- static const double ONE_DAY = 24. ∗ 60. ∗ 60.

13.7.1 Function Documentation

13.7.1.1 QString urlParam (const QString & *param*) [inline]

Definition at line 71 of file animscript.cpp.

Referenced by AnimScript::load().

```
71                                                {
72     return QUrl::fromPercentEncoding(param.trimmed().toLatin1()).trimmed();
73 }
```

Here is the caller graph for this function:

13.7.2 Variable Documentation

13.7.2.1 const double ONE_DAY = 24. ∗ 60. ∗ 60. [static]

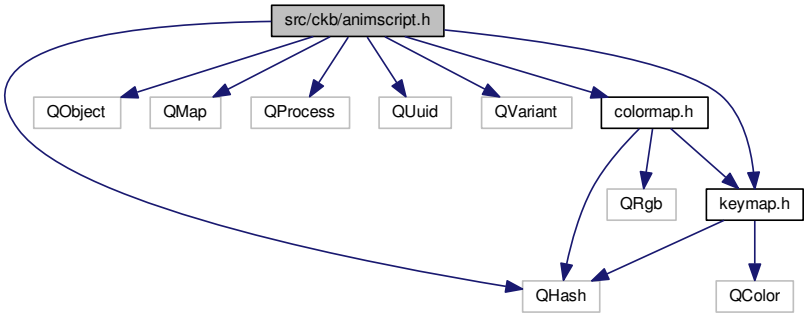
Definition at line 75 of file animscript.cpp.

Referenced by AnimScript::load().

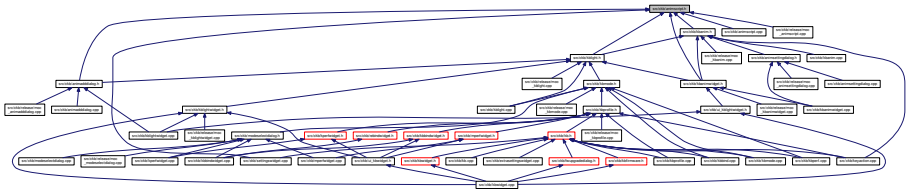
13.8 src/ckb/animscript.h File Reference

```
#include <QHash>
#include <QObject>
#include <QMap>
#include <QProcess>
#include <QUuid>
#include <QVariant>
#include "keymap.h"
#include "colormap.h"
```

Include dependency graph for animscript.h:



This graph shows which files directly or indirectly include this file:



Data Structures

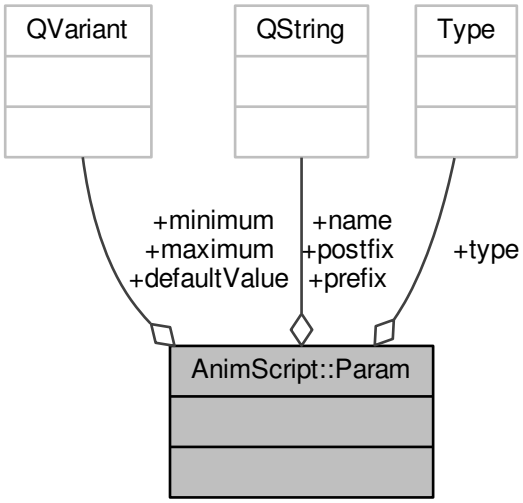
- class AnimScript
- struct AnimScript::Param

13.8.1 Data Structure Documentation

13.8.1.1 struct AnimScript::Param

Definition at line 21 of file animscript.h.

Collaboration diagram for AnimScript::Param:



Data Fields

	Type	

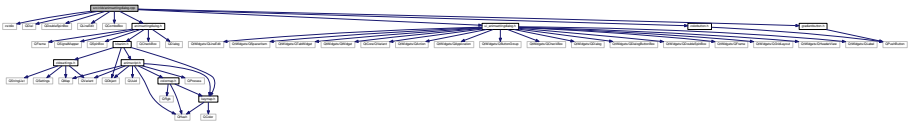
Data Fields

QVariant	defaultValue	
QVariant	maximum	
QVariant	minimum	
QString	name	
QString	postfix	
QString	prefix	
Type	type	

13.9 src/ckb/animsettingdialog.cpp File Reference

```
#include <cstdio>
#include <QDial>
#include <QDoubleSpinBox>
#include <QLineEdit>
#include <QComboBox>
#include "animsettingdialog.h"
#include "ui_animsettingdialog.h"
#include "colorbutton.h"
#include "gradientbutton.h"
```

Include dependency graph for animsettingdialog.cpp:



Functions

- static int angleFlip (int angle)

13.9.1 Function Documentation

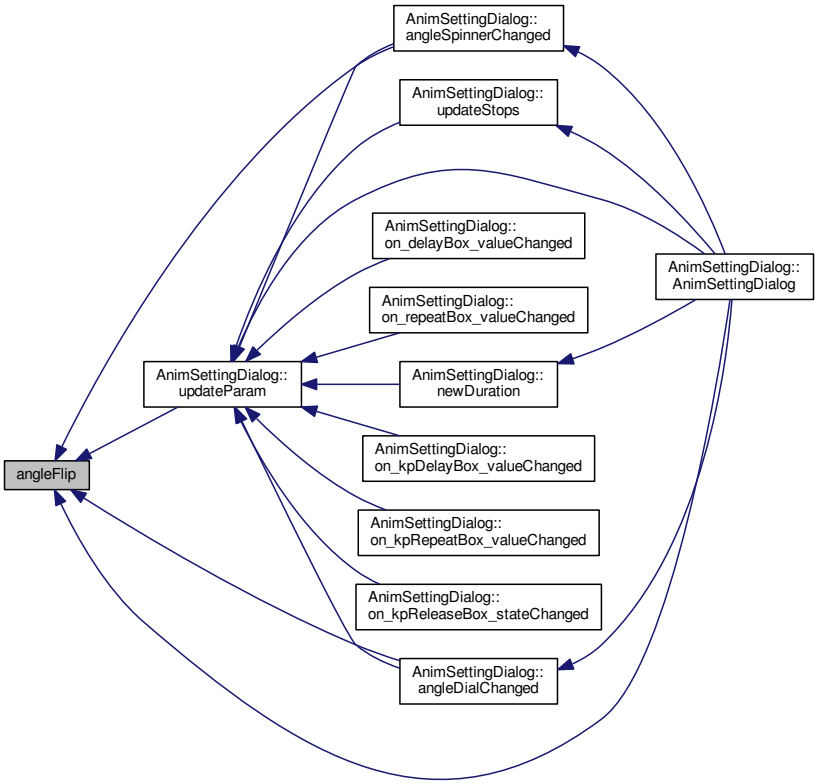
13.9.1.1 static int angleFlip (int *angle*) [inline], [static]

Definition at line 12 of file animsettingdialog.cpp.

Referenced by AnimSettingDialog::angleDialChanged(), AnimSettingDialog::angleSpinnerChanged(), AnimSettingDialog::AnimSettingDialog(), and AnimSettingDialog::updateParam().

```
12                                      {
13     if(angle <= 180)
14         return 180 - angle;
15     else
16         return 360 - (angle - 180);
17 }
```

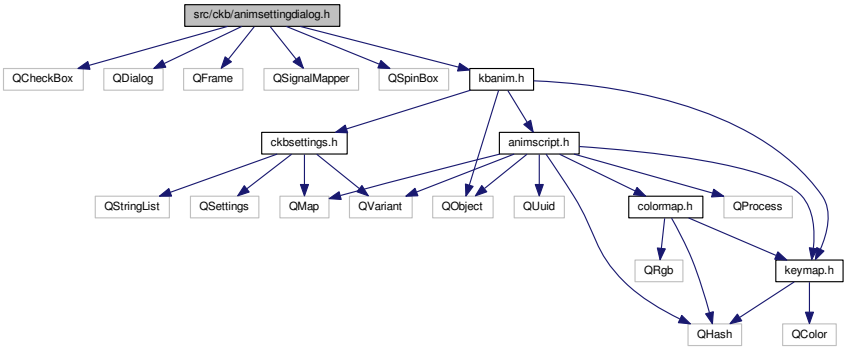
Here is the caller graph for this function:



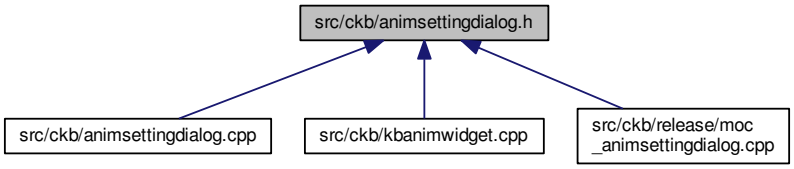
13.10 src/ckb/animsettingdialog.h File Reference

```
#include <QCheckBox>
#include <QDialog>
#include <QFrame>
#include <QSignalMapper>
#include <QSpinBox>
#include "kbanim.h"
```

Include dependency graph for animsettingdialog.h:



This graph shows which files directly or indirectly include this file:



Data Structures

- class AnimSettingDialog

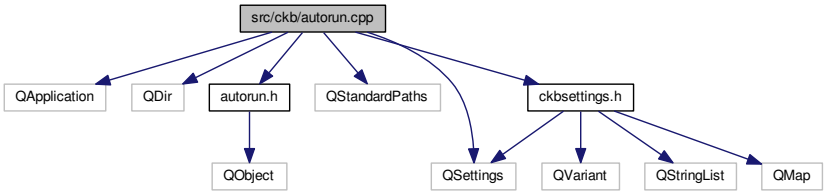
Namespaces

- Ui

13.11 src/ckb/autorun.cpp File Reference

```
#include <QApplication>
#include <QDir>
#include <QSettings>
#include <QStandardPaths>
#include "autorun.h"
#include "ckbsettings.h"
```

Include dependency graph for autorun.cpp:



Variables

- static const QString settingPath = "Program/DidLoginItem"

13.11.1 Variable Documentation

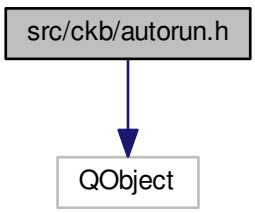
13.11.1.1 const QString settingPath = "Program/DidLoginItem" [static]

Definition at line 18 of file autorun.cpp.

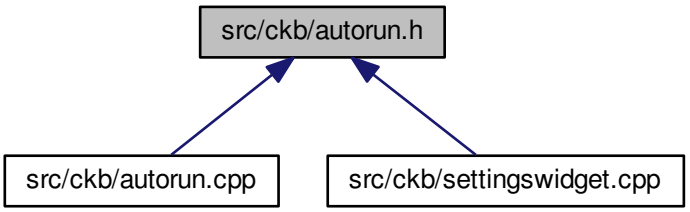
13.12 src/ckb/autorun.h File Reference

```
#include <QObject>
```

Include dependency graph for autorun.h:



This graph shows which files directly or indirectly include this file:



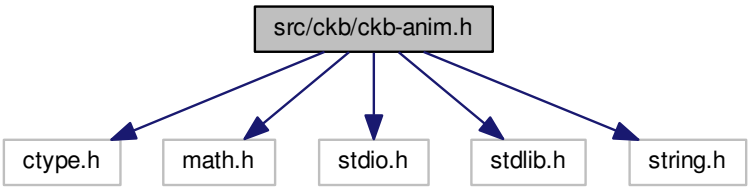
Data Structures

- class AutoRun

13.13 src/ckb/ckb-anim.h File Reference

```
#include <ctype.h>
#include <math.h>
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
```

Include dependency graph for ckb-anim.h:



Data Structures

- struct ckb_key
- struct ckb_runctx
- struct ckb_gradient

Macros

- #define M_PI 3.14159265358979323846
- #define TRUE 1

- #define FALSE 0
- #define CKB_DOWN 1
- #define CKB_UP 0
- #define CKB_CONTAINER(macro) do { macro } while(0)
- #define CKB_GUID(guid) CKB_CONTAINER(printf("guid "); printurl(guid); printf("\n");)
- #define CKB_NAME(name) CKB_CONTAINER(printf("name "); printurl(name); printf("\n");)
- #define CKB_VERSION(version) CKB_CONTAINER(printf("version "); printurl(version); printf("\n");)
- #define CKB_COPYRIGHT(year, author) CKB_CONTAINER(printf("author "); printurl(author); printf("\nyear %s\n", year);)
- #define CKB_LICENSE(license) CKB_CONTAINER(printf("license "); printurl(license); printf("\n");)
- #define CKB_DESCRIPTION(description) CKB_CONTAINER(printf("description "); printurl(description); printf("\n");)
- #define CKB_PARAM(type, name, prefix, postfix, extra) CKB_CONTAINER(printf("param %s %s ", type, name); printurl(prefix); printf(" "); printurl(postfix); printf(" "); extra; printf("\n");)
- #define CKB_PARAM_LONG(name, prefix, postfix, default, min, max) CKB_PARAM("long", name, prefix, postfix, printf("%ld %ld %ld", (long)(default), (long)(min), (long)(max)))
- #define CKB_PARAM_DOUBLE(name, prefix, postfix, default, min, max) CKB_PARAM("double", name, prefix, postfix, printf("%lf %lf %lf", (double)(default), (double)(min), (double)(max)))
- #define CKB_PARAM_BOOL(name, text, default) CKB_PARAM("bool", name, text, "", printf((default) ? "1" : "0"))
- #define CKB_PARAM_RGB(name, prefix, postfix, r, g, b) CKB_PARAM("rgb", name, prefix, postfix, printf("%02x%02x%02x", (unsigned char)(r), (unsigned char)(g), (unsigned char)(b)))
- #define CKB_PARAM_ARGB(name, prefix, postfix, a, r, g, b) CKB_PARAM("argb", name, prefix, postfix, printf("%02x%02x%02x%02x", (unsigned char)(a), (unsigned char)(r), (unsigned char)(g), (unsigned char)(b)))
- #define CKB_PARAM_GRADIENT(name, prefix, postfix, default) CKB_PARAM("gradient", name, prefix, postfix, printurl(default))
- #define CKB_PARAM_AGRADIENT(name, prefix, postfix, default) CKB_PARAM("agradient", name, prefix, postfix, printurl(default))
- #define CKB_PARAM_ANGLE(name, prefix, postfix, default) CKB_PARAM("angle", name, prefix, postfix, printf("%ld", (long)(default)))
- #define CKB_PARAM_STRING(name, prefix, postfix, default) CKB_PARAM("string", name, prefix, postfix, printurl(default))
- #define CKB_PARAM_LABEL(name, text) CKB_PARAM("label", name, text, "",)
- #define CKB_PRESET_START(name) CKB_CONTAINER(printf("preset "); printurl(name);)
- #define CKB_PRESET_PARAM(name, value) CKB_CONTAINER(printf(" %s=", name); printurl(value);)
- #define CKB_PRESET_END CKB_CONTAINER(printf("\n");)
- #define CKB_KP_NONE "none"
- #define CKB_KP_NAME "name"
- #define CKB_KP_POSITION "position"
- #define CKB_KPMODE(mode) CKB_CONTAINER(printf("kpmode %s\n", mode);)
- #define CKB_TIME_DURATION "duration"
- #define CKB_TIME_ABSOLUTE "absolute"
- #define CKB_TIMEMODE(mode) CKB_CONTAINER(printf("time %s\n", mode);)
- #define CKB_REPEAT(enable) CKB_CONTAINER(printf("repeat %s\n", (enable) ? "on" : "off");)
- #define CKB_PREEMPT(enable) CKB_CONTAINER(printf("preempt %s\n", (enable) ? "on" : "off");)
- #define CKB_LIVEPARAMS(enable) CKB_CONTAINER(printf("parammode %s\n", (enable) ? "live" : "static");)
- #define CKB_PARSE_LONG(param_name, value_ptr) if(!strcmp(name, param_name) && sscanf(value, "%ld", value_ptr) == 1)
- #define CKB_PARSE_DOUBLE(param_name, value_ptr) if(!strcmp(name, param_name) && sscanf(value, "%lf", value_ptr) == 1)
- #define CKB_PARSE_BOOL(param_name, value_ptr) if(!strcmp(name, param_name) && sscanf(value, "%u", value_ptr) == 1)
- #define CKB_PARSE_RGB(param_name, r_ptr, g_ptr, b_ptr) if(!strcmp(name, param_name) && sscanf(value, "%2hhx%2hhx%2hhx", r_ptr, g_ptr, b_ptr) == 3)

- #define CKB_PARSE_ARGB(param_name, a_ptr, r_ptr, g_ptr, b_ptr) if(!strcmp(name, param_name) && sscanf(value, "%2hhx%2hhx%2hhx%2hhx", a_ptr, r_ptr, g_ptr, b_ptr) == 4)
- #define CKB_PARSE_GRADIENT(param_name, gradient_ptr) if(!strcmp(name, param_name) && ckb_scan_grad(value, gradient_ptr, 0))
- #define CKB_PARSE_AGRADIENT(param_name, gradient_ptr) if(!strcmp(name, param_name) && ckb_scan_grad(value, gradient_ptr, 1))
- #define CKB_PARSE_ANGLE(param_name, value_ptr) if(!strcmp(name, param_name) && sscanf(value, "%ld", value_ptr) == 1)
- #define CKB_PARSE_STRING(param_name) if(!strcmp(name, param_name))
- #define CKB_REAL_ANGLE(angle) fmod((-(angle) + 90.) * M_PI / 180. + M_PI * 2., M_PI * 2.)
- #define CKB_KEYNAME_MAX 12
- #define CKB_KEYCLEAR(context) CKB_CONTAINER(ckb_key* key = context->keys; unsigned count = context->keycount; unsigned i = 0; for(; i < count; i++) key[i].a = key[i].r = key[i].g = key[i].b = 0;)
- #define CKB_GRAD_MAX 100
- #define CKB_MAX_WORD (4 * 1024)

Functions

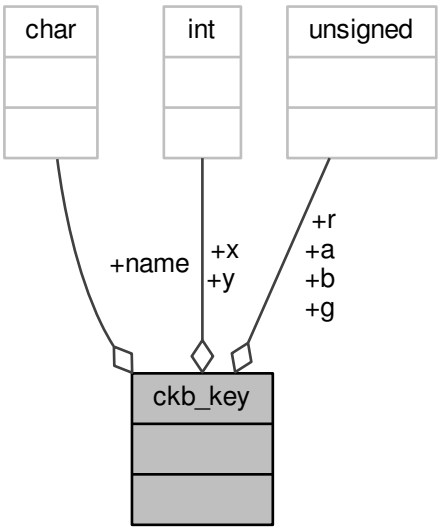
- int ckb_scan_grad (const char *string, ckb_gradient *gradient, int alpha)
- void ckb_grad_color (float *a, float *r, float *g, float *b, const ckb_gradient *grad, float pos)
- void ckb_alpha_blend (ckb_key *key, float a, float r, float g, float b)
- void printurl (const char *src)
- void urldecode (char *dst, const char *src)
- void ckb_getline (char word1[(4 *1024)], char word2[(4 *1024)], char word3[(4 *1024)])
- void ckb_info ()
- void ckb_init (ckb_runctx *context)
- void ckb_parameter (ckb_runctx *, const char *, const char *)
- void ckb_keypress (ckb_runctx *, ckb_key *, int, int, int)
- void ckb_start (ckb_runctx *, int)
- void ckb_time (ckb_runctx *, double)
- int ckb_frame (ckb_runctx *)
- void ckb_read_params (ckb_runctx *ctx)
- int main (int argc, char *argv[])

13.13.1 Data Structure Documentation

13.13.1.1 struct ckb_key

Definition at line 131 of file ckb-anim.h.

Collaboration diagram for ckb_key:



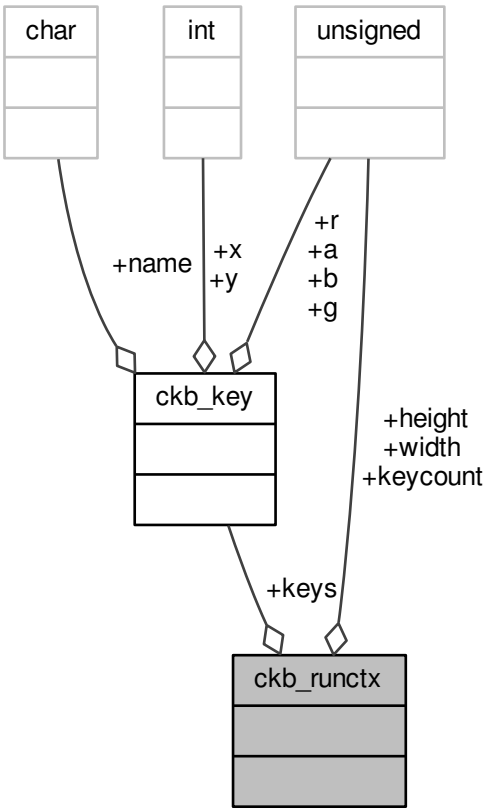
Data Fields

unsigned char	a	
unsigned char	b	
unsigned char	g	
char	name[12]	
unsigned char	r	
int	x	
int	y	

13.13.1.2 struct ckb_runctx

Definition at line 138 of file ckb-anim.h.

Collaboration diagram for ckb_runctx:



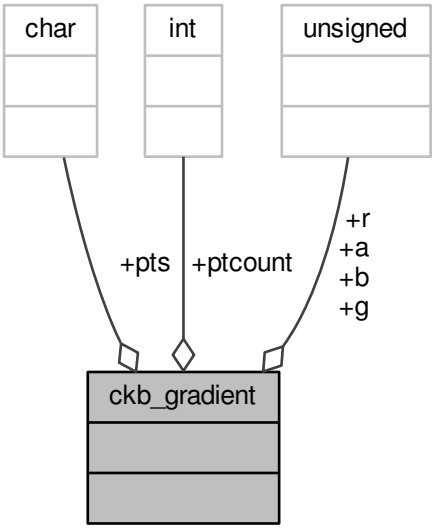
Data Fields

unsigned	height	
unsigned	keycount	
ckb_key *	keys	
unsigned	width	

13.13.1.3 struct ckb_gradient

Definition at line 152 of file ckb-anim.h.

Collaboration diagram for ckb_gradient:



Data Fields

unsigned char	a[100]	
unsigned char	b[100]	
unsigned char	g[100]	
int	ptcount	
char	pts[100]	
unsigned char	r[100]	

13.13.2 Macro Definition Documentation

13.13.2.1 #define CKB_CONTAINER(*macro*) do { macro } while(0)

Definition at line 57 of file ckb-anim.h.

13.13.2.2 #define CKB_COPYRIGHT(*year, author*) CKB_CONTAINER(printf("author "); printurl(author); printf("\nyear %s\n", year);)

Definition at line 66 of file ckb-anim.h.

13.13.2.3 #define CKB_DESCRIPTION(*description*) CKB_CONTAINER(printf("description "); printurl(description); printf("\n");)

Definition at line 70 of file ckb-anim.h.

13.13.2.4 #define CKB_DOWN 1

Definition at line 52 of file ckb-anim.h.

13.13.2.5 #define CKB_GRAD_MAX 100

Definition at line 151 of file ckb-anim.h.

13.13.2.6 #define CKB_GUID(*guid*) CKB_CONTAINER(printf("guid "); printurl(guid); printf("\n");)

Definition at line 60 of file ckb-anim.h.

13.13.2.7 #define CKB_KEYCLEAR(*context*) CKB_CONTAINER(ckb_key* key = context->keys; unsigned count = context->keycount; unsigned i = 0; for(; i < count; i++) key[i].a = key[i].r = key[i].g = key[i].b = 0;)

Definition at line 148 of file ckb-anim.h.

13.13.2.8 #define CKB_KEYNAME_MAX 12

Definition at line 130 of file ckb-anim.h.

Referenced by main().

13.13.2.9 #define CKB_KP_NAME "name"

Definition at line 91 of file ckb-anim.h.

13.13.2.10 #define CKB_KP_NONE "none"

Definition at line 90 of file ckb-anim.h.

13.13.2.11 #define CKB_KP_POSITION "position"

Definition at line 92 of file ckb-anim.h.

13.13.2.12 #define CKB_KPMODE(*mode*) CKB_CONTAINER(printf("kpmode %s\n", mode);)

Definition at line 93 of file ckb-anim.h.

13.13.2.13 #define CKB_LICENSE(*license*) CKB_CONTAINER(printf("license "); printurl(license); printf("\n");)

Definition at line 68 of file ckb-anim.h.

13.13.2.14 #define CKB_LIVEPARAMS(*enable*) CKB_CONTAINER(printf("parammode %s\n", (enable) ? "live" : "static");)

Definition at line 104 of file ckb-anim.h.

13.13.2.15 #define CKB_MAX_WORD (4 * 1024)

Definition at line 238 of file ckb-anim.h.

Referenced by ckb_read_params(), and main().

13.13.2.16 #define CKB_NAME(*name*) CKB_CONTAINER(printf("name "); printurl(name); printf("\n");)

Definition at line 62 of file ckb-anim.h.

13.13.2.17 #define CKB_PARAM(*type, name, prefix, postfix, extra*) CKB_CONTAINER(printf("param %s %s ", type, name); printurl(prefix); printf(" "); printurl(postfix); printf(" "); extra; printf("\n");)

Definition at line 73 of file ckb-anim.h.

13.13.2.18 #define CKB_PARAM_AGRADIENT(*name, prefix, postfix, default*) CKB_PARAM("agradient", name, prefix, postfix, printurl(default))

Definition at line 80 of file ckb-anim.h.

13.13.2.19 #define CKB_PARAM_ANGLE(*name, prefix, postfix, default*) CKB_PARAM("angle", name, prefix, postfix, printf("%ld", (long)(default)))

Definition at line 81 of file ckb-anim.h.

13.13.2.20 #define CKB_PARAM_ARGB(*name, prefix, postfix, a, r, g, b*) CKB_PARAM("argb", name, prefix, postfix, printf("%02x%02x%02x%02x", (unsigned char)(a), (unsigned char)(r), (unsigned char)(g), (unsigned char)(b)))

Definition at line 78 of file ckb-anim.h.

13.13.2.21 #define CKB_PARAM_BOOL(*name, text, default*) CKB_PARAM("bool", name, text, "", printf((default) ? "1" : "0"))

Definition at line 76 of file ckb-anim.h.

13.13.2.22 #define CKB_PARAM_DOUBLE(*name, prefix, postfix, default, min, max*) CKB_PARAM("double", name, prefix, postfix, printf("%lf %lf %lf", (double)(default), (double)(min), (double)(max)))

Definition at line 75 of file ckb-anim.h.

13.13.2.23 #define CKB_PARAM_GRADIENT(*name, prefix, postfix, default*) CKB_PARAM("gradient", name, prefix, postfix, printurl(default))

Definition at line 79 of file ckb-anim.h.

13.13.2.24 #define CKB_PARAM_LABEL(*name, text*) CKB_PARAM("label", name, text, "",)

Definition at line 83 of file ckb-anim.h.

13.13.2.25 #define CKB_PARAM_LONG(*name, prefix, postfix, default, min, max*) CKB_PARAM("long", name, prefix, postfix, printf("%ld %ld %ld", (long)(default), (long)(min), (long)(max)))

Definition at line 74 of file ckb-anim.h.

13.13.2.26 #define CKB_PARAM_RGB(*name, prefix, postfix, r, g, b*) **CKB_PARAM**("rgb", name, prefix, postfix, printf("%02x%02x%02x", (unsigned char)(r), (unsigned char)(g), (unsigned char)(b)))

Definition at line 77 of file ckb-anim.h.

13.13.2.27 #define CKB_PARAM_STRING(*name, prefix, postfix, default*) **CKB_PARAM**("string", name, prefix, postfix, **printurl**(default))

Definition at line 82 of file ckb-anim.h.

13.13.2.28 #define CKB_PARSE_AGRADIENT(*param_name, gradient_ptr*) if(!strcmp(name, param_name) && **ckb_scan_grad**(value, gradient_ptr, 1))

Definition at line 121 of file ckb-anim.h.

13.13.2.29 #define CKB_PARSE_ANGLE(*param_name, value_ptr*) if(!strcmp(name, param_name) && sscanf(value, "%ld", value_ptr) == 1)

Definition at line 122 of file ckb-anim.h.

13.13.2.30 #define CKB_PARSE_ARGB(*param_name, a_ptr, r_ptr, g_ptr, b_ptr*) if(!strcmp(name, param_name) && sscanf(value, "%2hhx%2hhx%2hhx%2hhx", a_ptr, r_ptr, g_ptr, b_ptr) == 4)

Definition at line 119 of file ckb-anim.h.

13.13.2.31 #define CKB_PARSE_BOOL(*param_name, value_ptr*) if(!strcmp(name, param_name) && sscanf(value, "%u", value_ptr) == 1)

Definition at line 117 of file ckb-anim.h.

13.13.2.32 #define CKB_PARSE_DOUBLE(*param_name, value_ptr*) if(!strcmp(name, param_name) && sscanf(value, "%lf", value_ptr) == 1)

Definition at line 116 of file ckb-anim.h.

13.13.2.33 #define CKB_PARSE_GRADIENT(*param_name, gradient_ptr*) if(!strcmp(name, param_name) && **ckb_scan_grad**(value, gradient_ptr, 0))

Definition at line 120 of file ckb-anim.h.

13.13.2.34 #define CKB_PARSE_LONG(*param_name, value_ptr*) if(!strcmp(name, param_name) && sscanf(value, "%ld", value_ptr) == 1)

Definition at line 115 of file ckb-anim.h.

13.13.2.35 #define CKB_PARSE_RGB(*param_name, r_ptr, g_ptr, b_ptr*) if(!strcmp(name, param_name) && sscanf(value, "%2hhx%2hhx%2hhx", r_ptr, g_ptr, b_ptr) == 3)

Definition at line 118 of file ckb-anim.h.

13.13.2.36 #define CKB_PARSE_STRING(*param_name*) if(!strcmp(name, param_name))

Definition at line 123 of file ckb-anim.h.

13.13.2.37 #define CKB_PREEMPT(*enable*) CKB_CONTAINER(printf("preempt %s\n", (enable) ? "on" : "off");)

Definition at line 102 of file ckb-anim.h.

13.13.2.38 #define CKB_PRESET_END CKB_CONTAINER(printf("\n");)

Definition at line 87 of file ckb-anim.h.

13.13.2.39 #define CKB_PRESET_PARAM(*name, value*) CKB_CONTAINER(printf(" %s=", name); printurl(value);)

Definition at line 86 of file ckb-anim.h.

13.13.2.40 #define CKB_PRESET_START(*name*) CKB_CONTAINER(printf("preset "); printurl(name);)

Definition at line 85 of file ckb-anim.h.

13.13.2.41 #define CKB_REAL_ANGLE(*angle*) fmod((-(angle) + 90.) * M_PI / 180. + M_PI * 2., M_PI * 2.)

Definition at line 127 of file ckb-anim.h.

13.13.2.42 #define CKB_REPEAT(*enable*) CKB_CONTAINER(printf("repeat %s\n", (enable) ? "on" : "off");)

Definition at line 99 of file ckb-anim.h.

13.13.2.43 #define CKB_TIME_ABSOLUTE "absolute"

Definition at line 96 of file ckb-anim.h.

13.13.2.44 #define CKB_TIME_DURATION "duration"

Definition at line 95 of file ckb-anim.h.

13.13.2.45 #define CKB_TIMEMODE(*mode*) CKB_CONTAINER(printf("time %s\n", mode);)

Definition at line 97 of file ckb-anim.h.

13.13.2.46 #define CKB_UP 0

Definition at line 53 of file ckb-anim.h.

13.13.2.47 #define CKB_VERSION(*version*) CKB_CONTAINER(printf("version "); printurl(version); printf("\n");)

Definition at line 64 of file ckb-anim.h.

13.13.2.48 #define FALSE 0

Definition at line 49 of file ckb-anim.h.

13.13.2.49 #define M_PI 3.14159265358979323846

Definition at line 42 of file ckb-anim.h.

13.13.2.50 #define TRUE 1

Definition at line 46 of file ckb-anim.h.

13.13.3 Function Documentation

13.13.3.1 void ckb_alpha_blend (ckb_key * key, float a, float r, float g, float b)

Definition at line 283 of file ckb-anim.h.

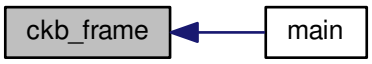
References ckb_key::a, ckb_key::b, ckb_key::g, and ckb_key::r.

```
283                                                                    {
284     a /= 255.f;
285     float ka = key->a / 255.f;
286     float a2 = a + (1.f - a) * ka;
287     if(a2 == 0.f){
288         key->a = key->r = key->g = key->b = 0;
289         return;
290     }
291     key->a = round(a2 * 255.f);
292     key->r = round((r * a + key->r * ka * (1.f - a)) / a2);
293     key->g = round((g * a + key->g * ka * (1.f - a)) / a2);
294     key->b = round((b * a + key->b * ka * (1.f - a)) / a2);
295 }
```

13.13.3.2 int ckb_frame (ckb_runctx *)

Referenced by main().

Here is the caller graph for this function:



13.13.3.3 void ckb_getline (char word1[(4 * 1024)], char word2[(4 * 1024)], char word3[(4 * 1024)])

Definition at line 239 of file ckb-anim.h.

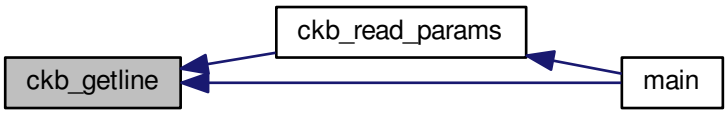
References urldecode().

Referenced by ckb_read_params(), and main().

```
239                                                                                                 {
240     char line[CKB_MAX_WORD * 3 + 3];
241     fgets(line, sizeof(line), stdin);
242     word1[0] = word1[1] = word1[2] = 0;
243     int res = sscanf(line, "%s %s %s", word1, word2, word3);
244     if(res >= 1)
245         urldecode(word1, word1);
246     if(res >= 2)
247         urldecode(word2, word2);
248     if(res >= 3)
249         urldecode(word3, word3);
250     line[strlen(line) - 1] = 0;
251 }
```

Here is the call graph for this function:

Here is the caller graph for this function:



13.13.3.4 void ckb_grad_color (float * *a*, float * *r*, float * *g*, float * *b*, const ckb_gradient * *grad*, float *pos*)

Definition at line 254 of file ckb-anim.h.

References ckb_gradient::a, ckb_gradient::b, ckb_gradient::g, ckb_gradient::ptcount, ckb_gradient::pts, and ckb_-gradient::r.

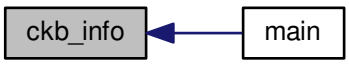
```
254                                                                                                 {
255     // Find the points surrounding this position
256     int count = grad->ptcount;
257     if(count == 0){
258         *a = *r = *g = *b = 0.f;
259         return;
260     }
261     int i = 1;
262     for(; i < count; i++){
263         if(grad->pts[i] >= pos)
264             break;
265     }
266     // Get color by linear interpolation. Premultiply the alpha value so that it returns the expected color
267     // (i.e. stops with zero opacity won't contribute to color)
268     float distance = grad->pts[i] - grad->pts[i - 1];
269     float dx = (pos - grad->pts[i - 1]) / distance;
270     float a1 = grad->a[i] / 255., a2 = grad->a[i - 1] / 255.;
271     float a3 = *a = a1 * dx + a2 * (1.f - dx);
272     if(a3 == 0.){
```

```
273         *a = *r = *g = *b = 0.f;
274         return;
275     }
276     *a *= 255.f;
277     *r = (grad->r[i] * a1 * dx + grad->r[i - 1] * a2 * (1.f - dx)) / a3;
278     *g = (grad->g[i] * a1 * dx + grad->g[i - 1] * a2 * (1.f - dx)) / a3;
279     *b = (grad->b[i] * a1 * dx + grad->b[i - 1] * a2 * (1.f - dx)) / a3;
280 }
```

13.13.3.5 void ckb_info ( )

Referenced by main().

Here is the caller graph for this function:



13.13.3.6 void ckb_init (ckb_runctx * *context*)

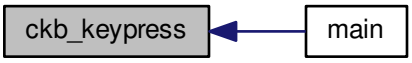
Referenced by main().

Here is the caller graph for this function:

13.13.3.7 void ckb_keypress (ckb_runctx *, ckb_key *, int , int , int)

Referenced by main().

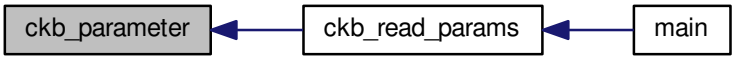
Here is the caller graph for this function:



13.13.3.8 void ckb_parameter (ckb_runctx * , const char * , const char *)

Referenced by ckb_read_params().

Here is the caller graph for this function:



13.13.3.9 void ckb_read_params (ckb_runctx * *ctx*)

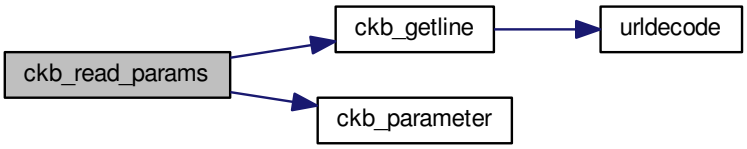
Definition at line 349 of file ckb-anim.h.

References ckb_getline(), CKB_MAX_WORD, and ckb_parameter().

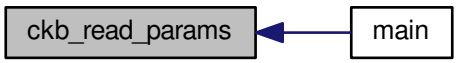
Referenced by main().

```
349                                         {
350     char cmd[CKB_MAX_WORD], param[CKB_MAX_WORD], value[
      CKB_MAX_WORD];
351     do {
352         ckb_getline(cmd, param, value);
353         if(!*cmd){
354             printf("Error [ckb-main]: Reached EOF reading parameters");
355             return;
356         }
357         if(!strcmp(cmd, "end") && !strcmp(param, "params"))
358             break;
359         if(strcmp(cmd, "param"))
360             continue;
361         ckb_parameter(ctx, param, value);
362     } while(1);
363 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



13.13.3.10 int ckb_scan_grad (const char * *string*, ckb_gradient * *gradient*, int *alpha*)

Definition at line 298 of file ckb-anim.h.

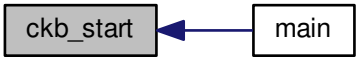
References ckb_gradient::a, ckb_gradient::b, ckb_gradient::g, ckb_gradient::ptcount, ckb_gradient::pts, and ckb_-gradient::r.

```
298                                                                         {
299     char pos = -1;
300     unsigned char a = 255, r, g, b;
301     int count = 0;
302     while(1){
303         int scanned = 0;
304         char newpos;
305         if(sscanf(string, "%hhd:%2hhx%2hhx%2hhx%2hhx%n", &newpos, &a, &r, &g, &b, &scanned) != 5)
306             break;
307         string += scanned;
308         // Don't allow stops out-of-order or past 100
309         if(newpos <= pos || newpos > 100)
310             return 0;
311         pos = newpos;
312         if(!alpha)
313             a = 255;
314         gradient->pts[count] = pos;
315         gradient->a[count] = a;
316         gradient->r[count] = r;
317         gradient->g[count] = g;
318         gradient->b[count] = b;
319         count++;
320     }
321     if(count == 0){
322         // If nothing was read, try a single ARGB constant.
323         if(sscanf(string, "%2hhx%2hhx%2hhx%2hhx", &a, &r, &g, &b) != 4)
324             return 0;
325         count = 2;
326         gradient->pts[0] = 0;
327         gradient->pts[1] = 100;
328         gradient->a[0] = a;
329         gradient->a[1] = 0;
330         gradient->r[0] = gradient->r[1] = r;
331         gradient->g[0] = gradient->g[1] = g;
332         gradient->b[0] = gradient->b[1] = b;
333     }
334     if(count < 2)
335         return 0;
336     gradient->ptcount = count;
337     return 1;
338 }
```

13.13.3.11 void ckb_start (ckb_runctx * , int)

Referenced by main().

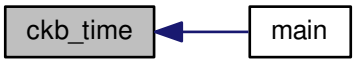
Here is the caller graph for this function:



13.13.3.12 void ckb_time (ckb_runctx *, double)

Referenced by main().

Here is the caller graph for this function:



13.13.3.13 int main (int *argc*, char * *argv[]*)

Definition at line 365 of file ckb-anim.h.

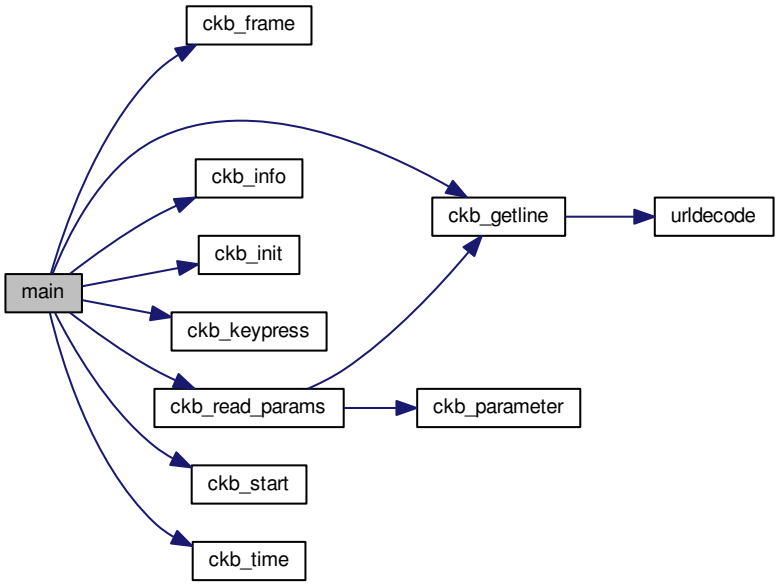
References ckb_key::a, ckb_key::b, ckb_frame(), ckb_getline(), ckb_info(), ckb_init(), CKB_KEYNAME_MAX, ckb_keypress(), CKB_MAX_WORD, ckb_read_params(), ckb_start(), ckb_time(), ckb_key::g, ckb_runctx::height, ckb_runctx::keycount, ckb_runctx::keys, ckb_key::name, ckb_key::r, ckb_runctx::width, ckb_key::x, and ckb_key::y.

```
365                                 {
366     if(argc == 2){
367         if(!strcmp(argv[1], "--ckb-info")){
368             ckb_info();
369             fflush(stdout);
370             return 0;
371         } else if(!strcmp(argv[1], "--ckb-run")){
372             ckb_runctx ctx;
373             // Read the keymap lines
374             char cmd[CKB_MAX_WORD], param[CKB_MAX_WORD], value[
    CKB_MAX_WORD];
375             // Skip anything up until "begin keymap"
376             do {
377                 ckb_getline(cmd, param, value);
378                 if(!*cmd){
379                     // If end-of-file is reached, abort
380                     printf("Error [ckb-main]: Reached EOF looking for \"begin keymap\"");
381                     return -2;
382                 }
383             } while(strcmp(cmd, "begin") || strcmp(param, "keymap"));
384             ckb_getline(cmd, param, value);
385             unsigned keycount;
386             if(strcmp(cmd, "keycount") || sscanf(param, "%u", &keycount) != 1 || keycount == 0){
387                 // If keycount isn't the next line, something is wrong
388                 printf("Error [ckb-main]: \"begin keymap\" not followed with \"keycount\"");
389                 return -3;
390             }
391             ctx.keys = (ckb_key*)calloc(keycount, sizeof(ckb_key));
```

```
392             ctx.keycount = keycount;
393             unsigned max_x = 0, max_y = 0;
394             unsigned i = 0;
395             for(; i < ctx.keycount; i++){
396                 ckb_getline(cmd, param, value);
397                 unsigned x, y;
398                 if(strcmp(cmd, "key") || !*param || !sscanf(value, "%u,%u", &x, &y)){
399                     i--;
400                     continue;
401                 }
402                 ckb_key* key = ctx.keys + i;
403                 strncpy(key->name, param, CKB_KEYNAME_MAX);
404                 key->x = x;
405                 key->y = y;
406                 if(x > max_x)
407                     max_x = x;
408                 if(y > max_y)
409                     max_y = y;
410             }
411             ctx.width = max_x + 1;
412             ctx.height = max_y + 1;
413             // Skip anything else until "end keymap"
414             do {
415                 ckb_getline(cmd, param, value);
416                 if(!*cmd){
417                     printf("Error [ckb-main]: Reached EOF looking for \"end keymap\"");
418                     return -2;
419                 }
420             } while(strcmp(cmd, "end") || strcmp(param, "keymap"));
421             // Run init function
422             ckb_init(&ctx);
423             // Skip anything else until "begin params"
424             do {
425                 ckb_getline(cmd, param, value);
426                 if(!*cmd){
427                     printf("Error [ckb-main]: Reached EOF looking for \"begin params\"");
428                     return -2;
429                 }
430             } while(strcmp(cmd, "begin") || strcmp(param, "params"));
431             // Parse parameters
432             ckb_read_params(&ctx);
433             // Skip anything else until "begin run"
434             do {
435                 ckb_getline(cmd, param, value);
436                 if(!*cmd){
437                     printf("Error [ckb-main]: Reached EOF looking for \"begin run\"");
438                     return -2;
439                 }
440             } while(strcmp(cmd, "begin") || strcmp(param, "run"));
441             // Run the main loop
442             printf("begin run\n");
443             fflush(stdout);
444             while(1){
445                 ckb_getline(cmd, param, value);
446                 if(!*cmd || (!strcmp(cmd, "end") && !strcmp(param, "run")))
447                     break;
448                 // Parse input
449                 if(!strcmp(cmd, "start"))
450                     ckb_start(&ctx, 1);
451                 else if(!strcmp(cmd, "stop"))
452                     ckb_start(&ctx, 0);
453                 else if(!strcmp(cmd, "begin") && !strcmp(param, "params"))
454                     ckb_read_params(&ctx);
455                 else if(!strcmp(cmd, "key")){
456                     int x, y;
457                     if(sscanf(param, "%d,%d", &x, &y) == 2){
458                         // Find a key with this position
459                         ckb_key* key = 0;
460                         for(i = 0; i < keycount; i++){
461                             if(ctx.keys[i].x == x && ctx.keys[i].y == y){
462                                 key = ctx.keys + i;
463                                 break;
464                             }
465                         }
466                         if(key)
467                             ckb_keypress(&ctx, key, key->x, key->y, !strcmp(value, "down"));
468                         else
469                             ckb_keypress(&ctx, 0, x, y, !strcmp(value, "down"));
470                     } else {
471                         // Find a key with this name
472                         ckb_key* key = 0;
473                         for(i = 0; i < keycount; i++){
474                             if(!strcmp(ctx.keys[i].name, param)){
475                                 key = ctx.keys + i;
476                                 break;
477                             }
478                         }
```

```
479                         if(key)
480                             ckb_keypress(&ctx, key, key->x, key->y, !strcmp(value, "down"));
481                     }
482                 } else if(!strcmp(cmd, "frame")){
483                     int end = ckb_frame(&ctx);
484                     // Output the frame
485                     printf("begin frame\n");
486                     for(i = 0; i < ctx.keycount; i++){
487                         ckb_key* key = ctx.keys + i;
488                         printf("argb %s %02hhx%02hhx%02hhx%02hhx\n", key->name, key->
    a, key->r, key->g, key->b);
489                     }
490                     printf("end frame\n");
491                     if(end)
492                         break;
493                     fflush(stdout);
494                 } else {
495                     double delta = 0.;
496                     if(!strcmp(cmd, "time") && sscanf(param, "%lf", &delta) == 1)
497                         ckb_time(&ctx, delta);
498                 }
499             }
500             printf("end run\n");
501             fflush(stdout);
502             free(ctx.keys);
503             return 0;
504         }
505     }
506     printf("This program must be run from within ckb\n");
507     return -1;
508 }
```

Here is the call graph for this function:



13.13.3.14 void printurl (const char * *src*)

Definition at line 178 of file ckb-anim.h.

```
178                                 {
179     char out[strlen(src) * 3 + 1];
180     char* dst = out;
181     char s;
182     while((s = *src++)){
```

```
183         if(s <= ',' || s == '/' ||
184                 (s >= ':' && s <= '@') ||
185                 s == '[' || s == ']' ||
186                 s >= 0x7F){
187             char a = s >> 4, b = s & 0xF;
188             if(a >= 10)
189                 a += 'A' - 10;
190             else
191                 a += '0';
192             if(b >= 10)
193                 b += 'A' - 10;
194             else
195                 b += '0';
196             dst[0] = '%';
197             dst[1] = a;
198             dst[2] = b;
199             dst += 3;
200         } else
201             *dst++ = s;
202     }
203     *dst = '\0';
204     printf("%s", out);
205 }
```

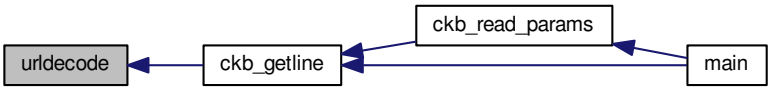
13.13.3.15 void urldecode (char * *dst*, const char * *src*)

Definition at line 208 of file ckb-anim.h.

Referenced by ckb_getline().

```
208                                             {
209     char a, b;
210     char s;
211     while((s = *src)){
212         if((s == '%') &&
213                 ((a = src[1]) && (b = src[2])) &&
214                 (isxdigit(a) && isxdigit(b))){
215             if(a >= 'a')
216                 a -= 'a'-'A';
217             if(a >= 'A')
218                 a -= 'A' - 10;
219             else
220                 a -= '0';
221             if(b >= 'a')
222                 b -= 'a'-'A';
223             if(b >= 'A')
224                 b -= 'A' - 10;
225             else
226                 b -= '0';
227             *dst++ = 16*a + b;
228             src += 3;
229         } else {
230             *dst++ = s;
231             src++;
232         }
233     }
234     *dst = '\0';
235 }
```

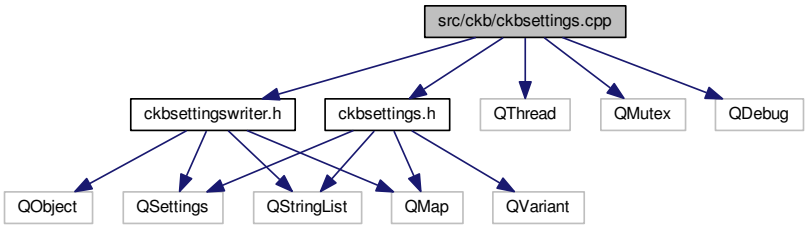
Here is the caller graph for this function:



13.14 src/ckb/ckbsettings.cpp File Reference

```
#include "ckbsettings.h"
#include "ckbsettingswriter.h"
#include <QThread>
#include <QMutex>
#include <QDebug>
```

Include dependency graph for ckbsettings.cpp:



Macros

- #define lockMutex QMutexLocker locker(backing == _globalSettings ? &settingsMutex : 0)
- #define lockMutexStatic QMutexLocker locker(&settingsMutex)
- #define lockMutexStatic2 QMutexLocker locker2(&settingsMutex)
- #define lockMutexCache QMutexLocker locker(&settingsCacheMutex)

Functions

- QAtomicInt cacheWritesInProgress (0)
- QMutex settingsMutex (QMutex::Recursive)
- QMutex settingsCacheMutex (QMutex::Recursive)
- static QSettings * globalSettings ()

Variables

- static QSettings * _globalSettings = 0
- static QThread * globalThread = 0
- static QMap< QString, QVariant > globalCache

13.14.1 Macro Definition Documentation

13.14.1.1 #define lockMutex QMutexLocker locker(backing == _globalSettings ? &settingsMutex : 0)

Definition at line 15 of file ckbsettings.cpp.

Referenced by CkbSettings::childGroups(), CkbSettings::childKeys(), CkbSettings::contains(), and CkbSettings::value().

13.14.1.2 #define lockMutexCache QMutexLocker locker(&settingsCacheMutex)

Definition at line 18 of file ckbsettings.cpp.

Referenced by CkbSettings::get(), CkbSettings::set(), and CkbSettings::setValue().

13.14.1.3 #define lockMutexStatic QMutexLocker locker(&settingsMutex)

Definition at line 16 of file ckbsettings.cpp.

Referenced by globalSettings(), and CkbSettings::set().

13.14.1.4 #define lockMutexStatic2 QMutexLocker locker2(&settingsMutex)

Definition at line 17 of file ckbsettings.cpp.

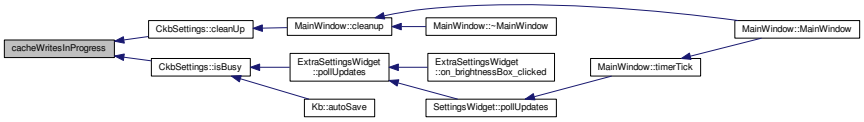
Referenced by CkbSettings::get().

13.14.2 Function Documentation

13.14.2.1 QAtomicInt cacheWritesInProgress (0)

Referenced by CkbSettings::cleanUp(), and CkbSettings::isBusy().

Here is the caller graph for this function:



13.14.2.2 static QSettings* globalSettings ( ) [static]

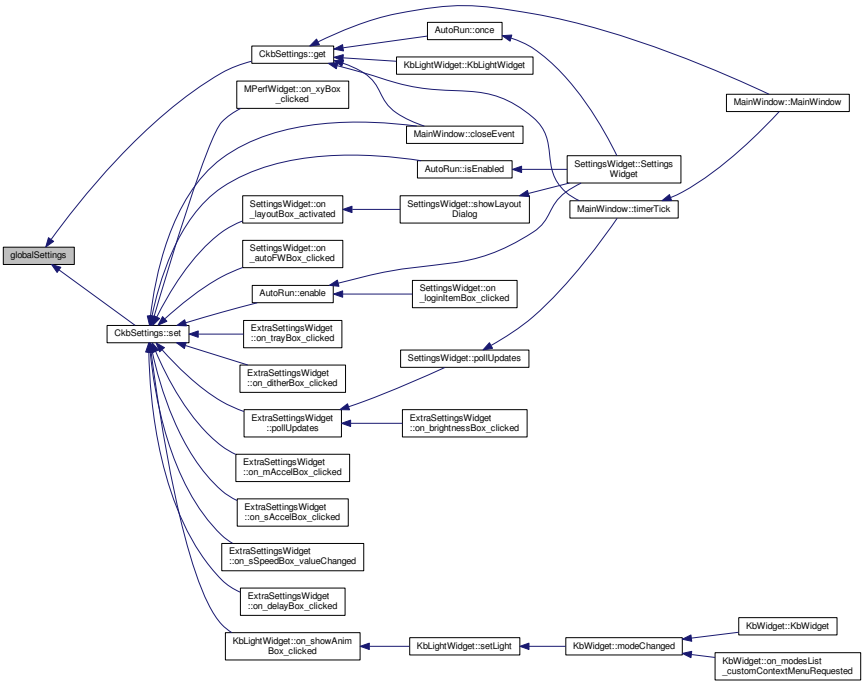
Definition at line 27 of file ckbsettings.cpp.

References _globalSettings, and lockMutexStatic.

Referenced by CkbSettings::get(), and CkbSettings::set().

```
27                                {
28     if(!_globalSettings){
29         lockMutexStatic;
30         if(!(volatile QSettings*)_globalSettings){   // Check again after locking mutex in
       case another thread created the object
31             // Put the settings object in a separate thread so that it won't lock up the GUI when it syncs
32             globalThread = new QThread;
33             globalThread->start();
34             _globalSettings = new QSettings;
35             qInfo() << "Path  to settings is" << _globalSettings->fileName();
36             _globalSettings->moveToThread(globalThread);
37         }
38     }
39     return _globalSettings;
40 }
```

Here is the caller graph for this function:



13.14.2.3 QMutex settingsCacheMutex (QMutex::Recursive)

13.14.2.4 QMutex settingsMutex (QMutex::Recursive)

13.14.3 Variable Documentation

13.14.3.1 QSettings* _globalSettings = 0 [static]

Definition at line 8 of file ckbsettings.cpp.

Referenced by CkbSettings::cleanUp(), and globalSettings().

13.14.3.2 QMap<QString, QVariant> globalCache [static]

Definition at line 12 of file ckbsettings.cpp.

Referenced by CkbSettings::get(), CkbSettings::set(), and CkbSettings::setValue().

13.14.3.3 QThread* globalThread = 0 [static]

Definition at line 9 of file ckbsettings.cpp.

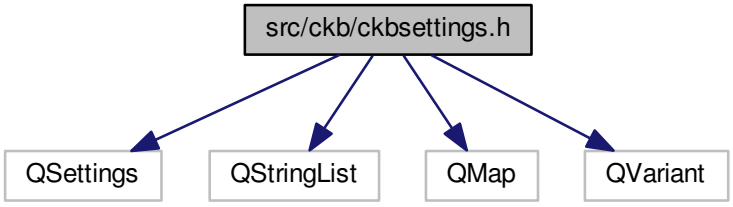
Referenced by CkbSettings::cleanUp().

13.15 src/ckb/ckbsettings.h File Reference

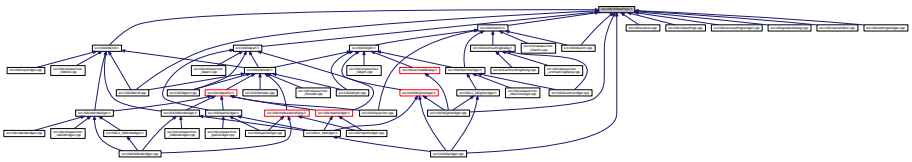
```
#include <QSettings>
```

```
#include <QStringList>
#include <QMap>
#include <QVariant>
```

Include dependency graph for ckbsettings.h:



This graph shows which files directly or indirectly include this file:



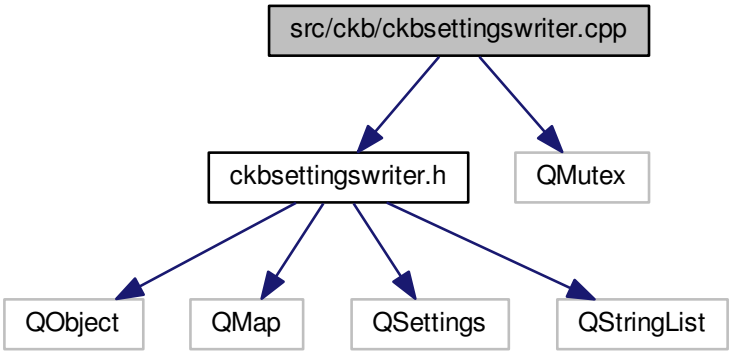
Data Structures

- class CkbSettings
- class SGroup

13.16 src/ckb/ckbsettingswriter.cpp File Reference

```
#include "ckbsettingswriter.h"
#include <QMutex>
```

Include dependency graph for ckbsettingswriter.cpp:



Macros

- #define lockMutex QMutexLocker locker(backing == _globalSettings ? &mutex : 0)
- #define lockMutexStatic QMutexLocker locker(&settingsMutex)
- #define lockMutexStatic2 QMutexLocker locker2(&settingsMutex)
- #define lockMutexCache QMutexLocker locker(&settingsCacheMutex)

Variables

- QAtomicInt cacheWritesInProgress
- QMutex settingsMutex
- QMutex settingsCacheMutex

13.16.1 Macro Definition Documentation

13.16.1.1 #define lockMutex QMutexLocker locker(backing == _globalSettings ? &mutex : 0)

Definition at line 7 of file ckbsettingswriter.cpp.

13.16.1.2 #define lockMutexCache QMutexLocker locker(&settingsCacheMutex)

Definition at line 10 of file ckbsettingswriter.cpp.

13.16.1.3 #define lockMutexStatic QMutexLocker locker(&settingsMutex)

Definition at line 8 of file ckbsettingswriter.cpp.

Referenced by CkbSettingsWriter::run().

13.16.1.4 #define lockMutexStatic2 QMutexLocker locker2(&settingsMutex)

Definition at line 9 of file ckbsettingswriter.cpp.

13.16.2 Variable Documentation

13.16.2.1 QAtomicInt cacheWritesInProgress

Referenced by CkbSettingsWriter::CkbSettingsWriter(), and CkbSettingsWriter::~CkbSettingsWriter().

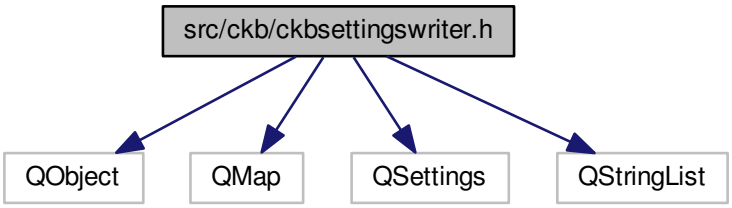
13.16.2.2 QMutex settingsCacheMutex

13.16.2.3 QMutex settingsMutex

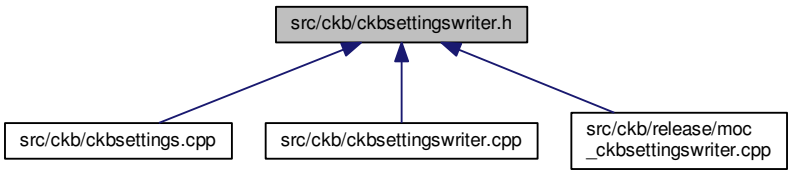
13.17 src/ckb/ckbsettingswriter.h File Reference

```
#include <QObject>
#include <QMap>
#include <QSettings>
#include <QStringList>
```

Include dependency graph for ckbsettingswriter.h:



This graph shows which files directly or indirectly include this file:



Data Structures

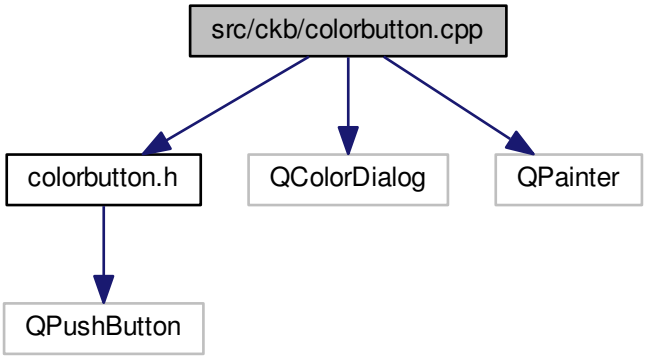
- class CkbSettingsWriter

13.18 src/ckb/colorbutton.cpp File Reference

```
#include "colorbutton.h"
```

```
#include <QColorDialog>
#include <QPainter>
```

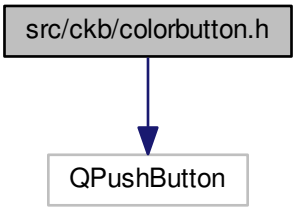
Include dependency graph for colorbutton.cpp:



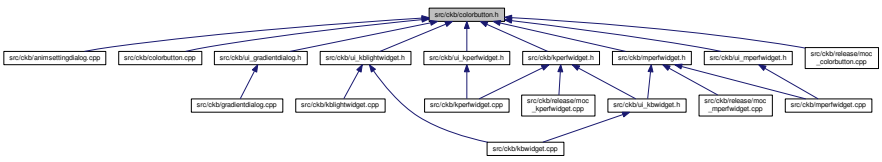
13.19 src/ckb/colorbutton.h File Reference

```
#include <QPushButton>
```

Include dependency graph for colorbutton.h:



This graph shows which files directly or indirectly include this file:



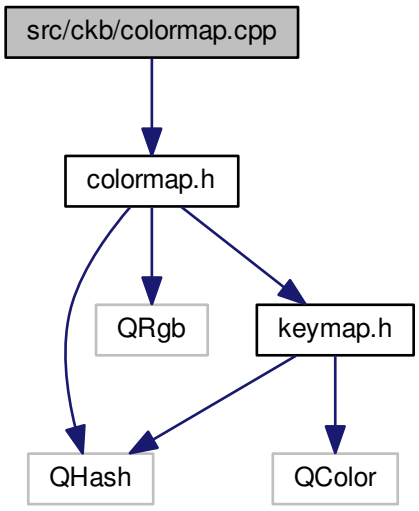
Data Structures

- class ColorButton

13.20 src/ckb/colormap.cpp File Reference

```
#include "colormap.h"
```

Include dependency graph for colormap.cpp:



Functions

- static int qs_strcmp (const void *lhs, const void *rhs)

13.20.1 Function Documentation

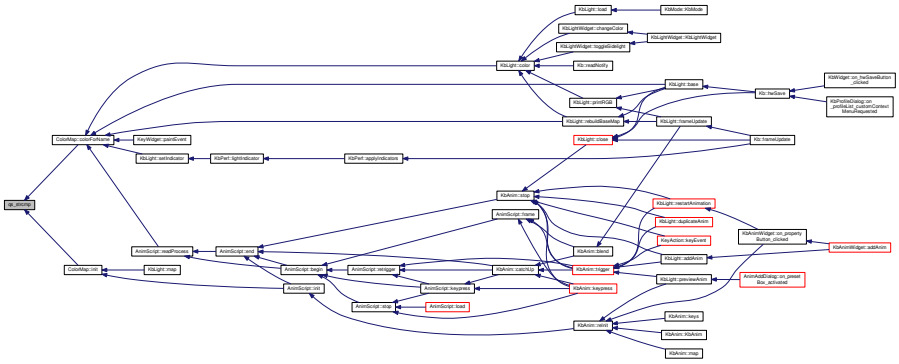
13.20.1.1 static int qs_strcmp (const void * *lhs*, const void * *rhs*) [static]

Definition at line 65 of file colormap.cpp.

Referenced by ColorMap::colorForName(), and ColorMap::init().

```
65                                                      {
66     return strcmp(*(const char**)lhs, *(const char**)rhs);
67 }
```

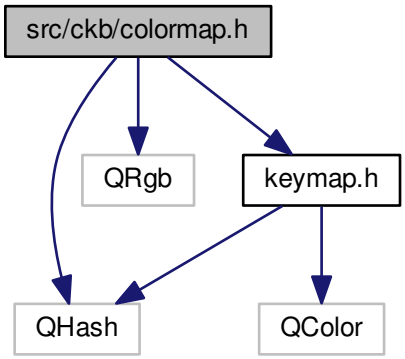
Here is the caller graph for this function:



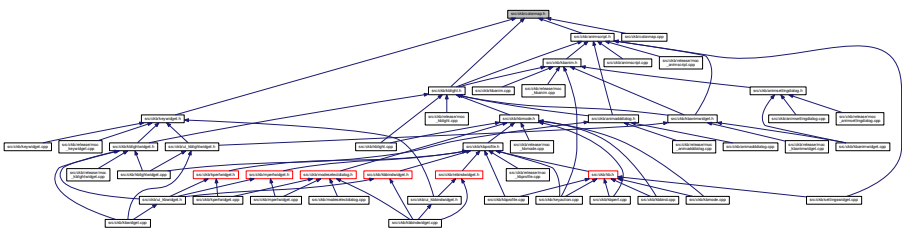
13.21 src/ckb/colormap.h File Reference

```
#include <QHash>
#include <QRgb>
#include "keymap.h"
```

Include dependency graph for colormap.h:



This graph shows which files directly or indirectly include this file:



Data Structures

- class ColorMap

Typedefs

- typedef QHash< QString, QRgb > QColorMap
- typedef QHashIterator< QString, QRgb > QColorMapIterator
- typedef QMutableHashIterator < QString, QRgb > QMutableColorMapIterator

13.21.1 Typedef Documentation

13.21.1.1 typedef QHash<QString, QRgb> QColorMap

Definition at line 9 of file colormap.h.

13.21.1.2 typedef QHashIterator<QString, QRgb> QColorMapIterator

Definition at line 10 of file colormap.h.

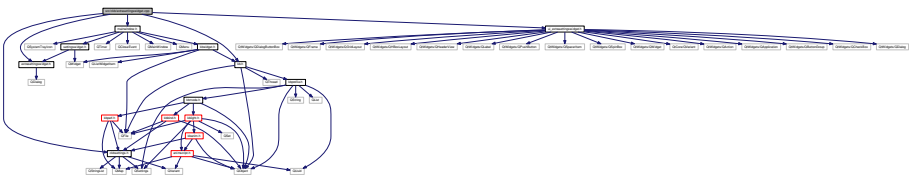
13.21.1.3 typedef QMutableHashIterator<QString, QRgb> QMutableColorMapIterator

Definition at line 11 of file colormap.h.

13.22 src/ckb/extrasettingswidget.cpp File Reference

```
#include "extrasettingswidget.h"
#include "ui_extrasettingswidget.h"
#include "kb.h"
#include "mainwindow.h"
#include "ckbsettings.h"
```

Include dependency graph for extrasettingswidget.cpp:



Variables

- static int lastSharedDimming = -2

13.22.1 Variable Documentation

13.22.1.1 int lastSharedDimming = -2 [static]

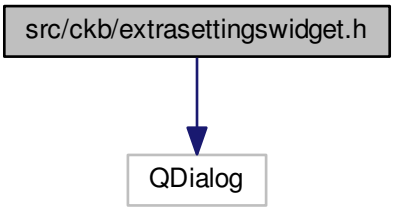
Definition at line 8 of file extrasettingswidget.cpp.

Referenced by ExtraSettingsWidget::ExtraSettingsWidget(), and ExtraSettingsWidget::pollUpdates().

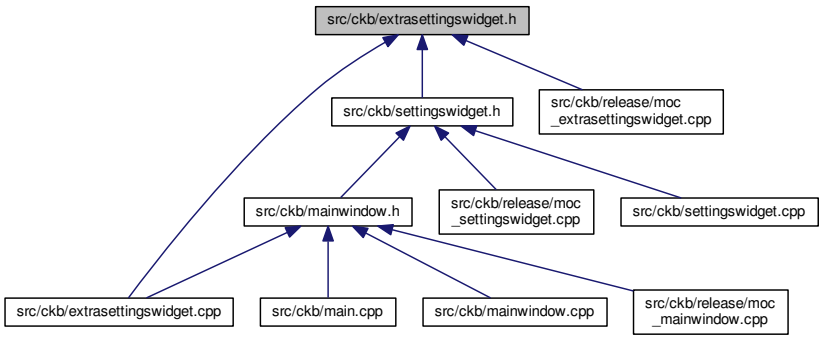
13.23 src/ckb/extrasettingswidget.h File Reference

```
#include <QDialog>
```

Include dependency graph for extrasettingswidget.h:



This graph shows which files directly or indirectly include this file:



Data Structures

- class ExtraSettingsWidget

Namespaces

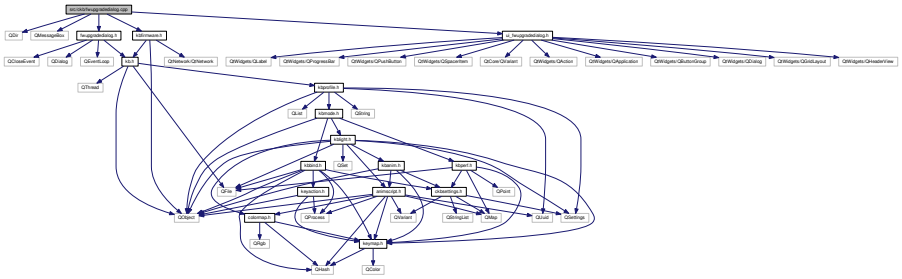
- Ui

13.24 src/ckb/fwupgradedialog.cpp File Reference

```
#include <QDir>
```

```
#include <QMessageBox>
#include "fwupgradedialog.h"
#include "kbfirmware.h"
#include "ui_fwupgradedialog.h"
```

Include dependency graph for fwupgradedialog.cpp:



Data Structures

- struct Kbld

Functions

- static const int DIALOG_HEIGHT_MAX (240)
- static float verifyFw (const QByteArray &blob, const QString &features)

Variables

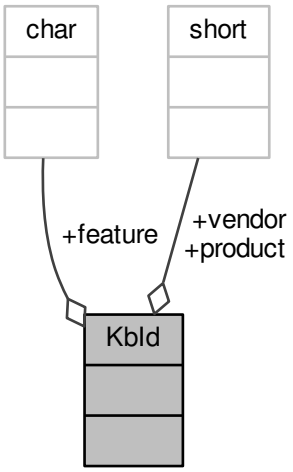
- static Kbld ids []
- static const int DIALOG_WIDTH = 420
- static const int DIALOG_HEIGHT_MIN = 200

13.24.1 Data Structure Documentation

13.24.1.1 struct Kbld

Definition at line 8 of file fwupgradedialog.cpp.

Collaboration diagram for KbId:



Data Fields

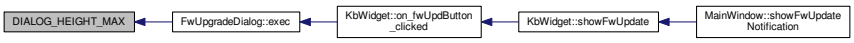
const char *	feature	
short	product	
short	vendor	

13.24.2 Function Documentation

13.24.2.1 static const int DIALOG_HEIGHT_MAX (240) [static]

Referenced by FwUpgradeDialog::exec().

Here is the caller graph for this function:



13.24.2.2 static float verifyFw (const QByteArray & *blob*, const QString & *features*) [static]

Definition at line 86 of file fwupgradedialog.cpp.

References ids.

Referenced by FwUpgradeDialog::exec().

```
86                                                                      {
87     if(blob.length() < 0x0108)
88         return 0.f;
89     const char* bData = blob.data();
90     // Make sure it matches this device based on the vendor and product IDs embedded in the blob
91     bool match = false;
```

```
92      for(uint i = 0; i < sizeof(ids)/sizeof(KbId); i++){
93          if(!memcmp(&ids[i].vendor, bData + 0x102, 2) && !memcmp(&ids[i].product, bData + 0x104, 2)
94                  && features.startsWith(ids[i].feature, Qt::CaseInsensitive)){
95              match = true;
96              break;
97          }
98      }
99      if(!match)
100          return 0.f;
101      // Copy the version from the blob
102      short version;
103      memcpy(&version, bData + 0x106, 2);
104      // Un-hexify it
105      return QString::number(version, 16).toFloat() / 100.f;
106  }
```

Here is the caller graph for this function:

13.24.3 Variable Documentation

13.24.3.1 const int DIALOG_HEIGHT_MIN = 200 [static]

Definition at line 30 of file fwupgradedialog.cpp.

Referenced by FwUpgradeDialog::FwUpgradeDialog(), and FwUpgradeDialog::on_actionButton_clicked().

13.24.3.2 const int DIALOG_WIDTH = 420 [static]

Definition at line 29 of file fwupgradedialog.cpp.

Referenced by FwUpgradeDialog::exec(), FwUpgradeDialog::FwUpgradeDialog(), and FwUpgradeDialog::on_actionButton_clicked().

13.24.3.3 KbId ids[] [static]

Initial value:

```
= {

    { 0x1b1c, 0x1b17, "corsair k65 rgb" },
    { 0x1b1c, 0x1b3f, "corsair k68" },
    { 0x1b1c, 0x1b13, "corsair k70 rgb" },
    { 0x1b1c, 0x1b11, "corsair k95 rgb" },
    { 0x1b1c, 0x1b15, "corsair strafe monochrome" },
    { 0x1b1c, 0x1b20, "corsair strafe rgb" },

    { 0x1b1c, 0x1b12, "corsair m65 rgb" },
    { 0x1b1c, 0x1b1e, "corsair scimitar rgb" },

    { 0x1b1c, 0x1b32, "corsair sabre optical rgb" }
}
```

Definition at line 14 of file fwupgradedialog.cpp.

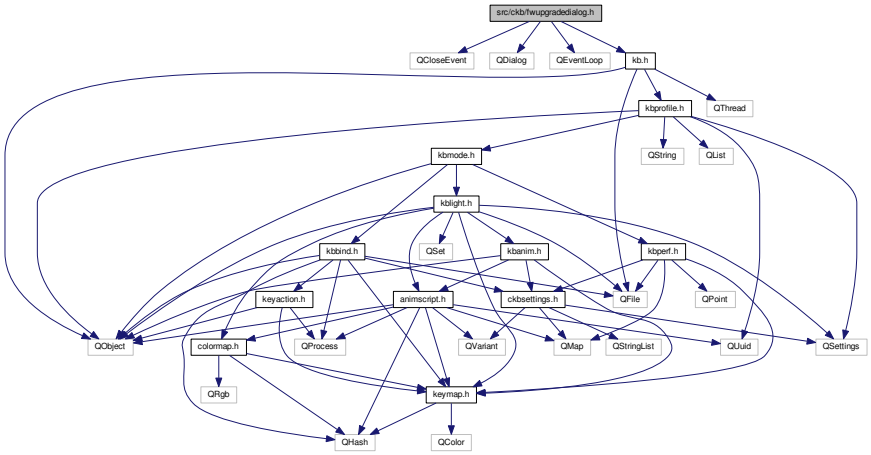
Referenced by verifyFw().

13.25 src/ckb/fwupgradedialog.h File Reference

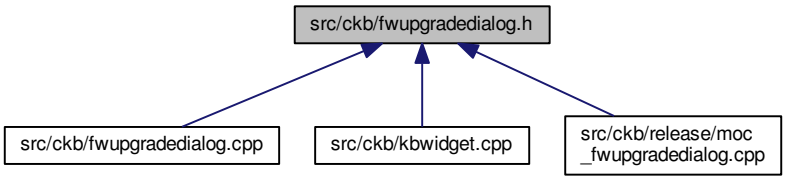
```
#include <QCloseEvent>
```

```
#include <QDialog>
#include <QEventLoop>
#include "kb.h"
```

Include dependency graph for fwupgradedialog.h:



This graph shows which files directly or indirectly include this file:



Data Structures

- class FwUpgradeDialog

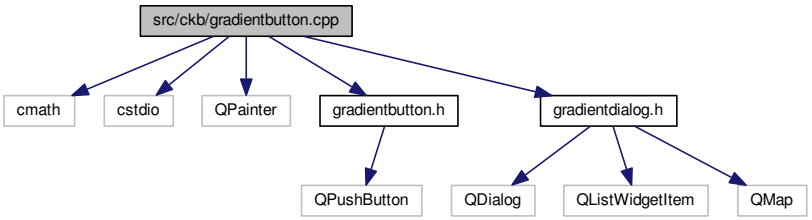
Namespaces

- Ui

13.26 src/ckb/gradientbutton.cpp File Reference

```
#include <cmath>
#include <cstdio>
#include <QPainter>
#include "gradientbutton.h"
#include "gradientdialog.h"
```

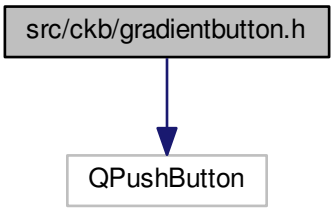
Include dependency graph for gradientbutton.cpp:



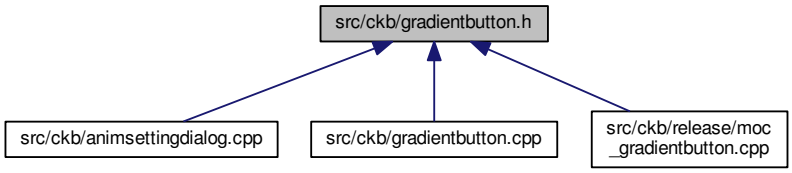
13.27 src/ckb/gradientbutton.h File Reference

```
#include <QPushButton>
```

Include dependency graph for gradientbutton.h:



This graph shows which files directly or indirectly include this file:



Data Structures

- class GradientButton

13.28 src/ckb/gradientdialog.cpp File Reference

```
#include <cmath>
#include <QMessageBox>
#include <QPainter>
#include <QSettings>
#include "ckbsettings.h"
#include "gradientdialog.h"
#include "ui_gradientdialog.h"
```

Include dependency graph for gradientdialog.cpp:

Variables

- static const QString prefsPath = "Stored Gradients"

13.28.1 Variable Documentation

13.28.1.1 const QString prefsPath = "Stored Gradients" [static]

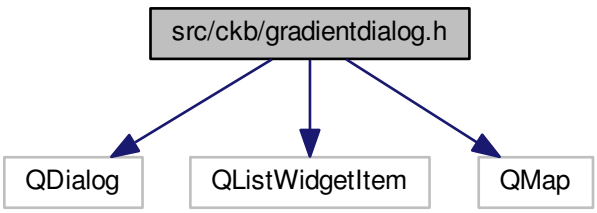
Definition at line 9 of file gradientdialog.cpp.

Referenced by GradientDialog::GradientDialog(), and GradientDialog::~GradientDialog().

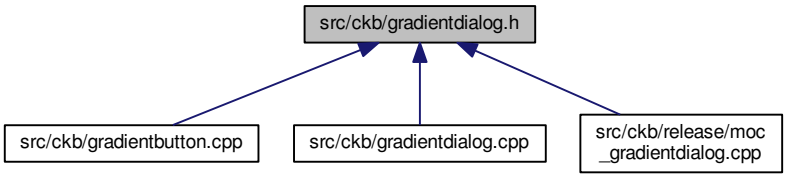
13.29 src/ckb/gradientdialog.h File Reference

```
#include <QDialog>
#include <QListWidgetItem>
#include <QMap>
```

Include dependency graph for gradientdialog.h:



This graph shows which files directly or indirectly include this file:



Data Structures

- class GradientDialog
- struct GradientDialog::Preset

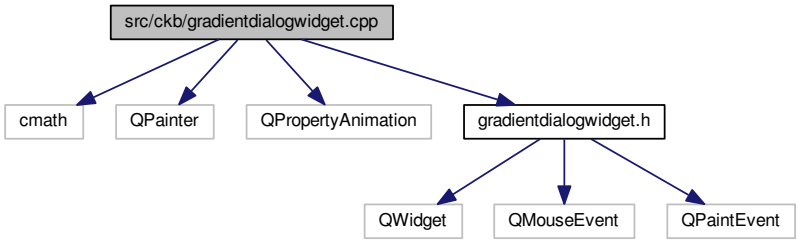
Namespaces

- Ui

13.30 src/ckb/gradientdialogwidget.cpp File Reference

```
#include <cmath>
#include <QPainter>
#include <QPropertyAnimation>
#include "gradientdialogwidget.h"
```

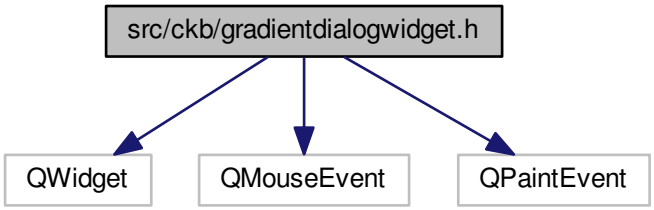
Include dependency graph for gradientdialogwidget.cpp:



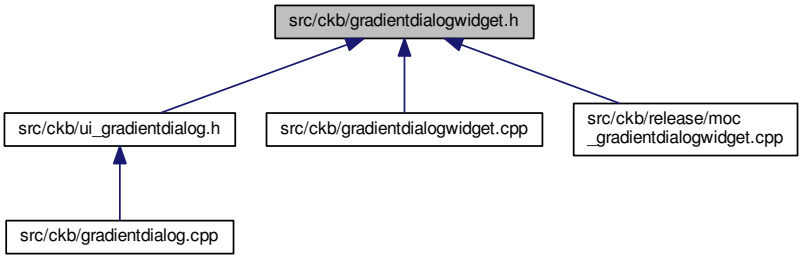
13.31 src/ckb/gradientdialogwidget.h File Reference

```
#include <QWidget>
#include <QMouseEvent>
#include <QPaintEvent>
```

Include dependency graph for gradientdialogwidget.h:



This graph shows which files directly or indirectly include this file:



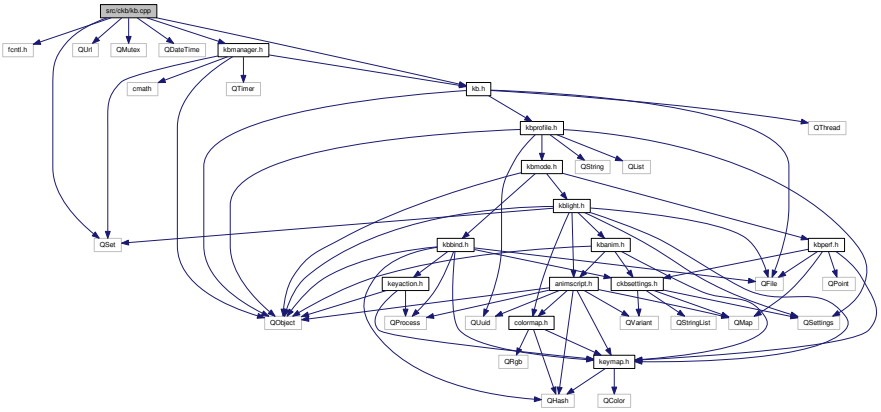
Data Structures

- class GradientDialogWidget

13.32 src/ckb/kb.cpp File Reference

```
#include <fcntl.h>
#include <QSet>
#include <QUrl>
#include <QMutex>
#include <QDateTime>
#include "kb.h"
#include "kbmanager.h"
```

Include dependency graph for kb.cpp:



Variables

- static QSet< Kb * > activeDevices
- static QSet< QString > notifyPaths
- static QMutex notifyPathMutex

13.32.1 Variable Documentation

13.32.1.1 QSet<Kb*> activeDevices [static]

Definition at line 10 of file kb.cpp.

Referenced by Kb::dither(), Kb::frameRate(), Kb::Kb(), Kb::layout(), Kb::macroDelay(), Kb::mouseAccel(), Kb::scrollSpeed(), and Kb::~Kb().

13.32.1.2 QMutex notifyPathMutex [static]

Definition at line 13 of file kb.cpp.

Referenced by Kb::Kb(), and Kb::run().

13.32.1.3 QSet<QString> notifyPaths [static]

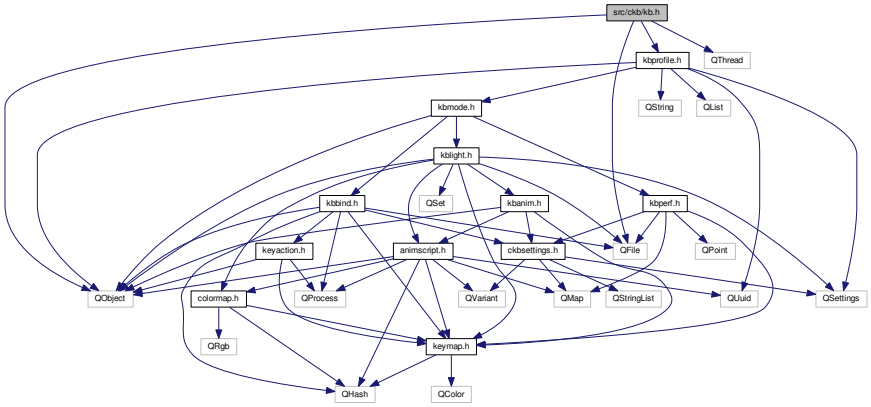
Definition at line 12 of file kb.cpp.

Referenced by Kb::Kb(), Kb::run(), and Kb::~Kb().

13.33 src/ckb/kb.h File Reference

```
#include <QObject>
#include <QFile>
#include <QThread>
#include "kbprofile.h"
```

Include dependency graph for kb.h:



This graph shows which files directly or indirectly include this file:

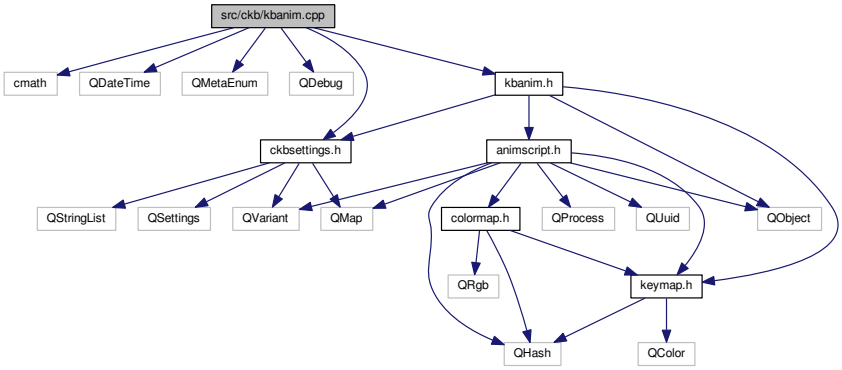
Data Structures

- class Kb

13.34 src/ckb/kbanim.cpp File Reference

```
#include <cmath>
#include <QDateTime>
#include <QMetaEnum>
#include <QDebug>
#include "ckbsettings.h"
#include "kbanim.h"
```

Include dependency graph for kbanim.cpp:



Typedefs

- typedef float(∗ blendFunc)(float, float)

Functions

- static float blendNormal (float bg, float fg)
- static float blendAdd (float bg, float fg)
- static float blendSubtract (float bg, float fg)
- static float blendMultiply (float bg, float fg)
- static float blendDivide (float bg, float fg)

Variables

- static blendFunc functions [5] = { blendNormal, blendAdd, blendSubtract, blendMultiply, blendDivide }

13.34.1 Typedef Documentation

13.34.1.1 typedef float(∗ blendFunc)(float, float)

Definition at line 362 of file kbanim.cpp.

13.34.2 Function Documentation

13.34.2.1 static float blendAdd (float *bg,* float *fg*) [static]

Definition at line 337 of file kbanim.cpp.

```
337                                             {
338     float res = bg + fg;
339     if(res > 1.f)
340         res = 1.f;
341     return res;
342 }
```

13.34.2.2 static float blendDivide (float *bg,* float *fg*) [static]

Definition at line 355 of file kbanim.cpp.

```
355                                               {
356     float res = bg / fg;
357     if(res > 1.f)
358         res = 1.f;
359     return res;
360 }
```

13.34.2.3 static float blendMultiply (float *bg,* float *fg*) [static]

Definition at line 351 of file kbanim.cpp.

```
351                                                 {
352     return bg * fg;
353 }
```

13.34.2.4 static float blendNormal (float *bg*, float *fg*) [static]

Definition at line 333 of file kbanim.cpp.

```
333                                          {
334     return fg;
335 }
```

13.34.2.5 static float blendSubtract (float *bg*, float *fg*) [static]

Definition at line 344 of file kbanim.cpp.

```
344                                             {
345     float res = bg - fg;
346     if(res < 0.f)
347         res = 0.f;
348     return res;
349 }
```

13.34.3 Variable Documentation

13.34.3.1 blendFunc functions[5] = { blendNormal, blendAdd, blendSubtract, blendMultiply, blendDivide } [static]

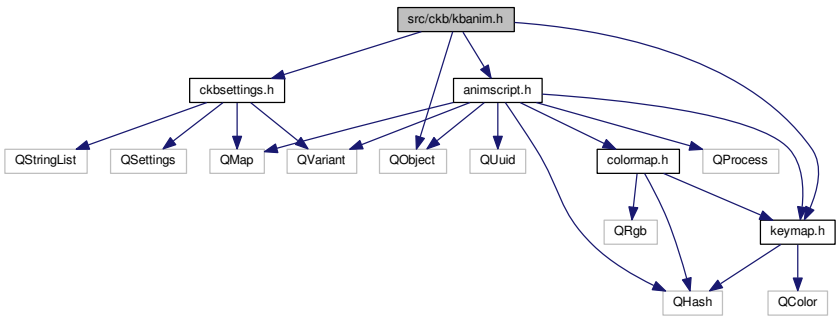
Definition at line 363 of file kbanim.cpp.

Referenced by KbAnim::blend().

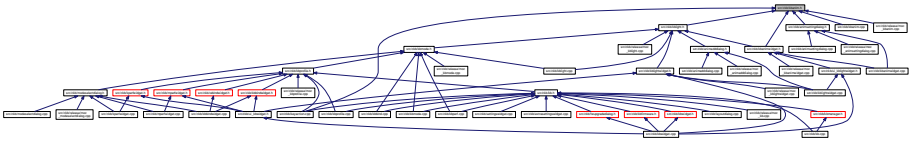
13.35 src/ckb/kbanim.h File Reference

```
#include <QObject>
#include "ckbsettings.h"
#include "animscript.h"
#include "keymap.h"
```

Include dependency graph for kbanim.h:



This graph shows which files directly or indirectly include this file:



Data Structures

- class KbAnim

13.36 src/ckb/kbanimwidget.cpp File Reference

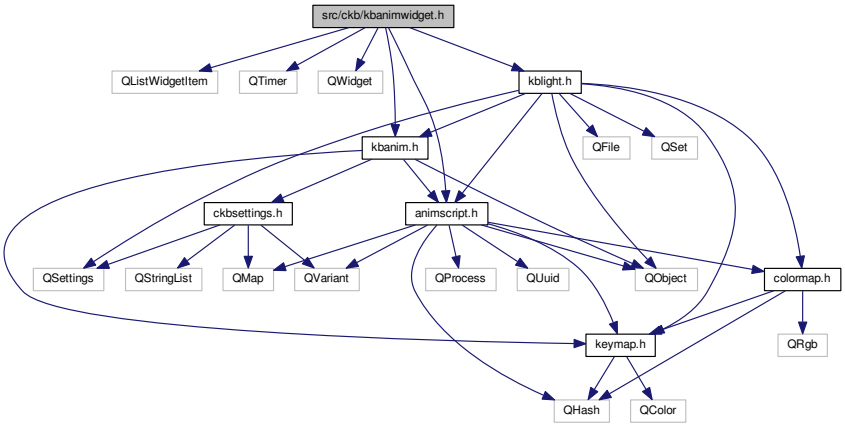
```
#include <QDateTime>
#include <QMenu>
#include "animsettingdialog.h"
#include "kbanimwidget.h"
#include "ui_kbanimwidget.h"
```

Include dependency graph for kbanimwidget.cpp:

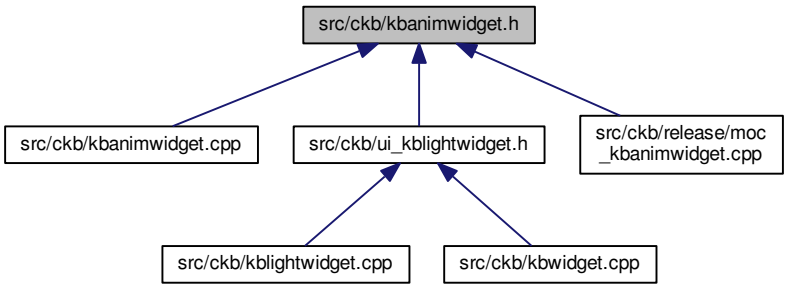
13.37 src/ckb/kbanimwidget.h File Reference

```
#include <QListWidgetItem>
#include <QTimer>
#include <QWidget>
#include "animscript.h"
#include "kbanim.h"
#include "kblight.h"
```

Include dependency graph for kbanimwidget.h:



This graph shows which files directly or indirectly include this file:



Data Structures

- class KbAnimWidget

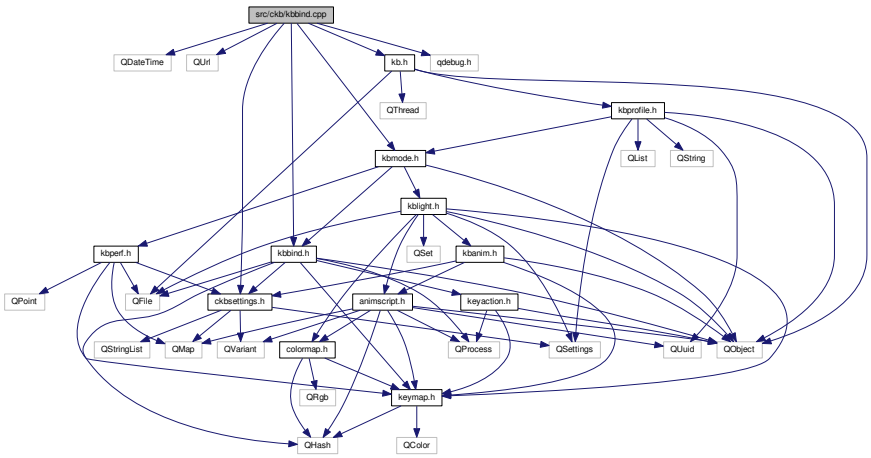
Namespaces

- Ui

13.38 src/ckb/kbbind.cpp File Reference

```
#include <QDateTime>
#include <QUrl>
#include "ckbsettings.h"
#include "kbbind.h"
#include "kbmode.h"
#include "kb.h"
#include "qdebug.h"
```

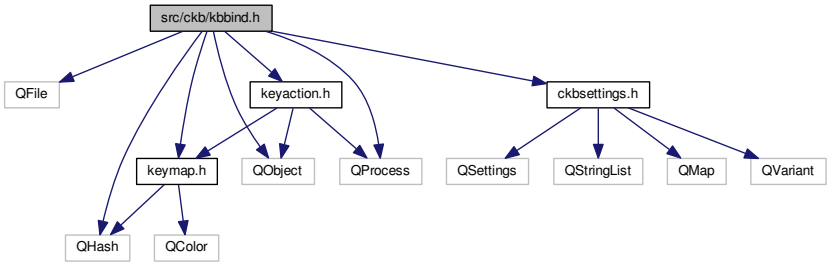
Include dependency graph for kbbind.cpp:



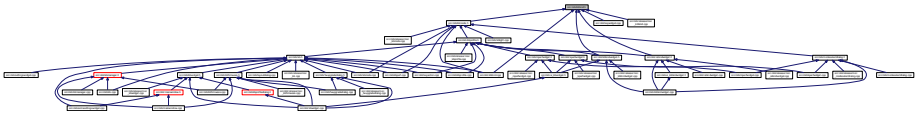
13.39 src/ckb/kbbind.h File Reference

```
#include <QFile>
#include <QHash>
#include <QObject>
#include <QProcess>
#include "ckbsettings.h"
#include "keymap.h"
#include "keyaction.h"
```

Include dependency graph for kbbind.h:



This graph shows which files directly or indirectly include this file:



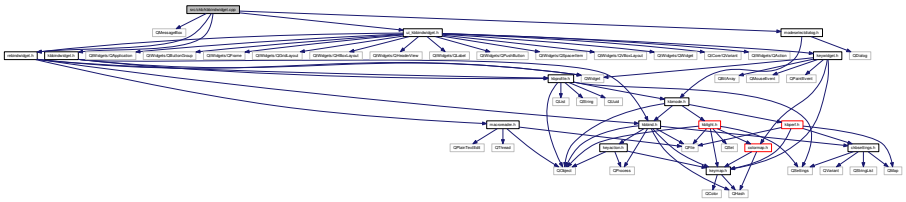
Data Structures

- class KbBind

13.40 src/ckb/kbbindwidget.cpp File Reference

```
#include <QMessageBox>
#include "kbbindwidget.h"
#include "modeselectdialog.h"
#include "rebindwidget.h"
#include "ui_kbbindwidget.h"
```

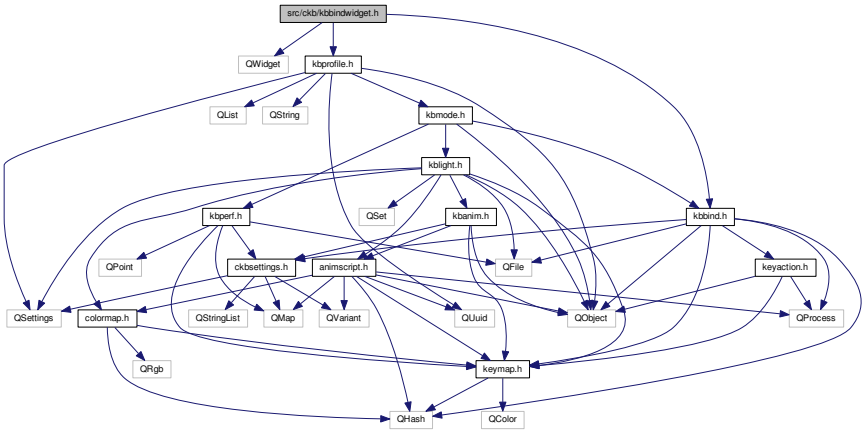
Include dependency graph for kbbindwidget.cpp:



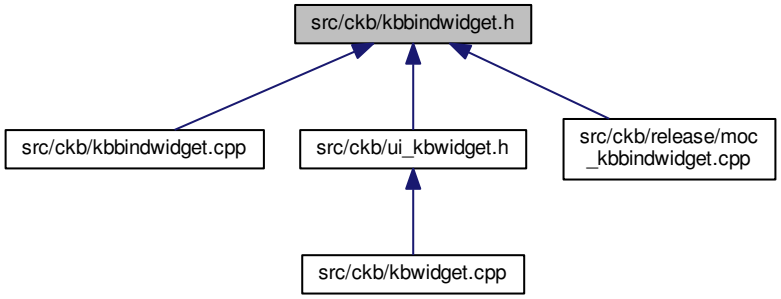
13.41 src/ckb/kbbindwidget.h File Reference

```
#include <QWidget>
#include "kbbind.h"
#include "kbprofile.h"
```

Include dependency graph for kbbindwidget.h:



This graph shows which files directly or indirectly include this file:



Data Structures

- class KbBindWidget

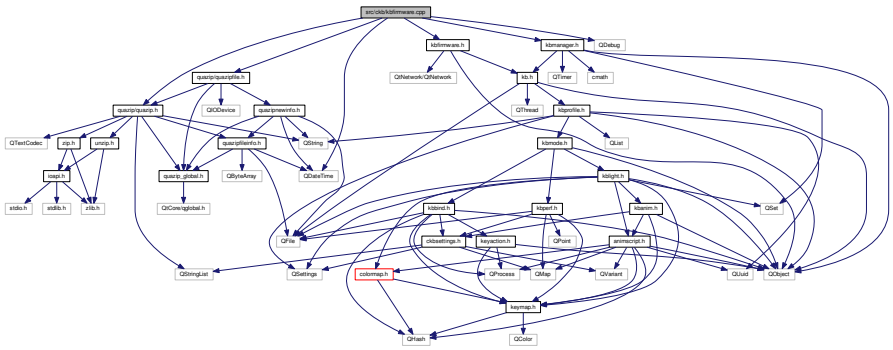
Namespaces

- Ui

13.42 src/ckb/kbfirmware.cpp File Reference

```
#include "kbfirmware.h"
#include "kbmanager.h"
#include "quazip/quazip.h"
#include "quazip/quazipfile.h"
#include <QDateTime>
#include <QDebug>
```

Include dependency graph for kbfirmware.cpp:



Functions

- static QString tableName (const QString &features)

Variables

- static const quint64 AUTO_CHECK_TIME = 60 ∗ 60 ∗ 1000

13.42.1 Function Documentation

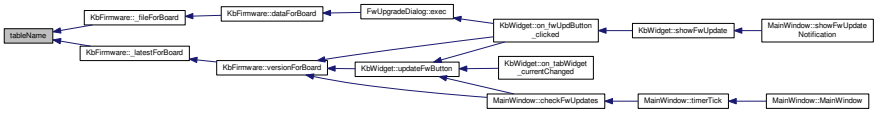
13.42.1.1 static QString tableName (const QString & *features*) [static]

Definition at line 150 of file kbfirmware.cpp.

Referenced by KbFirmware::_fileForBoard(), and KbFirmware::_latestForBoard().

```
150                                                {
151     QStringList components = features.split(" ");
152     if(components.length() < 2)
153         return "";
154     // First two components are vendor and model
155     QString vendorModel = components[0].toUpper() + "-" + components[1].toUpper();
156     // Add "RGB" on RGB boards
157     if(features.contains("rgb") && !features.contains("monochrome"))
158         vendorModel += "RGB";
159     return vendorModel;
160 }
```

Here is the caller graph for this function:



13.42.2 Variable Documentation

13.42.2.1 const quint64 AUTO_CHECK_TIME = 60 ∗ 60 ∗ 1000 [static]

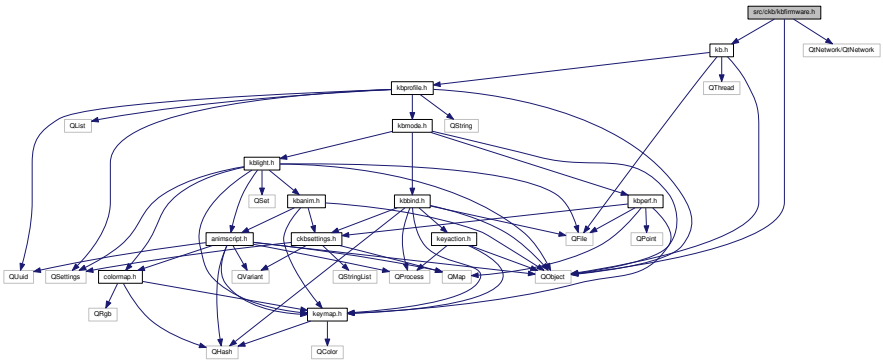
Definition at line 9 of file kbfirmware.cpp.

Referenced by KbFirmware::_checkUpdates().

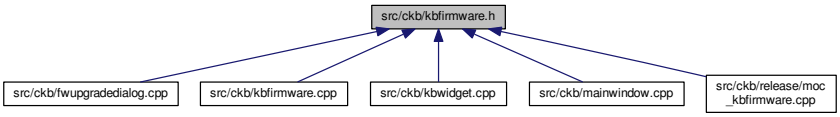
13.43 src/ckb/kbfirmware.h File Reference

```
#include "kb.h"
#include <QObject>
#include <QtNetwork/QtNetwork>
```

Include dependency graph for kbfirmware.h:



This graph shows which files directly or indirectly include this file:



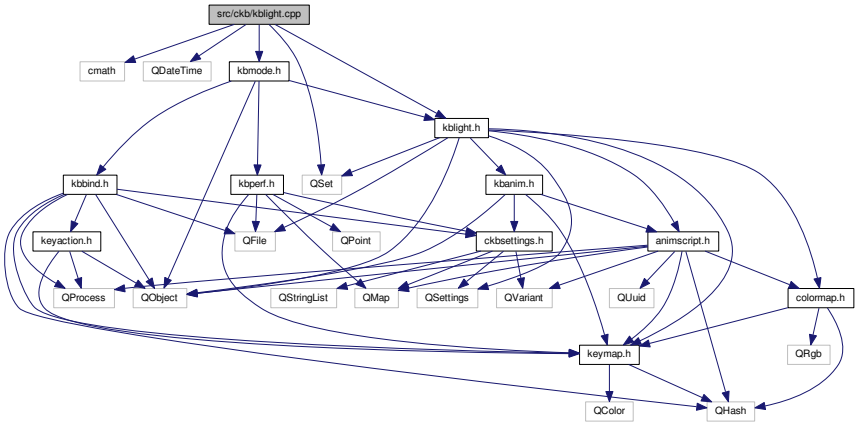
Data Structures

- class KbFirmware
- struct KbFirmware::FW

13.44 src/ckb/kblight.cpp File Reference

```
#include <cmath>
#include <QDateTime>
#include <QSet>
#include "kblight.h"
#include "kbmode.h"
```

Include dependency graph for kblight.cpp:



Functions

- static float sToL (float srgb)
- static float lToS (float linear)
- QRgb monoRgb (float r, float g, float b)

Variables

- static int _shareDimming = -1
- static QSet< KbLight * > activeLights

13.44.1 Function Documentation

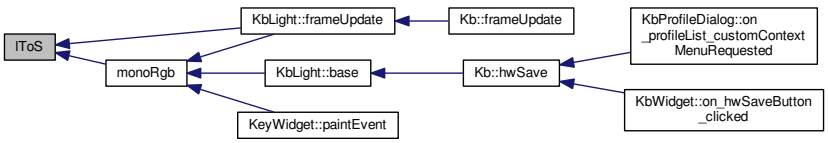
13.44.1.1 static float lToS (float *linear*) [static]

Definition at line 277 of file kblight.cpp.

Referenced by KbLight::frameUpdate(), and monoRgb().

```
277                                {
278     if(linear <= 0.0031308f)
279         return 12.92f * linear * 255.f;
280     return (1.055f * pow(linear, 1.f / 2.4f) - 0.055f) * 255.f;
281 }
```

Here is the caller graph for this function:



13.44.1.2 QRgb monoRgb (float *r*, float *g*, float *b*)

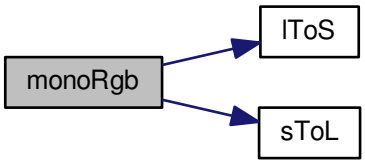
Definition at line 284 of file kblight.cpp.

References lToS(), and sToL().

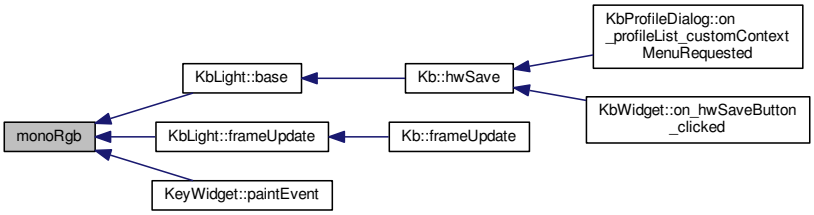
Referenced by KbLight::base(), KbLight::frameUpdate(), and KeyWidget::paintEvent().

```
284                                     {
285     // It's important to use a linear colorspace for this, otherwise the colors will appear inconsistent
286     // Note that although we could use linear space for alpha blending or the animation blending functions,
        we don't.
287     // The reason for this is that photo manipulation programs don't do it either, so even though the
        result would technically be more correct,
288     // it would look wrong to most people.
289     r = sToL(r);
290     g = sToL(g);
291     b = sToL(b);
292     int value = round(lToS((r + g + b) / 3.f));
293     return qRgb(value, value, value);
294 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



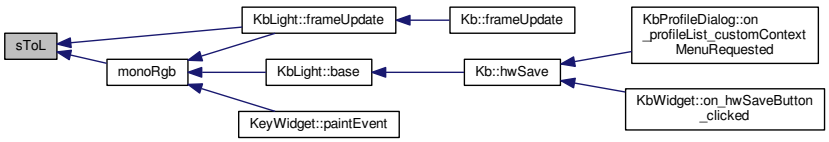
13.44.1.3 static float sToL (float *srgb*) [static]

Definition at line 270 of file kblight.cpp.

Referenced by KbLight::frameUpdate(), and monoRgb().

```
270                                {
271     srgb /= 255.f;
272     if(srgb <= 0.04045f)
273         return srgb / 12.92f;
274     return pow((srgb + 0.055f) / 1.055f, 2.4f);
275 }
```

Here is the caller graph for this function:



13.44.2 Variable Documentation

13.44.2.1 int _shareDimming = -1 [static]

Definition at line 7 of file kblight.cpp.

Referenced by KbLight::dimming(), KbLight::open(), and KbLight::shareDimming().

13.44.2.2 QSet<KbLight*> activeLights [static]

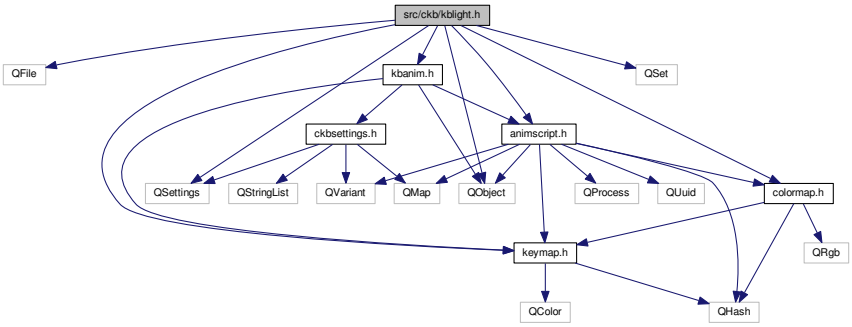
Definition at line 8 of file kblight.cpp.

Referenced by KbLight::close(), KbLight::open(), KbLight::shareDimming(), and KbLight::~KbLight().

13.45 src/ckb/kblight.h File Reference

```
#include <QFile>
#include <QObject>
#include <QSet>
#include <QSettings>
#include "animscript.h"
#include "kbanim.h"
#include "keymap.h"
#include "colormap.h"
```

Include dependency graph for kblight.h:



This graph shows which files directly or indirectly include this file:

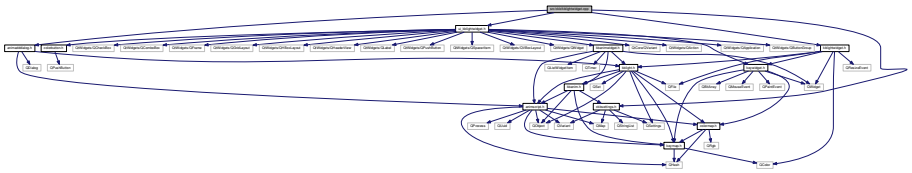
Data Structures

- class KbLight

13.46 src/ckb/kblightwidget.cpp File Reference

```
#include "animadddialog.h"
#include "ckbsettings.h"
#include "kblightwidget.h"
#include "ui_kblightwidget.h"
```

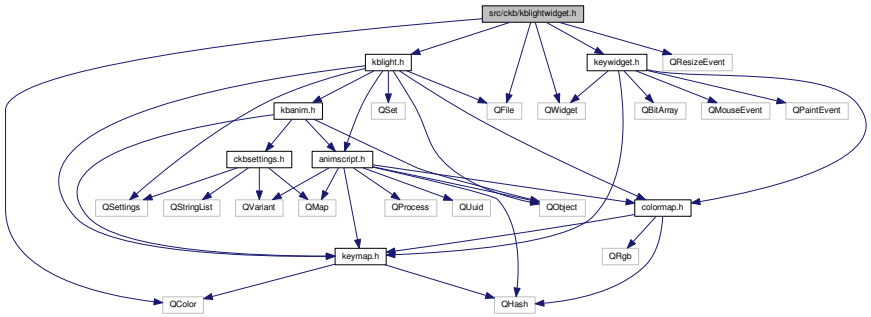
Include dependency graph for kblightwidget.cpp:



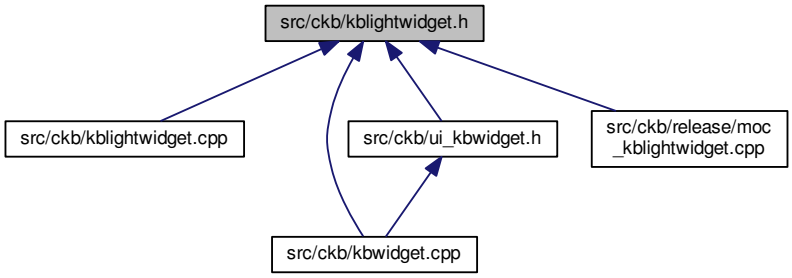
13.47 src/ckb/kblightwidget.h File Reference

```
#include <QWidget>
#include <QColor>
#include <QFile>
#include <QResizeEvent>
#include "kblight.h"
#include "keywidget.h"
```

Include dependency graph for kblightwidget.h:



This graph shows which files directly or indirectly include this file:



Data Structures

- class KbLightWidget

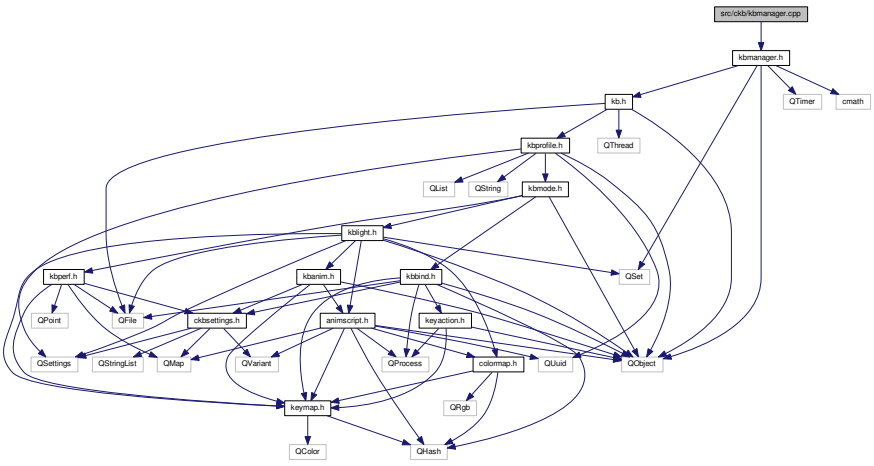
Namespaces

- Ui

13.48 src/ckb/kbmanager.cpp File Reference

```
#include "kbmanager.h"
```

Include dependency graph for kbmanager.cpp:



Variables

- QString devpath = "/dev/input/ckb%1"

13.48.1 Variable Documentation

13.48.1.1 QString devpath = "/dev/input/ckb%1"

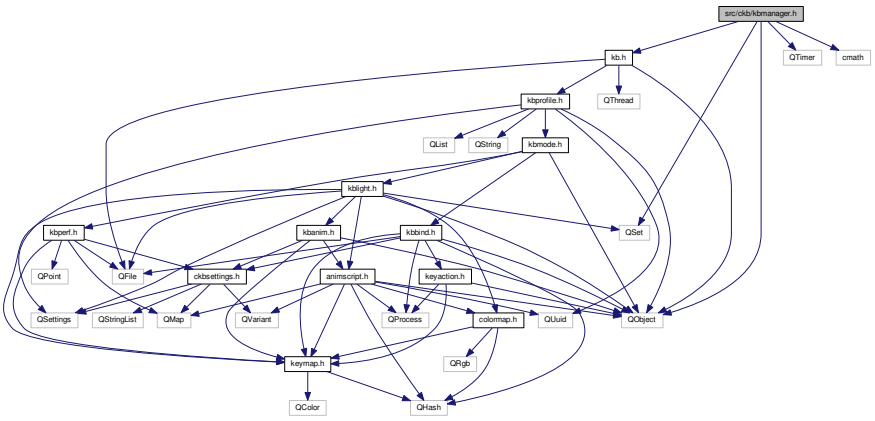
Definition at line 4 of file kbmanager.cpp.

Referenced by MainWindow::MainWindow(), and KbManager::scanKeyboards().

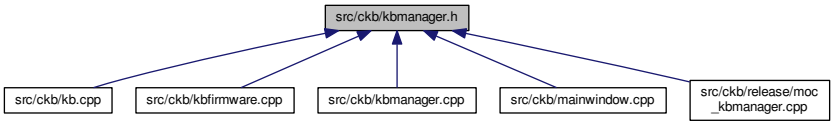
13.49 src/ckb/kbmanager.h File Reference

```
#include <QObject>
#include <QTimer>
#include <cmath>
#include <QSet>
#include "kb.h"
```

Include dependency graph for kbmanager.h:



This graph shows which files directly or indirectly include this file:



Data Structures

- class KbManager

Macros

- #define DAEMON_UNAVAILABLE_STR "<unavailable>"

13.49.1 Macro Definition Documentation

13.49.1.1 #define DAEMON_UNAVAILABLE_STR "<unavailable>"

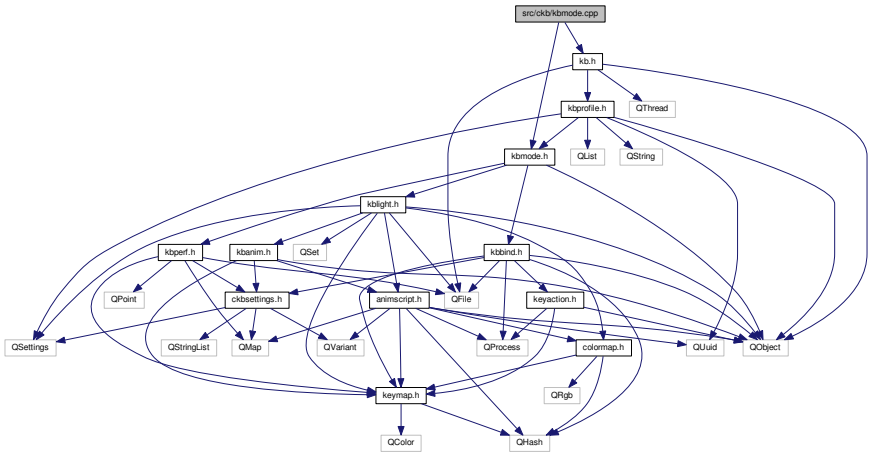
Definition at line 12 of file kbmanager.h.

Referenced by KbManager::ckbDaemonVersionF(), KbManager::scanKeyboards(), and MainWindow::updateVersion().

13.50 src/ckb/kbmode.cpp File Reference

```
#include "kbmode.h"
#include "kb.h"
```

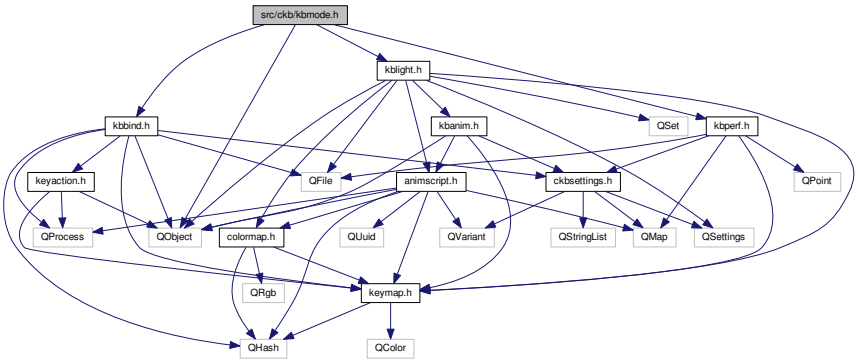
Include dependency graph for kbmode.cpp:



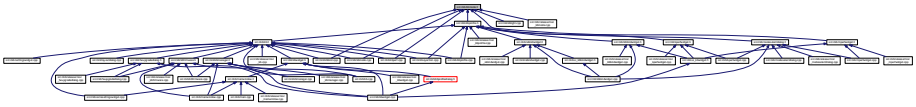
13.51 src/ckb/kbmode.h File Reference

```
#include <QObject>
#include "kblight.h"
#include "kbbind.h"
#include "kbperf.h"
```

Include dependency graph for kbmode.h:



This graph shows which files directly or indirectly include this file:



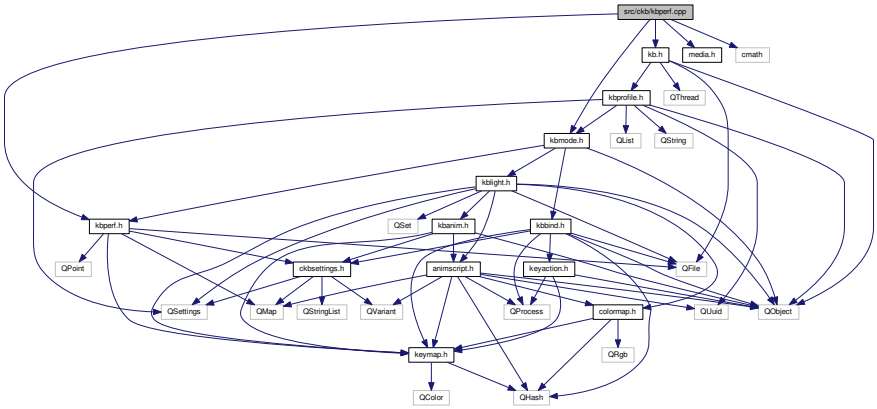
Data Structures

- struct UsbId
- class KbMode

13.52 src/ckb/kbperf.cpp File Reference

```
#include "kbperf.h"
#include "kbmode.h"
#include "kb.h"
#include "media.h"
#include <cmath>
```

Include dependency graph for kbperf.cpp:



Macros

- #define map_last(map) ((map).last())

13.52.1 Macro Definition Documentation

13.52.1.1 #define map_last(*map*) ((map).last())

Definition at line 8 of file kbperf.cpp.

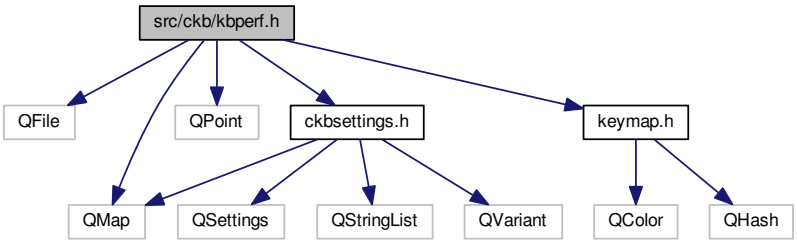
Referenced by KbPerf::popDpi().

13.53 src/ckb/kbperf.h File Reference

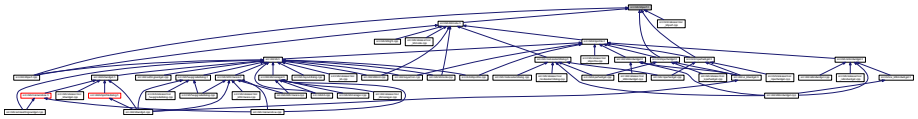
```
#include <QFile>
```

```
#include <QMap>
#include <QPoint>
#include "ckbsettings.h"
#include "keymap.h"
```

Include dependency graph for kbperf.h:



This graph shows which files directly or indirectly include this file:



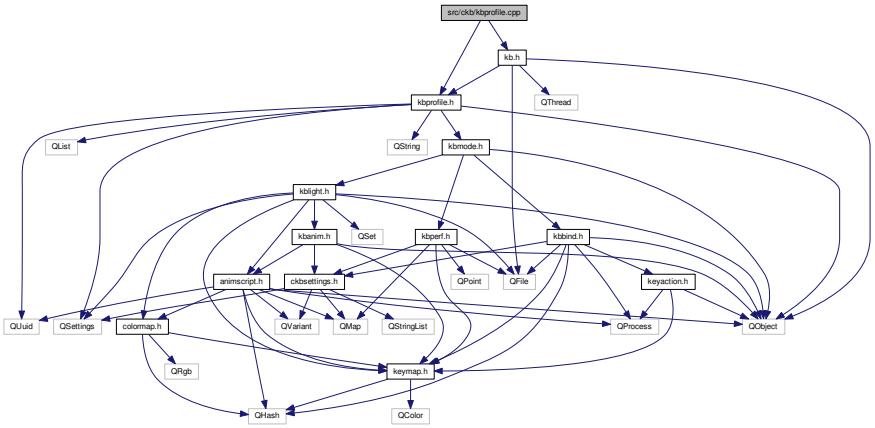
Data Structures

- class KbPerf

13.54 src/ckb/kbprofile.cpp File Reference

```
#include "kbprofile.h"
#include "kb.h"
```

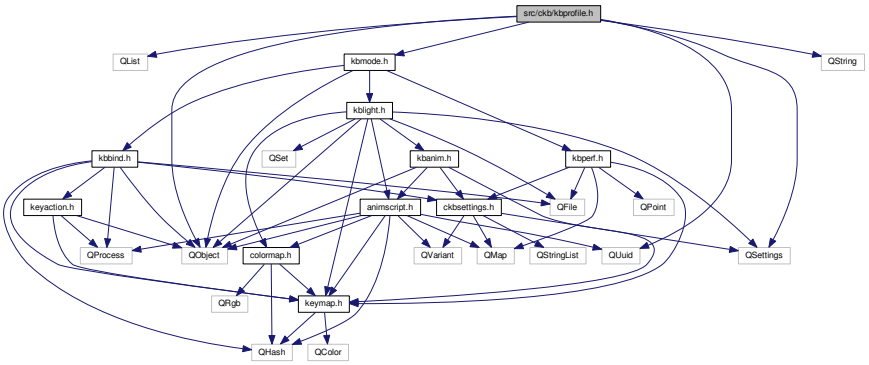
Include dependency graph for kbprofile.cpp:



13.55 src/ckb/kbprofile.h File Reference

```
#include <QList>
#include <QObject>
#include <QSettings>
#include <QString>
#include <QUuid>
#include "kbmode.h"
```

Include dependency graph for kbprofile.h:



This graph shows which files directly or indirectly include this file:

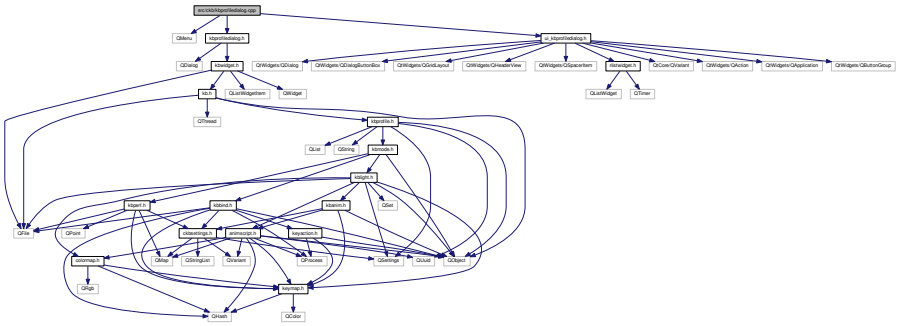
Data Structures

- class KbProfile

13.56 src/ckb/kbprofiledialog.cpp File Reference

```
#include <QMenu>
#include "kbprofiledialog.h"
#include "ui_kbprofiledialog.h"
```

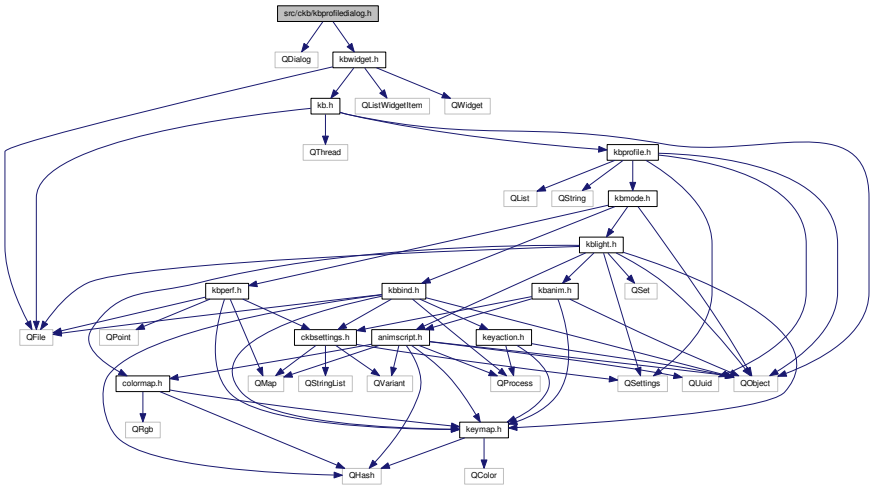
Include dependency graph for kbprofiledialog.cpp:



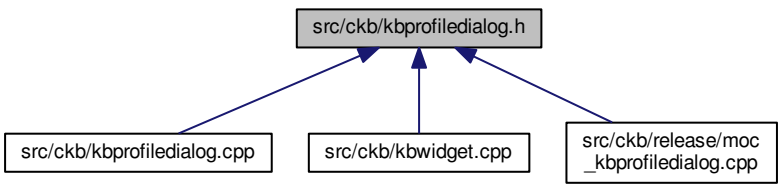
13.57 src/ckb/kbprofiledialog.h File Reference

```
#include <QDialog>
#include "kbwidget.h"
```

Include dependency graph for kbprofiledialog.h:



This graph shows which files directly or indirectly include this file:



Data Structures

- class KbProfileDialog

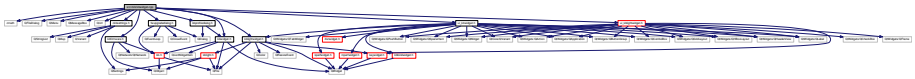
Namespaces

- Ui

13.58 src/ckb/kbwidget.cpp File Reference

```
#include <cmath>
#include <QFileDialog>
#include <QMenu>
#include <QMessageBox>
#include <QUrl>
#include "ckbsettings.h"
#include "fwupgradedialog.h"
#include "kbfirmware.h"
#include "kbwidget.h"
#include "kblightwidget.h"
#include "kbprofiledialog.h"
#include "ui_kbwidget.h"
#include "ui_kblightwidget.h"
```

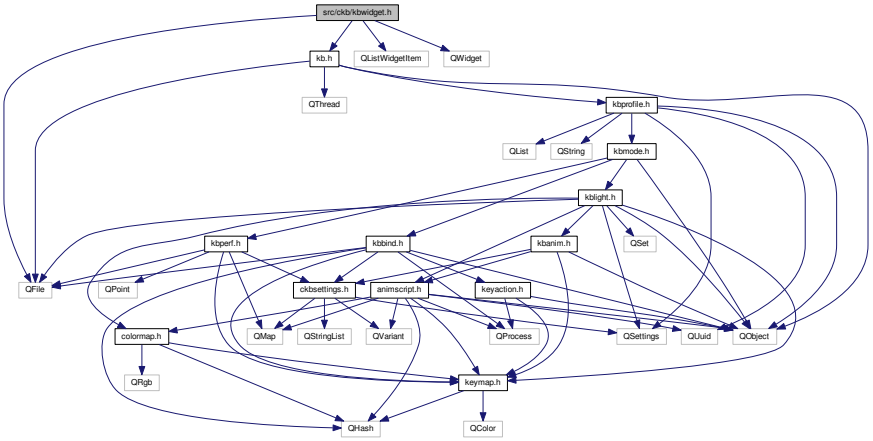
Include dependency graph for kbwidget.cpp:



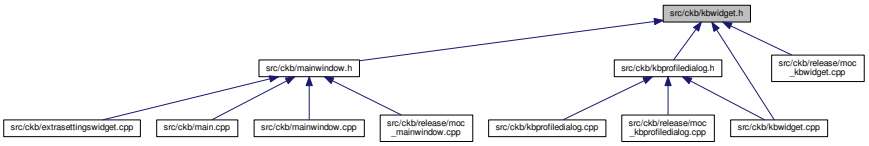
13.59 src/ckb/kbwidget.h File Reference

```
#include <QFile>
#include <QListWidgetItem>
#include <QWidget>
#include "kb.h"
```

Include dependency graph for kbwidget.h:



This graph shows which files directly or indirectly include this file:



Data Structures

- class KbWidget

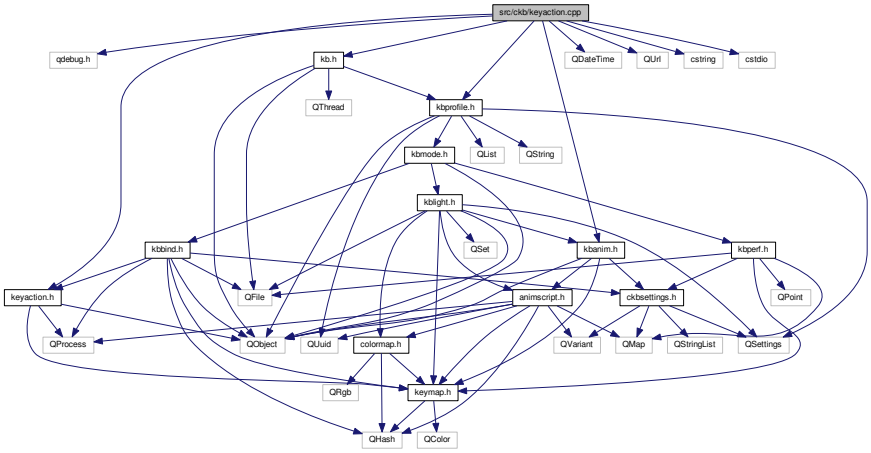
Namespaces

- Ui

13.60 src/ckb/keyaction.cpp File Reference

```
#include <qdebug.h>
#include "keyaction.h"
#include "kb.h"
#include "kbanim.h"
#include "kbprofile.h"
#include <QDateTime>
#include <QUrl>
#include <cstring>
#include <cstdio>
```

Include dependency graph for keyaction.cpp:



Variables

- static const int ANIM_ONCE = 0x1
- static const int ANIM_KRSTOP = 0x2

13.60.1 Variable Documentation

13.60.1.1 const int ANIM_KRSTOP = 0x2 [static]

Definition at line 182 of file keyaction.cpp.

Referenced by KeyAction::animAction(), and KeyAction::animInfo().

13.60.1.2 const int ANIM_ONCE = 0x1 [static]

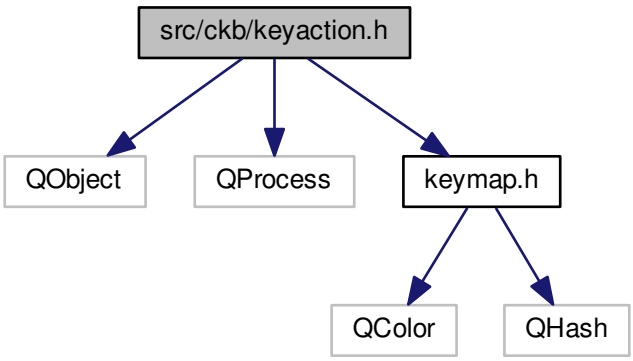
Definition at line 182 of file keyaction.cpp.

Referenced by KeyAction::animAction(), and KeyAction::animInfo().

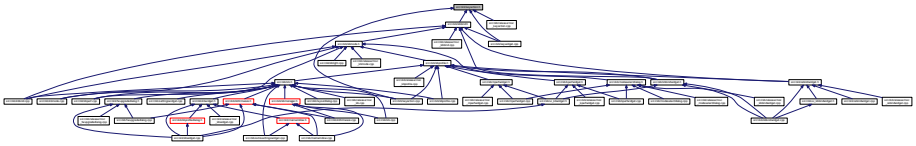
13.61 src/ckb/keyaction.h File Reference

```
#include <QObject>
#include <QProcess>
#include "keymap.h"
```

Include dependency graph for keyaction.h:



This graph shows which files directly or indirectly include this file:



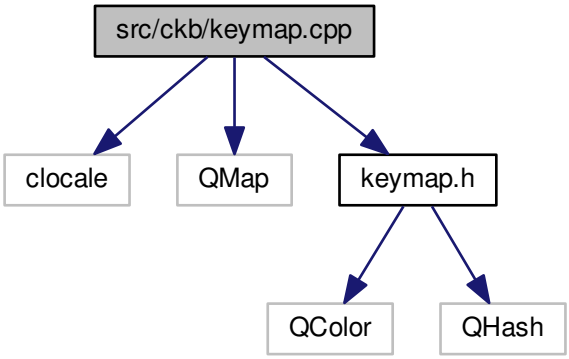
Data Structures

- class KeyAction

13.62 src/ckb/keymap.cpp File Reference

```
#include <clocale>
#include <QMap>
#include "keymap.h"
```

Include dependency graph for keymap.cpp:



Data Structures

- struct KeyPatch

Macros

- #define NS 12, 12
- #define LBS 17, 6
- #define KEYCOUNT_K95 (sizeof(K95Keys) / sizeof(Key))
- #define ANSI_ENTER_X 199
- #define ANSI_ENTER_Y 51
- #define ANSI_ENTER_W 26
- #define ANSI_ENTER_H 12
- #define ANSI_LSHIFT_X 45
- #define ANSI_LSHIFT_W 26
- #define PATCH_COUNT(patch) (sizeof(patch) / sizeof(KeyPatch))
- #define patch(map, patches) _patch(map, patches, PATCH_COUNT(patches))
- #define K95_WIDTH 298
- #define K95_HEIGHT 76
- #define K95P_HEIGHT 82
- #define K95P_X_START 20
- #define K95P_WIDTH (K95_WIDTH - K95P_X_START + 1)
- #define K70_X_START 38
- #define K70_WIDTH (K95_WIDTH - K70_X_START)
- #define K70_HEIGHT K95_HEIGHT
- #define K68_WIDTH K70_WIDTH
- #define K68_HEIGHT K70_HEIGHT
- #define K65_WIDTH 209

- #define K65_HEIGHT K70_HEIGHT
- #define K63_WIDTH K65_WIDTH
- #define K63_HEIGHT K65_HEIGHT
- #define K68_TOP_COUNT (sizeof(K68TopRow) / sizeof(Key))
- #define K65_TOP_COUNT (sizeof(K65TopRow) / sizeof(Key))
- #define K63_TOP_COUNT (sizeof(K63TopRow) / sizeof(Key))
- #define KSTRAFE_X_START 12
- #define KSTRAFE_WIDTH (K70_WIDTH + (KSTRAFE_X_START * 2))
- #define KSTRAFE_HEIGHT K95_HEIGHT
- #define KEYCOUNT_M65 (sizeof(M65Keys) / sizeof(Key))
- #define M65_WIDTH 52
- #define M65_HEIGHT 67
- #define KEYCOUNT_SABRE (sizeof(SabreKeys) / sizeof(Key))
- #define SABRE_WIDTH M65_WIDTH
- #define SABRE_HEIGHT M65_HEIGHT
- #define KEYCOUNT_HARPOON (sizeof(HarpoonKeys) / sizeof(Key))
- #define HARPOON_WIDTH M65_WIDTH
- #define HARPOON_HEIGHT M65_HEIGHT
- #define KEYCOUNT_GLAIVE (sizeof(GlaiveKeys) / sizeof(Key))
- #define GLAIVE_WIDTH M65_WIDTH
- #define GLAIVE_HEIGHT M65_HEIGHT
- #define KEYCOUNT_SCIM (sizeof(ScimKeys) / sizeof(Key))
- #define SCIM_WIDTH M65_WIDTH
- #define SCIM_HEIGHT M65_HEIGHT
- #define LBARCOUNT_K95P (sizeof(K95PLbar) / sizeof(Key))
- #define N_MODELS KeyMap::_MODEL_MAX
- #define N_LAYOUTS KeyMap::_LAYOUT_MAX

Functions

- static void _patch (QHash< QString, Key > &map, const KeyPatch *patches, int patchCount)
- static void patchnonJP106 (QHash< QString, Key > &map)
- static void patchISO (QHash< QString, Key > &map)
- static void patchANSI (QHash< QString, Key > &map)
- static void patchJP106 (QHash< QString, Key > &map)
- static void patchJP106fn (QHash< QString, Key > &map)
- static void patchABNT2 (QHash< QString, Key > &map)
- static QHash< QString, Key > getMap (KeyMap::Model model, KeyMap::Layout layout)

Variables

- static const Key K95Keys []
- static const KeyPatch patchDK []
- static const KeyPatch patchEU []
- static const KeyPatch patchFR []
- static const KeyPatch patchDE []
- static const KeyPatch patchIT []
- static const KeyPatch patchNO []
- static const KeyPatch patchMX []
- static const KeyPatch patchPTBR []
- static const KeyPatch patchES []
- static const KeyPatch patchSE []
- static const KeyPatch patchJP []
- static const KeyPatch patchDvorak []

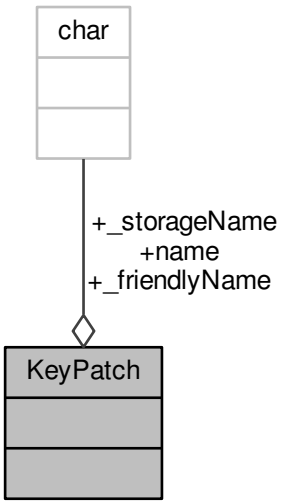
- static const Key K68TopRow []
- static const Key K65TopRow []
- static const Key K63TopRow []
- static const Key KStrafeKeys []
- static const Key M65Keys []
- static const Key SabreKeys []
- static const Key HarpoonKeys []
- static const Key GlaiveKeys []
- static const Key ScimKeys []
- static const Key K95PLbar []
- static QHash< QString, Key > K95BaseMap
- static QHash< QString, Key > standardMaps [KeyMap::_MODEL_MAX][KeyMap::_LAYOUT_MAX]

13.62.1 Data Structure Documentation

13.62.1.1 struct KeyPatch

Definition at line 32 of file keymap.cpp.

Collaboration diagram for KeyPatch:



Data Fields

const char *	_friendlyName	
const char *	_storageName	
const char *	name	

13.62.2 Macro Definition Documentation

13.62.2.1 #define ANSI_ENTER_H 12

Definition at line 27 of file keymap.cpp.

Referenced by patchANSI().

13.62.2.2 #define ANSI_ENTER_W 26

Definition at line 26 of file keymap.cpp.

Referenced by patchANSI().

13.62.2.3 #define ANSI_ENTER_X 199

Definition at line 24 of file keymap.cpp.

Referenced by patchANSI().

13.62.2.4 #define ANSI_ENTER_Y 51

Definition at line 25 of file keymap.cpp.

Referenced by patchANSI().

13.62.2.5 #define ANSI_LSHIFT_W 26

Definition at line 29 of file keymap.cpp.

Referenced by patchANSI(), and patchJP106().

13.62.2.6 #define ANSI_LSHIFT_X 45

Definition at line 28 of file keymap.cpp.

Referenced by patchANSI(), and patchJP106().

13.62.2.7 #define GLAIVE_HEIGHT M65_HEIGHT

Definition at line 310 of file keymap.cpp.

13.62.2.8 #define GLAIVE_WIDTH M65_WIDTH

Definition at line 309 of file keymap.cpp.

13.62.2.9 #define HARPOON_HEIGHT M65_HEIGHT

Definition at line 298 of file keymap.cpp.

13.62.2.10 #define HARPOON_WIDTH M65_WIDTH

Definition at line 297 of file keymap.cpp.

13.62.2.11 #define K63_HEIGHT K65_HEIGHT

Definition at line 233 of file keymap.cpp.

13.62.2.12 #define K63_TOP_COUNT (sizeof(K63TopRow) / sizeof(Key))

Definition at line 248 of file keymap.cpp.

Referenced by getMap().

13.62.2.13 #define K63_WIDTH K65_WIDTH

Definition at line 232 of file keymap.cpp.

Referenced by KeyMap::modelWidth().

13.62.2.14 #define K65_HEIGHT K70_HEIGHT

Definition at line 229 of file keymap.cpp.

13.62.2.15 #define K65_TOP_COUNT (sizeof(K65TopRow) / sizeof(Key))

Definition at line 243 of file keymap.cpp.

Referenced by getMap().

13.62.2.16 #define K65_WIDTH 209

Definition at line 228 of file keymap.cpp.

Referenced by getMap(), and KeyMap::modelWidth().

13.62.2.17 #define K68_HEIGHT K70_HEIGHT

Definition at line 225 of file keymap.cpp.

13.62.2.18 #define K68_TOP_COUNT (sizeof(K68TopRow) / sizeof(Key))

Definition at line 238 of file keymap.cpp.

Referenced by getMap().

13.62.2.19 #define K68_WIDTH K70_WIDTH

Definition at line 224 of file keymap.cpp.

Referenced by KeyMap::modelWidth().

13.62.2.20 #define K70_HEIGHT K95_HEIGHT

Definition at line 222 of file keymap.cpp.

13.62.2.21 #define K70_WIDTH (K95_WIDTH - K70_X_START)

Definition at line 221 of file keymap.cpp.

Referenced by KeyMap::modelWidth().

13.62.2.22 #define K70_X_START 38

Definition at line 220 of file keymap.cpp.

Referenced by getMap().

13.62.2.23 #define K95_HEIGHT 76

Definition at line 213 of file keymap.cpp.

Referenced by KeyMap::modelHeight().

13.62.2.24 #define K95_WIDTH 298

Definition at line 212 of file keymap.cpp.

Referenced by KeyMap::modelWidth().

13.62.2.25 #define K95P_HEIGHT 82

Definition at line 215 of file keymap.cpp.

Referenced by KeyMap::modelHeight().

13.62.2.26 #define K95P_WIDTH (K95_WIDTH - K95P_X_START + 1)

Definition at line 217 of file keymap.cpp.

Referenced by KeyMap::modelWidth().

13.62.2.27 #define K95P_X_START 20

Definition at line 216 of file keymap.cpp.

Referenced by getMap().

13.62.2.28 #define KEYCOUNT_GLAIVE (sizeof(GlaiveKeys) / sizeof(Key))

Definition at line 307 of file keymap.cpp.

Referenced by getMap().

13.62.2.29 #define KEYCOUNT_HARPOON (sizeof(HarpoonKeys) / sizeof(Key))

Definition at line 295 of file keymap.cpp.

Referenced by getMap().

13.62.2.30 #define KEYCOUNT_K95 (sizeof(K95Keys) / sizeof(Key))

Definition at line 21 of file keymap.cpp.

Referenced by getMap().

13.62.2.31 #define KEYCOUNT_M65 (sizeof(M65Keys) / sizeof(Key))

Definition at line 270 of file keymap.cpp.

Referenced by getMap().

13.62.2.32 #define KEYCOUNT_SABRE (sizeof(SabreKeys) / sizeof(Key))

Definition at line 283 of file keymap.cpp.

Referenced by getMap().

13.62.2.33 #define KEYCOUNT_SCIM (sizeof(ScimKeys) / sizeof(Key))

Definition at line 324 of file keymap.cpp.

Referenced by getMap().

13.62.2.34 #define KSTRAFE_HEIGHT K95_HEIGHT

Definition at line 253 of file keymap.cpp.

13.62.2.35 #define KSTRAFE_WIDTH (K70_WIDTH + (KSTRAFE_X_START * 2))

Definition at line 252 of file keymap.cpp.

Referenced by KeyMap::modelWidth().

13.62.2.36 #define KSTRAFE_X_START 12

Definition at line 251 of file keymap.cpp.

Referenced by getMap().

13.62.2.37 #define LBARCOUNT_K95P (sizeof(K95PLbar) / sizeof(Key))

Definition at line 336 of file keymap.cpp.

Referenced by getMap().

13.62.2.38 #define LBS 17, 6

Definition at line 9 of file keymap.cpp.

13.62.2.39 #define M65_HEIGHT 67

Definition at line 273 of file keymap.cpp.

Referenced by KeyMap::modelHeight().

13.62.2.40 #define M65_WIDTH 52

Definition at line 272 of file keymap.cpp.

Referenced by KeyMap::modelWidth().

13.62.2.41 #define N_LAYOUTS KeyMap::_LAYOUT_MAX

Definition at line 341 of file keymap.cpp.

Referenced by getMap().

13.62.2.42 #define N_MODELS KeyMap::_MODEL_MAX

Definition at line 340 of file keymap.cpp.

Referenced by getMap().

13.62.2.43 #define NS 12, 12

Definition at line 6 of file keymap.cpp.

13.62.2.44 #define patch(*map, patches*) _patch(map, patches, PATCH_COUNT(patches))

Definition at line 120 of file keymap.cpp.

Referenced by getMap().

13.62.2.45 #define PATCH_COUNT(patch) (sizeof(patch) / sizeof(KeyPatch))

Definition at line 119 of file keymap.cpp.

13.62.2.46 #define SABRE_HEIGHT M65_HEIGHT

Definition at line 286 of file keymap.cpp.

13.62.2.47 #define SABRE_WIDTH M65_WIDTH

Definition at line 285 of file keymap.cpp.

13.62.2.48 #define SCIM_HEIGHT M65_HEIGHT

Definition at line 327 of file keymap.cpp.

13.62.2.49 #define SCIM_WIDTH M65_WIDTH

Definition at line 326 of file keymap.cpp.

13.62.3 Function Documentation

13.62.3.1 static void _patch (QHash< QString, Key > & *map,* const KeyPatch * *patches,* int *patchCount*) [static]

Definition at line 121 of file keymap.cpp.

References Key::_friendlyName, and Key::_storageName.

```
121                                                                                                 {
122     for(const KeyPatch* p = patches; p < patches + patchCount; p++){
123         Key& key = map[p->name];
124         key._storageName = p->_storageName;
```

```
125         key._friendlyName = p->_friendlyName;
126     }
127 }
```

13.62.3.2 static QHash<QString, Key> getMap (KeyMap::Model *model,* KeyMap::Layout *layout*) [static]

Definition at line 344 of file keymap.cpp.

References KeyMap::DE, KeyMap::DK, KeyMap::ES, KeyMap::EU, KeyMap::EU_DVORAK, KeyMap::FR, KeyMap::GB_DVORAK, KeyMap::GLAIVE, KeyMap::HARPOON, Key::height, KeyMap::isISO(), KeyMap::isJP(), KeyMap::isPTBR(), KeyMap::IT, KeyMap::JP, KeyMap::K63, K63_TOP_COUNT, KeyMap::K65, K65_TOP_COUNT, K65_WIDTH, KeyMap::K68, K68_TOP_COUNT, KeyMap::K70, K70_X_START, KeyMap::K95, K95BaseMap, KeyMap::K95P, K95P_X_START, KEYCOUNT_GLAIVE, KEYCOUNT_HARPOON, KEYCOUNT_K95, KEYCOUNT_M65, KEYCOUNT_SABRE, KEYCOUNT_SCIM, KSTRAFE_X_START, LBARCOUNT_K95P, KeyMap::M65, KeyMap::MX, N_LAYOUTS, N_MODELS, KeyMap::NO, patch, patchABNT2(), patchANSI(), patchISO(), patchJP106(), patchJP106fn(), KeyMap::PT_BR, KeyMap::SABRE, KeyMap::SCIMITAR, KeyMap::SE, standardMaps, KeyMap::STRAFE, KeyMap::US_DVORAK, Key::width, Key::x, and Key::y.

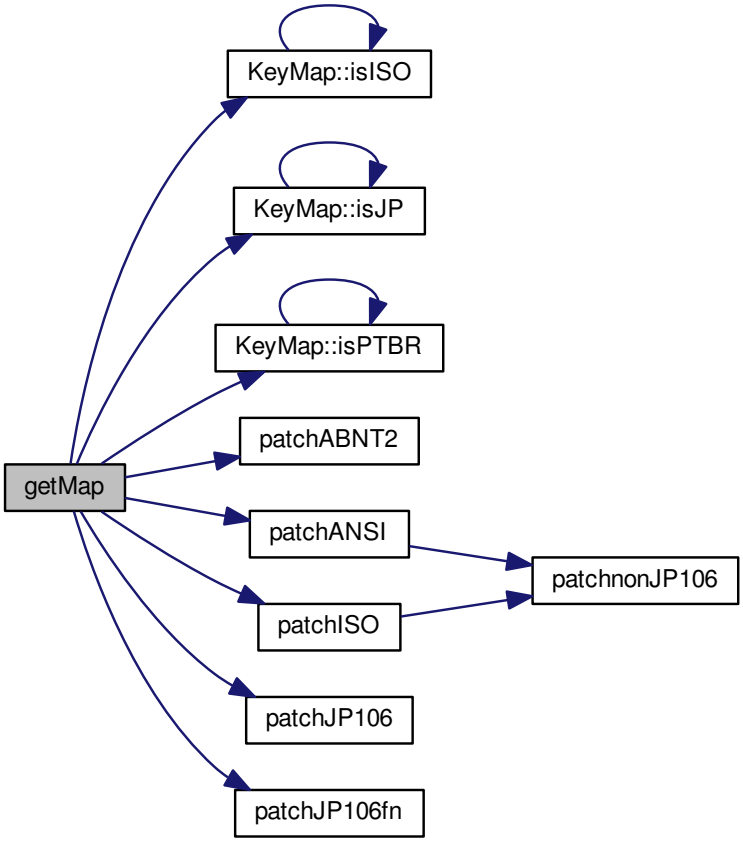
```
344                                                                     {
345     if(model < 0 || layout < 0 || model >= N_MODELS || layout >=
    N_LAYOUTS)
346         return QHash<QString, Key>();
347     // Return the map if it's already filled out
348     QHash<QString, Key>& map = standardMaps[model][layout];
349     if(!map.empty())
350         return map;
351     // Otherwise, create it
352     switch(model){
353     case KeyMap::K95:{
354         // The K95 maps serve as bases for all the other keyboards
355         // Fetch the master map, or create it if not yet done
356         if(K95BaseMap.empty()){
357             for(const Key* key = K95Keys; key < K95Keys +
    KEYCOUNT_K95; key++)
358                 K95BaseMap[key->name] = *key;
359         }
360         map = K95BaseMap;
361         // Patch the map for the layout
362         switch(layout){
363         case KeyMap::DK:
364             patch(map, patchDK);
365             break;
366         case KeyMap::EU_DVORAK:
367             patch(map, patchDvorak);    // fall through
368         case KeyMap::EU:
369             patch(map, patchEU);
370             break;
371         case KeyMap::GB_DVORAK:
372         case KeyMap::US_DVORAK:
373             patch(map, patchDvorak);
374             break;
375         case KeyMap::FR:
376             patch(map, patchFR);
377             break;
378         case KeyMap::DE:
379             patch(map, patchDE);
380             break;
381         case KeyMap::IT:
382             patch(map, patchIT);
383             break;
384         case KeyMap::NO:
385             patch(map, patchNO);
386             break;
387         case KeyMap::MX:
388             patch(map, patchMX);
389             break;
390         case KeyMap::PT_BR:
391             patch(map, patchPTBR);
392             break;
393         case KeyMap::ES:
394             patch(map, patchES);
395             break;
396         case KeyMap::SE:
397             patch(map, patchSE);
398             break;
399         case KeyMap::JP:
400             patch(map, patchJP);
401             break;
```

```
402         default:;
403             // English QWERTY - no patch needed
404         }
405         if(KeyMap::isJP(layout))
406             patchJP106(map);
407         else if(KeyMap::isPTBR(layout))
408             patchABNT2(map);
409         else if(KeyMap::isISO(layout))
410             patchISO(map);
411         else
412             patchANSI(map);
413         // Done! return the map
414         break;
415     }
416     case KeyMap::K95P:{
417         // The K95 Platinum map is based on the K95
418         map = getMap(KeyMap::K95, layout);
419         // Remove excess G keys
420         map.remove("g7");
421         map.remove("g8");
422         map.remove("g9");
423         map.remove("g10");
424         map.remove("g11");
425         map.remove("g12");
426         map.remove("g13");
427         map.remove("g14");
428         map.remove("g15");
429         map.remove("g16");
430         map.remove("g17");
431         map.remove("g18");
432         // Place the remaining G keys vertically
433         map["g1"].x = 22;
434         map["g2"].x = 22;
435         map["g3"].x = 22;
436         map["g4"].x = 22;
437         map["g5"].x = 22;
438         map["g6"].x = 22;
439         //map["g1"].y = 14+(12*0);
440         map["g2"].y = 26; //14+(12*1)
441         map["g3"].y = 38; //14+(12*2)
442         map["g4"].y = 50; //14+(12*3)
443         map["g5"].y = 62; //14+(12*4)
444         map["g6"].y = 74; //14+(12*5)
445         // Remove M keys
446         map.remove("m1");
447         map.remove("m2");
448         map.remove("m3");
449         // Centre MR/Brightness/Winlock between F1 and F4
450         map["mr"].x = 64;
451         map["light"].x = 75;
452         map["lock"].x = 86;
453         // Resize them
454         map["mr"].height = 8;
455         map["light"].height = 8;
456         map["lock"].height = 8;
457         map["mr"].width = 11;
458         map["light"].width = 11;
459         map["lock"].width = 11;
460         // Mute is above Stop
461         map["mute"].x -= 11;
462         map["voldn"].x -= 11;
463         map["volup"].x -= 11;
464         // Move the buttons on the top a bit further down to centre them
465         map["mr"].y += 1;
466         map["light"].y += 1;
467         map["lock"].y += 1;
468         map["mute"].y += 1;
469         map["voldn"].y += 1;
470         map["volup"].y += 1;
471         // Shift all keys down (to make room for the lightbar), and to the left
472         QMutableHashIterator<QString, Key> i(map);
473         while(i.hasNext()){
474             i.next();
475             i.value().x -= K95P_X_START;
476             i.value().y += 6;
477         }
478         // Add lightbar
479         for(const Key* key = K95PLbar; key < K95PLbar +
    LBARCOUNT_K95P; key++)
480             map.insert(key->name, *key);
481
482         break;
483     }
484     case KeyMap::K70:{
485         // The K70 maps are based on the K95 maps. However all the keys are shifted left and the G keys are
     removed
486         map = getMap(KeyMap::K95, layout);
```

```
487         QMutableHashIterator<QString, Key> i(map);
488         while(i.hasNext()){
489             i.next();
490             // Move key to left. Remove it if it fell off the edge
491             if((i.value().x -= K70_X_START) < 0)
492                 i.remove();
493         }
494         // Remove the M buttons as well
495         map.remove("mr");
496         map.remove("m1");
497         map.remove("m2");
498         map.remove("m3");
499         // Done!
500         break;
501     }
502     case KeyMap::K68:{
503         // Same as the K70, except volume up and down keys
504         map = getMap(KeyMap::K70, layout);
505         for(const Key* key = K68TopRow; key < K68TopRow +
    K68_TOP_COUNT; key++)
506             map[key->name] = *key;
507 
508         break;
509     }
510     case KeyMap::K65:{
511         // The K65 maps additionally remove the numpad and have a modified top row
512         map = getMap(KeyMap::K70, layout);
513         QMutableHashIterator<QString, Key> i(map);
514         while(i.hasNext()){
515             i.next();
516             if(i.value().x >= K65_WIDTH)
517                 i.remove();
518         }
519         for(const Key* key = K65TopRow; key < K65TopRow +
    K65_TOP_COUNT; key++)
520             map[key->name] = *key;
521 
522         map.remove("rwin");
523         map["fn"] = KStrafeKeys[3];
524         map["fn"].x -= 12;
525 
526         if(KeyMap::isJP(layout))
527             patchJP106fn(map);
528 
529         // Done!
530         break;
531     }
532     case KeyMap::K63:{
533         // Same as the K65 but without the Fn key
534         map = getMap(KeyMap::K70, layout);
535         QMutableHashIterator<QString, Key> i(map);
536         while(i.hasNext()){
537             i.next();
538             if(i.value().x >= K65_WIDTH)
539                 i.remove();
540         }
541         for(const Key* key = K63TopRow; key < K63TopRow +
    K63_TOP_COUNT; key++)
542             map[key->name] = *key;
543 
544         break;
545     }
546     case KeyMap::STRAFE:{
547         // The Strafe RGB maps are based on the K70 map minus the media keys
548         map = getMap(KeyMap::K70, layout);
549         //move light and lock right
550         map["light"].x=285 - K70_X_START;
551         //map["light"].hasLed=false;
552         map["lock"].x=297 - K70_X_START;
553         //map["lock"].hasLed=false;
554         // move everything right to make the space for the left sidelight
555         QMutableHashIterator<QString, Key> i(map);
556         while(i.hasNext()){
557             i.next();
558             i.value().x += KSTRAFE_X_START;
559         }
560         // Add Strafe lights and keys
561         map["lsidel"] = KStrafeKeys[0];
562         map["rsidel"] = KStrafeKeys[1];
563         map["logo"] = KStrafeKeys[2];
564         map["fn"] = KStrafeKeys[3];
565         map.remove("rwin");
566         // remove media controls
567         map.remove("mute");
568         map.remove("volup");
569         map.remove("voldn");
570         map.remove("stop");
```

```
571         map.remove("prev");
572         map.remove("play");
573         map.remove("next");
574 
575         if(KeyMap::isJP(layout))
576             patchJP106fn(map);
577 
578         // Done!
579         break;
580     }
581     case KeyMap::M65:{
582         // M65 isn't a keyboard; all mouse maps are unique.
583         for(const Key* key = M65Keys; key < M65Keys +
    KEYCOUNT_M65; key++){
584             // Keyboard keys are written from the center because that's where the LEDs are, but the mouse
     buttons are odd shapes so they're
585             // written from the upper left
586             Key translatedKey = *key;
587             translatedKey.x += translatedKey.width / 2;
588             translatedKey.y += translatedKey.height / 2;
589             map[key->name] = translatedKey;
590         }
591         // Mice also have no layout patches - no other changes necessary
592         break;
593     }
594     case KeyMap::SABRE:{
595         // Sabre mouse
596         for(const Key* key = SabreKeys; key < SabreKeys +
    KEYCOUNT_SABRE; key++){
597             // Like the M65, the keys are upper-left justified
598             Key translatedKey = *key;
599             translatedKey.x += translatedKey.width / 2;
600             translatedKey.y += translatedKey.height / 2;
601             map[key->name] = translatedKey;
602         }
603         break;
604     }
605     case KeyMap::SCIMITAR:{
606         // Scimitar mouse
607         for(const Key* key = ScimKeys; key < ScimKeys +
    KEYCOUNT_SCIM; key++){
608             Key translatedKey = *key;
609             translatedKey.x += translatedKey.width / 2;
610             translatedKey.y += translatedKey.height / 2;
611             map[key->name] = translatedKey;
612         }
613         break;
614     }
615     case KeyMap::HARPOON:{
616         // Harpoon mouse
617         for(const Key* key = HarpoonKeys; key < HarpoonKeys +
    KEYCOUNT_HARPOON; key++){
618             Key translatedKey = *key;
619             translatedKey.x += translatedKey.width / 2;
620             translatedKey.y += translatedKey.height / 2;
621             map[key->name] = translatedKey;
622         }
623         break;
624     }
625     case KeyMap::GLAIVE:{
626         // Glaive mouse
627         for(const Key* key = GlaiveKeys; key < GlaiveKeys +
    KEYCOUNT_GLAIVE; key++){
628             Key translatedKey = *key;
629             translatedKey.x += translatedKey.width / 2;
630             translatedKey.y += translatedKey.height / 2;
631             map[key->name] = translatedKey;
632         }
633         break;
634     }
635     default:;    // <- stop GCC from complaining
636     }
637     // Map is finished, return result
638     return map;
639 }
```

Here is the call graph for this function:



13.62.3.3 static void patchABNT2 (QHash< QString, Key > & *map*) [static]

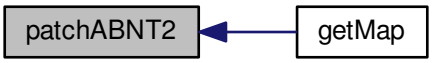
Definition at line 198 of file keymap.cpp.

References Key::width, and Key::x.

Referenced by getMap().

```
198                                                 {
199     map.remove("yen");
200     map.remove("henkan");
201     map.remove("muhenkan");
202     map.remove("katahira");
203     map.remove("bslash");
204
205     // Resize RShift
206     Key& rshift = map["rshift"];
207     rshift.width -= 12;
208     rshift.x += 6;
209 }
```

Here is the caller graph for this function:



13.62.3.4 static void patchANSI (QHash< QString, Key > & *map*) [static]

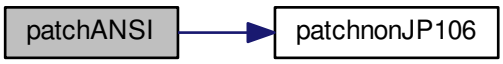
Definition at line 141 of file keymap.cpp.

References ANSI_ENTER_H, ANSI_ENTER_W, ANSI_ENTER_X, ANSI_ENTER_Y, ANSI_LSHIFT_W, ANSI_LSHIFT_X, Key::height, patchnonJP106(), Key::width, Key::x, and Key::y.

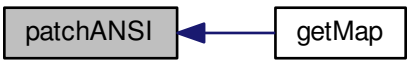
Referenced by getMap().

```
141                                                 {
142     patchnonJP106(map);
143     map.remove("bslash_iso");
144     map.remove("hash");
145     Key& enter = map["enter"];
146     enter.x = ANSI_ENTER_X;
147     enter.y = ANSI_ENTER_Y;
148     enter.width = ANSI_ENTER_W;
149     enter.height = ANSI_ENTER_H;
150     Key& lshift = map["lshift"];
151     lshift.x = ANSI_LSHIFT_X;
152     lshift.width = ANSI_LSHIFT_W;
153 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



13.62.3.5 static void patchISO (QHash< QString, Key > & *map*) [static]

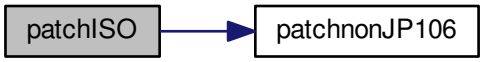
Definition at line 137 of file keymap.cpp.

References patchnonJP106().

Referenced by getMap().

```
137                                                 {
138     patchnonJP106(map);
139     map.remove("bslash");
140 }
```

Here is the call graph for this function:



Here is the caller graph for this function:

13.62.3.6 static void patchJP106 (QHash< QString, Key > & *map*) [static]

Definition at line 154 of file keymap.cpp.

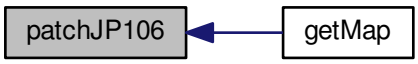
References ANSI_LSHIFT_W, ANSI_LSHIFT_X, Key::width, and Key::x.

Referenced by getMap().

```
154                                                   {
155     // First apply the ISO patch
156     map.remove("bslash");
157
158     // Resize Backspace
159     Key& bspace = map["bspace"];
160     bspace.width -= 12;
161     bspace.x = 206;
162
163     // Resize RShift
164     Key& rshift = map["rshift"];
165     rshift.width -= 12;
166     rshift.x += 6;
167
168     // Resize Spacebar
169     Key& space = map["space"];
170     space.width -= 26;
171     space.x -= 1;
172
173     // Left shift
```

```
174     Key& lshift = map["lshift"];
175     lshift.x = ANSI_LSHIFT_X;
176     lshift.width = ANSI_LSHIFT_W;
177 
178     map.remove("bslash_iso");
179 
180     // Resize and move ralt to make space for the extra keys
181     Key& ralt = map["ralt"];
182     ralt.x += 11;
183     ralt.width += 2;
184 
185     // None of these layouts have rwin
186     map.remove("rwin");
187 }
```

Here is the caller graph for this function:



13.62.3.7 static void patchJP106fn (QHash< QString, Key > & *map*) [static]

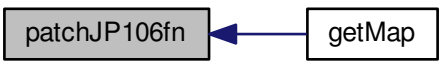
Definition at line 189 of file keymap.cpp.

References Key::width, and Key::x.

Referenced by getMap().

```
189                                                 {
190     if(map.contains("fn")){
191         map.remove("ralt");
192         Key& fn = map["fn"];
193         fn.width += 4;
194         fn.x -= 2;
195     }
196 }
```

Here is the caller graph for this function:



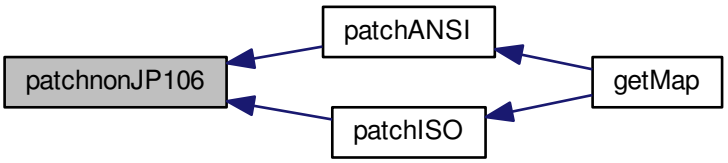
13.62.3.8 static void patchnonJP106 (QHash< QString, Key > & *map*) [static]

Definition at line 130 of file keymap.cpp.

Referenced by patchANSI(), and patchISO().

```
130                                                 {
131     map.remove("yen");
132     map.remove("henkan");
133     map.remove("muhenkan");
134     map.remove("katahira");
135     map.remove("ro");
136 }
```

Here is the caller graph for this function:



13.62.4 Variable Documentation

13.62.4.1 const Key GlaiveKeys[] [static]

Initial value:

```
= {
    {0, "Left Mouse", "mouse1", 17, 3, 14, 18, false, true}, {0, "Right Mouse", "mouse2", 37, 3, 14, 18,
      false, true}, {0, "Middle Mouse", "mouse3", 31, 9, 7, 7, false, true}, {0, "Front light", "front", 16, -5, 36,
       8, true, false },
    {0, "Wheel Up", "wheelup", 31, 5, 7, 5, false, true}, {0, "Wheel Down", "wheeldn", 31, 15, 7, 5, false,
       true}, {0, "Side Lights", "side", 22, 24, 7, 20, true, false},
    {0, "DPI Cycle", "dpiup", 31, 19, 6, 12, false, true}, {0, "Logo Light", "back", 24, 43, 20, 20, true,
      false},
    {0, "Forward", "mouse5", 15, 22, 5, 11, false, true}, {0, "Back", "mouse4", 15, 32, 5, 11, false, true}
    }
```

Definition at line 301 of file keymap.cpp.

13.62.4.2 const Key HarpoonKeys[] [static]

Initial value:

```
= {
    {0, "Left Mouse", "mouse1", 10, 5, 14, 26, false, true}, {0, "Right Mouse", "mouse2", 30, 5, 14, 26,
      false, true}, {0, "Middle Mouse", "mouse3", 25, 11, 6, 7, false, true},
    {0, "Wheel Up", "wheelup", 25, 7, 6, 5, false, true}, {0, "Wheel Down", "wheeldn", 25, 17, 6, 5, false,
       true},
    {0, "DPI Cycle", "dpiup", 25, 23, 6, 10, false, true}, {0, "Logo Light", "dpi", 17, 40, 20, 20, true,
      false},
    {0, "Forward", "mouse5", 3, 24, 5, 10, false, true}, {0, "Back", "mouse4", 3, 33, 5, 10, false, true}
    }
```

Definition at line 289 of file keymap.cpp.

13.62.4.3 const Key K63TopRow[] [static]

Initial value:

```
= {
    {0, "Stop", "stop",  38  - 37, 0, 12, 8, true, true}, {0, "Previous", "prev",  38  - 26, 0, 12, 8, true
      , true}, {0, "Play/Pause", "play",  38  - 15, 0, 12, 8, true, true}, {0, "Next", "next",  38  - 4, 0, 12, 8,
       true, true}, {0, "Brightness", "light", 170 -  38 , 0, 12, 12, true, true}, {0, "Windows Lock", "lock", 180
       -  38 , 0, 12, 12, true, true}, {0, "Mute", "mute", 222 -  38 , 0, 13, 8, true, true}, {0, "Volume Down", "
      voldn", 234 -  38 , 0, 13, 8, true, true}, {0, "Volume Up", "volup", 246 -  38 , 0, 13, 8, true, true}
}
```

Definition at line 245 of file keymap.cpp.

13.62.4.4 const Key K65TopRow[] [static]

Initial value:

```
= {
    {0, "Brightness", "light", 164 -  38 , 0, 12, 12, true, true}, {0, "Mute", "mute", 176 -  38 , 0, 12,
      12, true, true}, {0, "Volume Down", "voldn", 192 -  38 , 0, 14, 8, true, true}, {0, "Volume Up", "volup", 205
       -  38 , 0, 14, 8, true, true}, {0, "Windows Lock", "lock", 222 -  38 , 0, 12, 12, true, true}
}
```

Definition at line 240 of file keymap.cpp.

13.62.4.5 const Key K68TopRow[] [static]

Initial value:

```
= {
    {0, "Volume Down", "voldn", 285 -  38 , 0, 13, 8, true, true}, {0, "Volume Up", "volup", 297 -  38 , 0,
       13, 8, true, true},
}
```

Definition at line 235 of file keymap.cpp.

13.62.4.6 QHash<QString, Key> K95BaseMap [static]

Definition at line 342 of file keymap.cpp.

Referenced by getMap().

13.62.4.7 const Key K95Keys[] [static]

Initial value:

```
= {
    {0, 0, "mr", 38, 0,  12, 12 , true, true}, {0, 0, "m1", 50, 0,  12, 12 , true, true}, {0, 0, "m2", 62,
      0,  12, 12 , true, true}, {0, 0, "m3", 74, 0,  12, 12 , true, true}, {0, "Brightness", "light", 222, 0,  12,
       12 , true, true}, {0, "Windows Lock", "lock", 234, 0,  12, 12 , true, true}, {0, "Mute", "mute", 273, 0, 13
      , 8, true, true}, {0, "Volume Up", "volup", 290, -2, 18, 6, false, true}, {0, "Volume down", "voldn", 290, 2
      , 18, 6, false, true},
    {0, 0, "g1", 0, 14,  12, 12 , true, true}, {0, 0, "g2", 11, 14,  12, 12 , true, true}, {0, 0, "g3", 22,
       14,  12, 12 , true, true}, {0, "Esc", "esc", 38, 14,  12, 12 , true, true}, {0, 0, "f1", 58, 14,  12, 12 ,
      true, true}, {0, 0, "f2", 70, 14,  12, 12 , true, true}, {0, 0, "f3", 82, 14,  12, 12 , true, true}, {0, 0,
      "f4", 94, 14,  12, 12 , true, true}, {0, 0, "f5", 114, 14,  12, 12 , true, true}, {0, 0, "f6", 126, 14,  12,
       12 , true, true}, {0, 0, "f7", 138, 14,  12, 12 , true, true}, {0, 0, "f8", 150, 14,  12, 12 , true, true},
       {0, 0, "f9", 170, 14,  12, 12 , true, true}, {0, 0, "f10", 182, 14,  12, 12 , true, true}, {0, 0, "f11",
      194, 14,  12, 12 , true, true}, {0, 0, "f12", 206, 14,  12, 12 , true, true}, {0, "Print Screen\nSysRq", "
      prtscn", 222, 14,  12, 12 , true, true}, {0, "Scroll Lock", "scroll", 234, 14,  12, 12 , true, true}, {0, "Pause
      \nBreak", "pause", 246, 14,  12, 12 , true, true}, {0, "Stop", "stop", 262, 14, 12, 8, true, true}, {0, "
      Previous", "prev", 273, 14, 13, 8, true, true}, {0, "Play/Pause", "play", 285, 14, 13, 8, true, true}, {0, "
      Next", "next", 296, 14, 12, 8, true, true},
    {0, 0, "g4", 0, 25,  12, 12 , true, true}, {0, 0, "g5", 11, 25,  12, 12 , true, true}, {0, 0, "g6", 22,
       25,  12, 12 , true, true}, {0, "`", "grave", 38, 27,  12, 12 , true, true}, {0, 0, "1", 50, 27,  12, 12 ,
      true, true}, {0, 0, "2", 62, 27,  12, 12 , true, true}, {0, 0, "3", 74, 27,  12, 12 , true, true}, {0, 0, "4"
      , 86, 27,  12, 12 , true, true}, {0, 0, "5", 98, 27,  12, 12 , true, true}, {0, 0, "6", 110, 27,  12, 12 ,
      true, true}, {0, 0, "7", 122, 27,  12, 12 , true, true}, {0, 0, "8", 134, 27,  12, 12 , true, true}, {0, 0, "
      9", 146, 27,  12, 12 , true, true}, {0, 0, "0", 158, 27,  12, 12 , true, true}, {0, "-", "minus", 170, 27,
      12, 12 , true, true}, {0, "=", "equal", 182, 27,  12, 12 , true, true}, {0, "¥", "yen", 194, 27,  12, 12 ,
```

```
      true, true}, {0, "Backspace", "bspace", 200, 27, 24, 12, true, true}, {0, "Insert", "ins", 222, 27,  12, 12 ,
       true, true}, {0, "Home", "home", 234, 27,  12, 12 , true, true}, {0, "Page Up", "pgup", 246, 27,  12, 12 ,
      true, true}, {0, "Num Lock", "numlock", 261, 27,  12, 12 , true, true}, {0, "NumPad /", "numslash", 273, 27,
        12, 12 , true, true}, {0, "NumPad *", "numstar", 285, 27,  12, 12 , true, true}, {0, "NumPad -", "numminus
      ", 297, 27,  12, 12 , true, true},
    {0, 0, "g7", 0, 39,  12, 12 , true, true}, {0, 0, "g8", 11, 39,  12, 12 , true, true}, {0, 0, "g9", 22,
       39,  12, 12 , true, true}, {0, "Tab", "tab", 41, 39, 18, 12, true, true}, {0, 0, "q", 56, 39,  12, 12 ,
      true, true}, {0, 0, "w", 68, 39,  12, 12 , true, true}, {0, 0, "e", 80, 39,  12, 12 , true, true}, {0, 0, "r",
      92, 39,  12, 12 , true, true}, {0, 0, "t", 104, 39,  12, 12 , true, true}, {0, 0, "y", 116, 39,  12, 12 ,
      true, true}, {0, 0, "u", 128, 39,  12, 12 , true, true}, {0, 0, "i", 140, 39,  12, 12 , true, true}, {0, 0, "o
      ", 152, 39,  12, 12 , true, true}, {0, 0, "p", 164, 39,  12, 12 , true, true}, {0, "[", "lbrace", 176, 39,
      12, 12 , true, true}, {0, "]", "rbrace", 188, 39,  12, 12 , true, true}, {0, "\\", "bslash", 203, 39, 18, 12
      , true, true}, {0, "Enter", "enter", 203, 39, 18, 24, true, true}, {0, "Delete", "del", 222, 39,  12, 12 ,
      true, true}, {0, "End", "end", 234, 39,  12, 12 , true, true}, {0, "Page Down", "pgdn", 246, 39,  12, 12 ,
      true, true}, {0, "NumPad 7", "num7", 261, 39,  12, 12 , true, true}, {0, "NumPad 8", "num8", 273, 39,  12, 12
      , true, true}, {0, "NumPad 9", "num9", 285, 39,  12, 12 , true, true}, {0, "NumPad +", "numplus", 297, 45,
      12, 24, true, true},
    {0, 0, "g10", 0, 50,  12, 12 , true, true}, {0, 0, "g11", 11, 50,  12, 12 , true, true}, {0, 0, "g12",
      22, 50,  12, 12 , true, true}, {0, "Caps Lock", "caps", 42, 51, 20, 12, true, true}, {0, 0, "a", 59, 51,  12
      , 12 , true, true}, {0, 0, "s", 71, 51,  12, 12 , true, true}, {0, 0, "d", 83, 51,  12, 12 , true, true}, {0
      , 0, "f", 95, 51,  12, 12 , true, true}, {0, 0, "g", 107, 51,  12, 12 , true, true}, {0, 0, "h", 119, 51,
      12, 12 , true, true}, {0, 0, "j", 131, 51,  12, 12 , true, true}, {0, 0, "k", 143, 51,  12, 12 , true, true},
       {0, 0, "l", 155, 51,  12, 12 , true, true}, {0, ";", "colon", 167, 51,  12, 12 , true, true}, {0, "'", "
      quote", 179, 51,  12, 12 , true, true}, {0, "#", "hash", 191, 51,  12, 12 , true, true}, {0, "NumPad 4", "num4"
      , 261, 51,  12, 12 , true, true}, {0, "NumPad 5", "num5", 273, 51,  12, 12 , true, true}, {0, "NumPad 6", "
      num6", 285, 51,  12, 12 , true, true},
    {0, 0, "g13", 0, 64,  12, 12 , true, true}, {0, 0, "g14", 11, 64,  12, 12 , true, true}, {0, 0, "g15",
      22, 64,  12, 12 , true, true}, {0, "Left Shift", "lshift", 39, 63, 14, 12, true, true}, {"bslash", "\\", "
      bslash_iso", 53, 63,  12, 12 , true, true}, {0, 0, "z", 65, 63,  12, 12 , true, true}, {0, 0, "x", 77, 63,  12
      , 12 , true, true}, {0, 0, "c", 89, 63,  12, 12 , true, true}, {0, 0, "v", 101, 63,  12, 12 , true, true}, {
      0, 0, "b", 113, 63,  12, 12 , true, true}, {0, 0, "n", 125, 63,  12, 12 , true, true}, {0, 0, "m", 137, 63,
       12, 12 , true, true}, {0, ",", "comma", 149, 63,  12, 12 , true, true}, {0, ".", "dot", 161, 63,  12, 12 ,
      true, true}, {0, "/", "slash", 173, 63,  12, 12 , true, true}, {0, "_", "ro", 185, 63,  12, 12 , true, true}
      , {0, "Right Shift", "rshift", 196, 63, 32, 12, true, true}, {0, "Up", "up", 234, 63,  12, 12 , true, true},
       {0, "NumPad 1", "num1", 261, 63,  12, 12 , true, true}, {0, "NumPad 2", "num2", 273, 63,  12, 12 , true,
      true}, {0, "NumPad 3", "num3", 285, 63,  12, 12 , true, true}, {0, "NumPad Enter", "numenter", 297, 69, 12, 24
      , true, true},
    {0, 0, "g16", 0, 75,  12, 12 , true, true}, {0, 0, "g17", 11, 75,  12, 12 , true, true}, {0, 0, "g18",
      22, 75,  12, 12 , true, true}, {0, "Left Ctrl", "lctrl", 40, 75, 16, 12, true, true}, {0, "Left Windows", "
      lwin", 54, 75,  12, 12 , true, true}, {0, "Left Alt", "lalt", 67, 75, 14, 12, true, true}, {0, "", "
      muhenkan", 80, 75,  12, 12 , true, true}, {0, "Space", "space", 116, 75, 84, 12, true, true}, {0, "", "henkan",
      150, 75,  12, 12 , true, true}, {0, "  ", "katahira", 162, 75,  12, 12 , true, true}, {0, "Right
       Alt", "ralt", 165, 75, 14, 12, true, true}, {0, "Right Windows", "rwin", 178, 75,  12, 12 , true, true}, {0
      , "Menu", "rmenu", 190, 75,  12, 12 , true, true}, {0, "Right Ctrl", "rctrl", 204, 75, 16, 12, true, true},
      {0, "Left", "left", 222, 75,  12, 12 , true, true}, {0, "Down", "down", 234, 75,  12, 12 , true, true}, {0,
      "Right", "right", 246, 75,  12, 12 , true, true}, {0, "NumPad 0", "num0", 267, 75, 24, 12, true, true}, {0,
      "NumPad .", "numdot", 285, 75,  12, 12 , true, true}
}
```

Definition at line 12 of file keymap.cpp.

13.62.4.8 const Key K95PLbar[] [static]

Initial value:

```
= {
    {0, 0, "topbar1", 4, -3,  17, 6 , true, false}, {0, 0, "topbar2", 19, -3,  17, 6 , true, false}, {0, 0,
       "topbar3", 34, -3,  17, 6 , true, false}, {0, 0, "topbar4", 49, -3,  17, 6 , true, false}, {0, 0, "topbar5"
      , 64, -3,  17, 6 , true, false}, {0, 0, "topbar6", 79, -3,  17, 6 , true, false},
    {0, 0, "topbar7", 94, -3,  17, 6 , true, false}, {0, 0, "topbar8", 109, -3,  17, 6 , true, false}, {0,
      0, "topbar9", 124, -3,  17, 6 , true, false}, {0, 0, "topbar10", 139, -3,  17, 6 , true, false}, {0, 0, "
      topbar11", 154, -3,  17, 6 , true, false}, {0, 0, "topbar12", 169, -3,  17, 6 , true, false},
    {0, 0, "topbar13", 184, -3,  17, 6 , true, false}, {0, 0, "topbar14", 199, -3,  17, 6 , true, false}, {
      0, 0, "topbar15", 214, -3,  17, 6 , true, false}, {0, 0, "topbar16", 229, -3,  17, 6 , true, false}, {0, 0,
      "topbar17", 244, -3,  17, 6 , true, false}, {0, 0, "topbar18", 259, -3,  17, 6 , true, false},
    {0, 0, "topbar19", 274, -3,  17, 6 , true, false},
}
```

Definition at line 330 of file keymap.cpp.

13.62.4.9 const Key KStrafeKeys[] [static]

Initial value:

```
= {
    {0, "Sidelight", "lsidel", 0,   76  /2,  12 ,   76  , true, false},
```

```
    {0, "Sidelight", "rsidel",  ( ( 298  -  38 )  + ( 12  * 2)) ,   76  /2,  12 ,   76  , true, false},
    {0, "Logo", "logo",  12 , 0,  12, 12 , true, false},
    {0, "Function", "fn", 152, 75,  12, 12 , true, true}
}
```

Definition at line 255 of file keymap.cpp.

13.62.4.10 const Key M65Keys[] [static]

Initial value:

```
= {
    {0, "Left Mouse", "mouse1", 8, 0, 14, 32, false, true}, {0, "Right Mouse", "mouse2", 30, 0, 14, 32,
      false, true}, {0, "Middle Mouse", "mouse3", 22, 8, 8, 7, false, true},
    {0, "Wheel Up", "wheelup", 22, 4, 8, 5, false, true}, {0, "Wheel Down", "wheeldn", 22, 14, 8, 5, false,
       true}, {0, "Wheel Light", "front", 22, 15, 8, 8, true, false},
    {0, "DPI Up", "dpiup", 22, 19, 8, 6, false, true}, {0, "DPI Light", "dpi", 22, 24, 8, 8, true, false},
      {0, "DPI Down", "dpidn", 22, 31, 8, 6, false, true},
    {0, "Forward", "mouse5", 5, 24, 5, 9, false, true}, {0, "Back", "mouse4", 5, 33, 5, 10, false, true}, {
      0, "Sniper", "sniper", 0, 25, 5, 15, false, true},
    {0, "Logo", "back", 14, 55, 24, 12, true, false}
}
```

Definition at line 263 of file keymap.cpp.

13.62.4.11 const KeyPatch patchDE[] [static]

Initial value:

```
= {
    {"caret", "^", "grave"}, {"ss", "ß", "minus"}, {"grave", "´", "equal"},
    {"z", "Z", "y"}, {"ue", "Ü", "lbrace"}, {"plus", "+", "rbrace"},
    {"oe", "Ö", "colon"}, {"ae", "Ä", "quote"},
    {"angle", "<", "bslash_iso"}, {"y", "Y", "z"}, {"minus", "-", "slash"}
}
```

Definition at line 57 of file keymap.cpp.

13.62.4.12 const KeyPatch patchDK[] [static]

Initial value:

```
= {
    {0, "§", "grave"}, {0, "+", "minus"}, {0, "´", "equal"},
    {0, "Å", "lbrace"}, {0, "¨", "rbrace"},
    {0, "Æ", "colon"}, {0, "Ø", "quote"}, {0, "’", "hash"},
    {0, "<", "bslash_iso"}, {0, "-", "slash"},
}
```

Definition at line 38 of file keymap.cpp.

13.62.4.13 const KeyPatch patchDvorak[] [static]

Initial value:

```
= {
    {0, "[", "minus"}, {0, "]", "equal"},
    {0, "’", "q"}, {0, ",", "w"}, {0, ".", "e"}, {0, "P", "r"}, {0, "Y", "t"}, {0, "F", "y"}, {0, "G", "u"}
      , {0, "C", "i"}, {0, "R", "o"}, {0, "L", "p"}, {0, "/", "lbrace"}, {0, "=", "rbrace"},
    {0, "O", "s"}, {0, "E", "d"}, {0, "U", "f"}, {0, "I", "g"}, {0, "D", "h"}, {0, "H", "j"}, {0, "T", "k"}
      , {0, "N", "l"}, {0, "S", "colon"}, {0, "-", "quote"},
    {0, ";", "z"}, {0, "Q", "x"}, {0, "J", "c"}, {0, "K", "v"}, {0, "X", "b"}, {0, "B", "n"}, {0, "W", "
      comma"}, {0, "V", "dot"}, {0, "Z", "slash"},
}
```

Definition at line 111 of file keymap.cpp.

13.62.4.14 const KeyPatch patchES[] [static]

Initial value:

```
= {
    {"oa", "°", "grave"}, {"quote", "´", "minus"}, {"lexclam", "¡", "equal"},
    {"grave", "`", "lbrace"}, {"plus", "+", "rbrace"},
    {"nn", "Ñ", "colon"}, {"accent", "¨", "quote"}, {"cc", "Ç", "hash"},
    {"angle", "<", "bslash_iso"}, {"minus", "-", "slash"},
}
```

Definition at line 92 of file keymap.cpp.

13.62.4.15 const KeyPatch patchEU[] [static]

Initial value:

```
= {
    {0, "\\ (R)", "hash"},
    {0, "\\ (L)", "bslash_iso"},
}
```

Definition at line 45 of file keymap.cpp.

13.62.4.16 const KeyPatch patchFR[] [static]

Initial value:

```
= {
    {"sup2", "²", "grave"}, {0, "&", "1"}, {0, "É", "2"}, {0, "\"", "3"}, {0, "´", "4"}, {0, "(", "5"}, {0,
       "-", "6"}, {0, "È", "7"}, {0, "_", "8"}, {0, "Ç", "9"}, {0, "À", "0"}, {"rparen", ")", "minus"},
    {"a", "A", "q"}, {"z", "Z", "w"}, {"caret", "^", "lbrace"}, {"dollar", "$", "rbrace"},
    {"q", "Q", "a"}, {"m", "M", "colon"}, {"percent", "Ù", "quote"}, {"star", "*", "hash"},
    {"angle", "<", "bslash_iso"}, {"w", "W", "z"}, {"comma", ",", "m"}, {"semicolon", ";", "comma"}, {"
      colon", ":", "dot"}, {"exclam", "!", "slash"},
}
```

Definition at line 50 of file keymap.cpp.

13.62.4.17 const KeyPatch patchIT[] [static]

Initial value:

```
= {
    {0, "\\", "grave"}, {0, "´", "minus"}, {0, "ì", "equal"},
    {0, "è", "lbrace"}, {0, "+", "rbrace"},
    {0, "ò", "colon"}, {0, "à", "quote"}, {0, "ù", "hash"},
    {0, "<", "bslash_iso"}, {0, "-", "slash"},
}
```

Definition at line 64 of file keymap.cpp.

13.62.4.18 const KeyPatch patchJP[] [static]

Initial value:

```
= {

}
```

Definition at line 106 of file keymap.cpp.

13.62.4.19 const KeyPatch patchMX[] [static]

Initial value:

```
= {
    {0, "|", "grave"}, {0, "´", "minus"}, {0, "¿", "equal"},
    {0, "´", "lbrace"}, {0, "+", "rbrace"},
    {0, "Ñ", "colon"}, {0, "{", "quote"}, {0, "}", "hash"},
    {0, "<", "bslash_iso"}, {0, "-", "slash"},
}
```

Definition at line 78 of file keymap.cpp.

13.62.4.20 const KeyPatch patchNO[] [static]

Initial value:

```
= {
    {0, "§", "grave"}, {0, "+", "minus"}, {0, "´", "equal"},
    {0, "Å", "lbrace"}, {0, "¨", "rbrace"},
    {0, "Ø", "colon"}, {0, "Æ", "quote"}, {0, "'", "hash"},
    {0, "<", "bslash_iso"}, {0, "-", "slash"},
}
```

Definition at line 71 of file keymap.cpp.

13.62.4.21 const KeyPatch patchPTBR[] [static]

Initial value:

```
= {
    {"quote", "'", "grave"},
    {"accent", "´", "lbrace"}, {"lbrace", "[", "rbrace"},
    {"cc", "Ç", "colon"}, {"tilde", "~", "quote"}, {"rbrace", "]", "hash"},
    {"colon", ";", "slash"}, {"slash", "/", "ro"}, {"numcomma", ",", "numdot"},
}
```

Definition at line 85 of file keymap.cpp.

13.62.4.22 const KeyPatch patchSE[] [static]

Initial value:

```
= {
    {"section", "§", "grave"}, {"plus", "+", "minus"}, {"grave", "´", "equal"},
    {"aa", "Å", "lbrace"}, {"umlaut", "¨", "rbrace"},
    {"oe", "Ö", "colon"}, {"ae", "Ä", "quote"}, {"quote", "'", "hash"},
    {"angle", "<", "bslash_iso"}, {"minus", "-", "slash"},
}
```

Definition at line 99 of file keymap.cpp.

13.62.4.23 const Key SabreKeys[] [static]

Initial value:

```
= {
    {0, "Left Mouse", "mouse1", 8, 0, 14, 32, false, true}, {0, "Right Mouse", "mouse2", 30, 0, 14, 32,
      false, true}, {0, "Middle Mouse", "mouse3", 22, 9, 8, 7, false, true}, {0, "Front light", "front", 8, -2, 14, 8
      , true, false },
    {0, "Wheel Up", "wheelup", 22, 5, 8, 5, false, true}, {0, "Wheel Down", "wheeldn", 22, 15, 8, 5, false,
       true}, {0, "Wheel Light", "wheel", 22, 5, 8, 15, true, false}, {0, "Extra button", "thumb1", 22, 20, 8, 18,
       false, true},
    {0, "DPI Up", "dpiup", 5, 3, 5, 7, false, true}, {0, "DPI Down", "dpidn", 5, 10, 5, 7, false, true}, {0
```

```
      , "DPI Light", "dpi", 5, 4, 5, 12, true, false},
    {0, "Forward", "mouse5", 5, 24, 5, 9, false, true}, {0, "Back", "mouse4", 5, 33, 5, 10, false, true},
    {0, "Logo", "back", 14, 50, 24, 12, true, false}
}
```

Definition at line 276 of file keymap.cpp.

13.62.4.24 const Key ScimKeys[] [static]

Initial value:

```
= {
    {0, "Left Mouse", "mouse1", 8, 0, 14, 32, false, true}, {0, "Right Mouse", "mouse2", 30, 0, 12, 32,
      false, true}, {0, "Middle Mouse", "mouse3", 22, 9, 8, 6, false, true}, {0, "Front light", "front", 30, 0, 12, 8
      , true, false },
    {0, "Wheel Up", "wheelup", 22, 3, 8, 6, false, true}, {0, "Wheel Down", "wheeldn", 22, 14, 8, 6, false,
       true}, {0, "Wheel Light", "wheel", 22, 3, 8, 17, true, false},
    {0, "DPI Up", "dpiup", 22, 19, 8, 9, false, true}, {0, "DPI Light", "dpi", 1, 12, 8, 4, true, false}, {
      0, "DPI Down", "dpidn", 22, 28, 8, 9, false, true},
    {0, "Thumb light", "thumb", 0, 21, 10, 24, true, false},
    {0, "1", "thumb1", -13, 18, 7, 7, false, true}, {0, "2", "thumb2", -6, 18, 7, 7, false, true}, {0, "3",
       "thumb3", 1, 18, 7, 7, false, true},
    {0, "4", "thumb4", -13, 25, 7, 7, false, true}, {0, "5", "thumb5", -6, 25, 7, 7, false, true}, {0, "6",
       "thumb6", 1, 25, 7, 7, false, true},
    {0, "7", "thumb7", -13, 32, 7, 7, false, true}, {0, "8", "thumb8", -6, 32, 7, 7, false, true}, {0, "9",
       "thumb9", 1, 32, 7, 7, false, true},
    {0, "10", "thumb10", -13, 39, 7, 7, false, true}, {0, "11", "thumb11", -6, 39, 7, 7, false, true}, {0,
      "12", "thumb12", 1, 39, 7, 7, false, true},
    {0, "Logo", "back", 14, 50, 24, 16, true, false}
}
```

Definition at line 313 of file keymap.cpp.

13.62.4.25 QHash<QString, Key> standardMaps[KeyMap::_MODEL_MAX][KeyMap::_LAYOUT_MAX] [static]

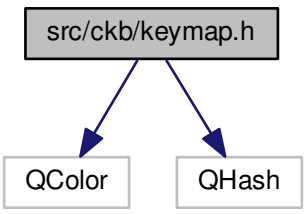
Definition at line 343 of file keymap.cpp.

Referenced by getMap().

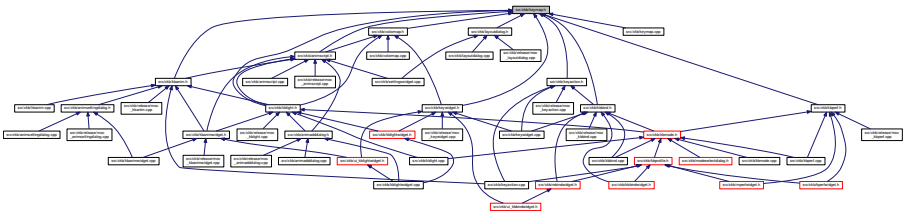
13.63 src/ckb/keymap.h File Reference

```
#include <QColor>
#include <QHash>
```

Include dependency graph for keymap.h:



This graph shows which files directly or indirectly include this file:



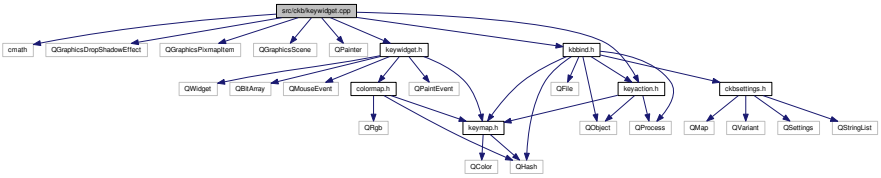
Data Structures

- struct Key
- class KeyMap

13.64 src/ckb/keywidget.cpp File Reference

```
#include <cmath>
#include <QGraphicsDropShadowEffect>
#include <QGraphicsPixmapItem>
#include <QGraphicsScene>
#include <QPainter>
#include "keywidget.h"
#include "keyaction.h"
#include "kbbind.h"
```

Include dependency graph for keywidget.cpp:



Functions

- QRgb monoRgb (float r, float g, float b)

Variables

- static const int KEY_SIZE = 12
- static QImage * m65Overlay = 0
- static QImage * sabOverlay = 0
- static QImage * scimOverlay = 0
- static QImage * harpOverlay = 0
- static QImage * glaiveOverlay = 0

13.64.1 Function Documentation

13.64.1.1 QRgb monoRgb (float *r*, float *g*, float *b*)

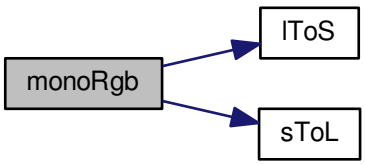
Definition at line 284 of file kblight.cpp.

References lToS(), and sToL().

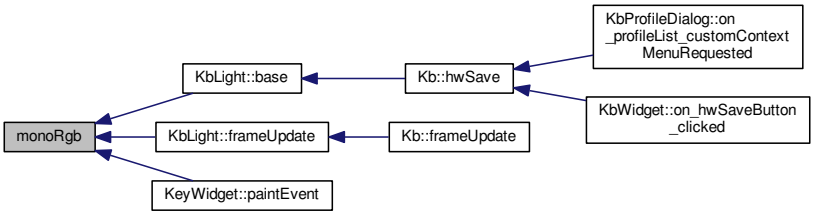
Referenced by KbLight::base(), KbLight::frameUpdate(), and KeyWidget::paintEvent().

```
284                                     {
285     // It's important to use a linear colorspace for this, otherwise the colors will appear inconsistent
286     // Note that although we could use linear space for alpha blending or the animation blending functions,
        we don't.
287     // The reason for this is that photo manipulation programs don't do it either, so even though the
        result would technically be more correct,
288     // it would look wrong to most people.
289     r = sToL(r);
290     g = sToL(g);
291     b = sToL(b);
292     int value = round(lToS((r + g + b) / 3.f));
293     return qRgb(value, value, value);
294 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



13.64.2 Variable Documentation

13.64.2.1 QImage * glaiveOverlay = 0 [static]

Definition at line 12 of file keywidget.cpp.

Referenced by KeyWidget::paintEvent().

13.64.2.2 QImage * harpOverlay = 0 [static]

Definition at line 12 of file keywidget.cpp.

Referenced by KeyWidget::paintEvent().

13.64.2.3 const int KEY_SIZE = 12 [static]

Definition at line 10 of file keywidget.cpp.

Referenced by KeyWidget::drawInfo(), and KeyWidget::map().

13.64.2.4 QImage* m65Overlay = 0 [static]

Definition at line 12 of file keywidget.cpp.

Referenced by KeyWidget::paintEvent().

13.64.2.5 QImage * sabOverlay = 0 [static]

Definition at line 12 of file keywidget.cpp.

Referenced by KeyWidget::paintEvent().

13.64.2.6 QImage * scimOverlay = 0 [static]

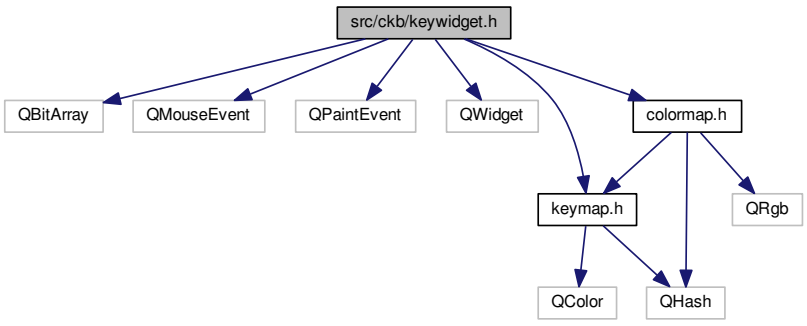
Definition at line 12 of file keywidget.cpp.

Referenced by KeyWidget::paintEvent().

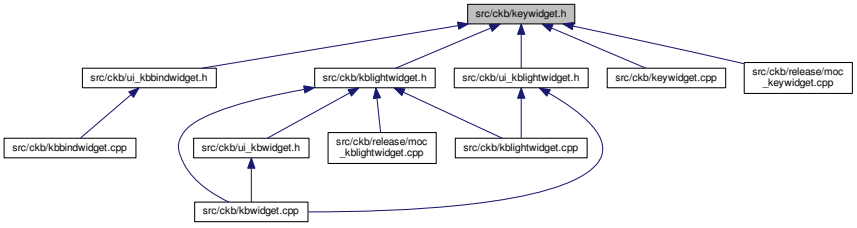
13.65 src/ckb/keywidget.h File Reference

```
#include <QBitArray>
#include <QMouseEvent>
#include <QPaintEvent>
#include <QWidget>
#include "keymap.h"
#include "colormap.h"
```

Include dependency graph for keywidget.h:



This graph shows which files directly or indirectly include this file:



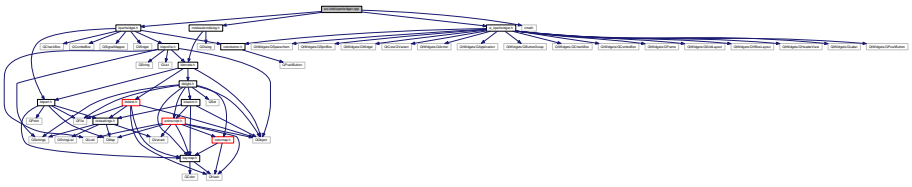
Data Structures

- class KeyWidget

13.66 src/ckb/kperfwidget.cpp File Reference

```
#include "kperfwidget.h"
#include "ui_kperfwidget.h"
#include "modeselectdialog.h"
#include <cmath>
```

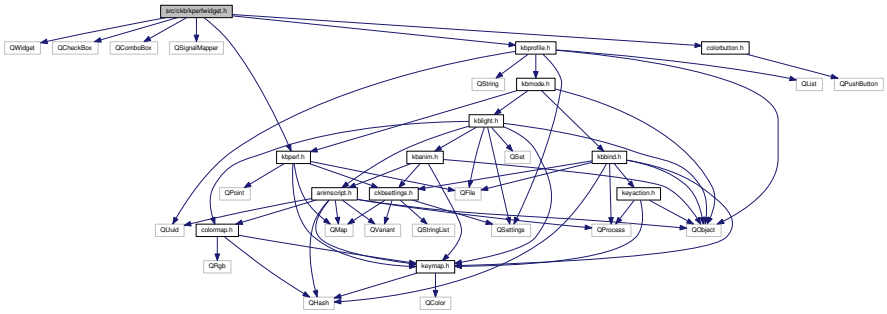
Include dependency graph for kperfwidget.cpp:



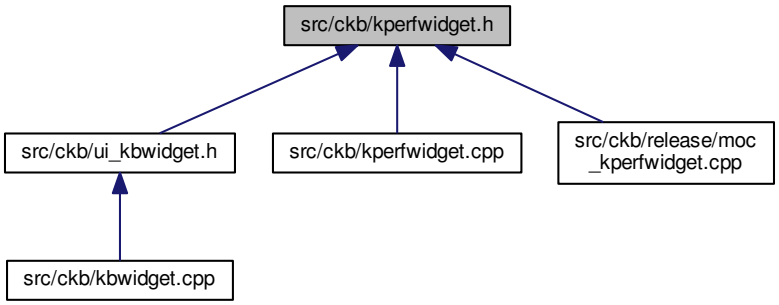
13.67 src/ckb/kperfwidget.h File Reference

```
#include <QWidget>
#include <QCheckBox>
#include <QComboBox>
#include <QSignalMapper>
#include "kbperf.h"
#include "kbprofile.h"
#include "colorbutton.h"
```

Include dependency graph for kperfwidget.h:



This graph shows which files directly or indirectly include this file:



Data Structures

- class KPerfWidget
- struct KPerfWidget::IndicatorUi

Namespaces

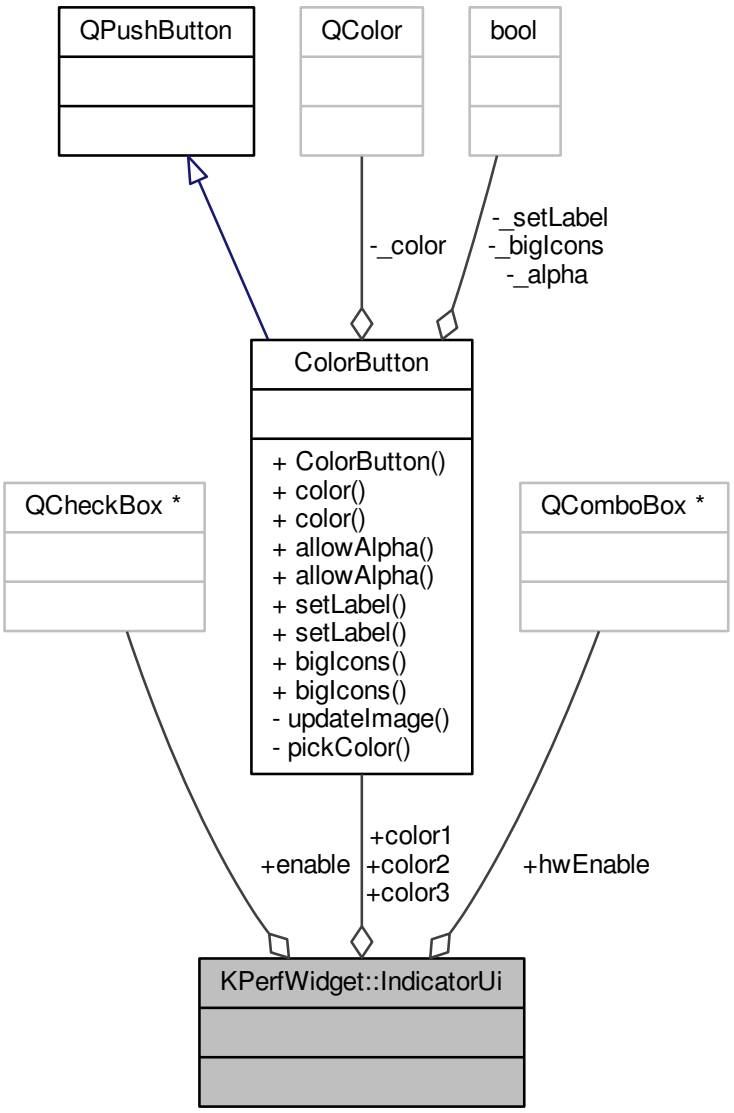
- Ui

13.67.1 Data Structure Documentation

13.67.1.1 struct KPerfWidget::IndicatorUi

Definition at line 54 of file kperfwidget.h.

Collaboration diagram for KPerfWidget::IndicatorUi:



Data Fields

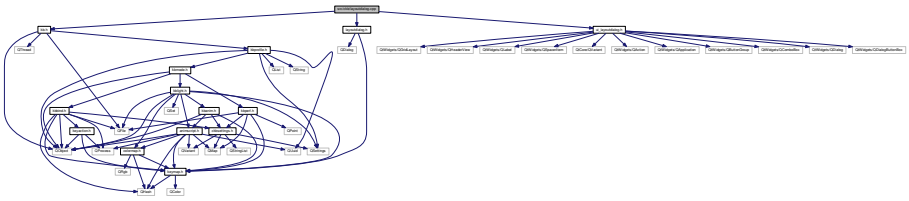
ColorButton *	color1	
ColorButton *	color2	
ColorButton *	color3	
QCheckBox *	enable	
QComboBox *	hwEnable	

13.68 src/ckb/layoutdialog.cpp File Reference

```
#include "layoutdialog.h"
```

```
#include "kb.h"
#include "ui_layoutdialog.h"
```

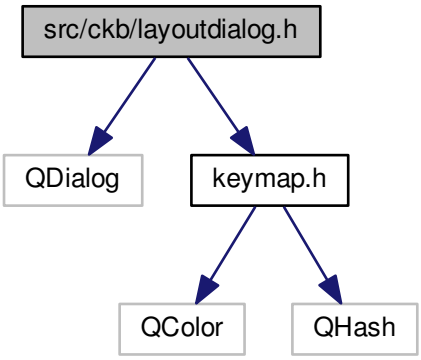
Include dependency graph for layoutdialog.cpp:



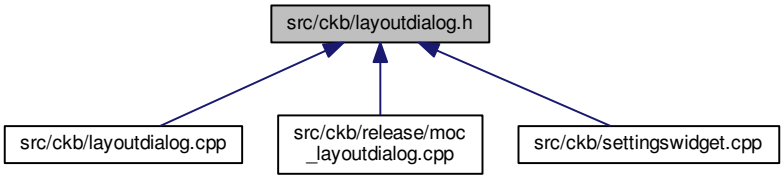
13.69 src/ckb/layoutdialog.h File Reference

```
#include <QDialog>
#include "keymap.h"
```

Include dependency graph for layoutdialog.h:



This graph shows which files directly or indirectly include this file:



Data Structures

- class LayoutDialog

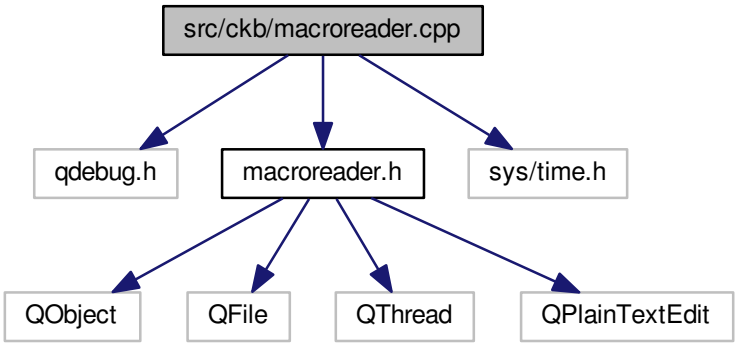
Namespaces

- Ui

13.70 src/ckb/macroreader.cpp File Reference

```
#include <qdebug.h>
#include "macroreader.h"
#include <sys/time.h>
```

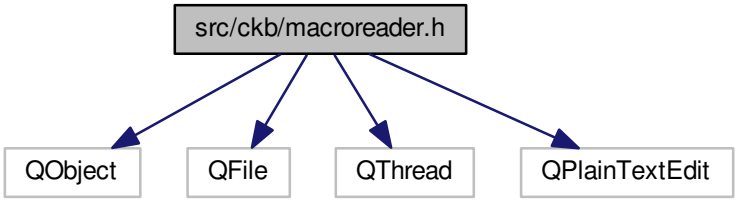
Include dependency graph for macroreader.cpp:



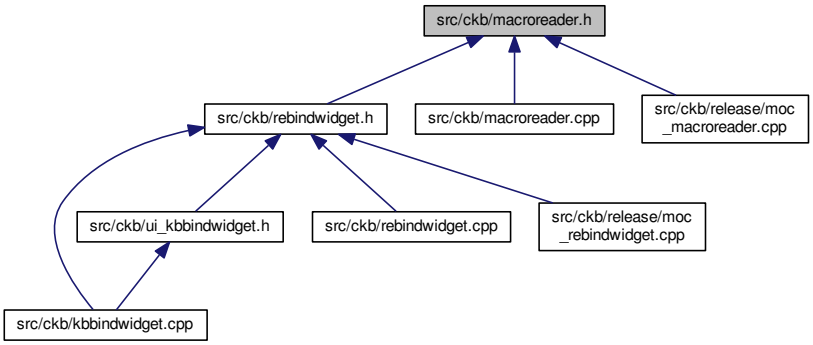
13.71 src/ckb/macroreader.h File Reference

```
#include <QObject>
#include <QFile>
#include <QThread>
#include <QPlainTextEdit>
```

Include dependency graph for macroreader.h:



This graph shows which files directly or indirectly include this file:



Data Structures

- class MacroReaderThread

  The MacroReaderThread class is responsible for reading Macro Key Values. It is created as a separate thread (worker thread) for reading macro commands from an fresh opened notify channel. Standard notify channel for macro definitions is number 2.

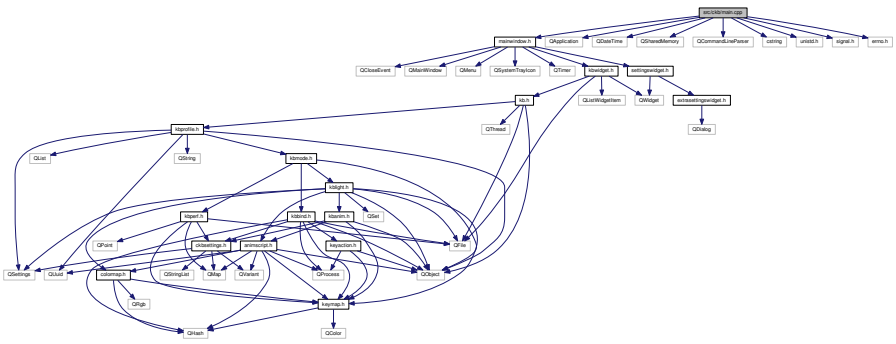
- class MacroReader

  The MacroReader class creates a worker thread object. It does a connect do delayed deletion of thread local variables in the case the worker thread terminates.

13.72 src/ckb/main.cpp File Reference

```
#include "mainwindow.h"
#include <QApplication>
#include <QDateTime>
#include <QSharedMemory>
#include <QCommandLineParser>
#include <cstring>
#include <unistd.h>
#include <signal.h>
#include <errno.h>
```

Include dependency graph for main.cpp:



Macros

- #define SHMEM_SIZE 128 ∗ 1024
- #define SHMEM_SIZE_V015 16

Enumerations

- enum CommandLineParseResults {
  CommandLineOK, CommandLineError, CommandLineVersionRequested, CommandLineHelpRequested,
  CommandLineClose, CommandLineBackground }

Functions

- QSharedMemory appShare ("ckb")
- CommandLineParseResults parseCommandLine (QCommandLineParser &parser, QString ∗errorMessage)

  parseCommandLine - Setup options and parse command line arguments.
- static bool pidActive (const QStringList &lines)
- static bool isRunning (const char ∗command)
- int main (int argc, char ∗argv[])

13.72.1 Macro Definition Documentation

13.72.1.1 #define SHMEM_SIZE 128 ∗ 1024

Definition at line 20 of file main.cpp.

Referenced by isRunning().

13.72.1.2 #define SHMEM_SIZE_V015 16

Definition at line 21 of file main.cpp.

Referenced by isRunning().

13.72.2 Enumeration Type Documentation

13.72.2.1 enum CommandLineParseResults

Enumerator

CommandLineOK

CommandLineError

CommandLineVersionRequested

CommandLineHelpRequested

CommandLineClose

CommandLineBackground

Definition at line 24 of file main.cpp.

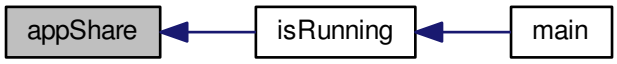
```
24                              {
25     CommandLineOK,
26     CommandLineError,
27     CommandLineVersionRequested,
28     CommandLineHelpRequested,
29     CommandLineClose,
30     CommandLineBackground
31 };
```

13.72.3 Function Documentation

13.72.3.1 QSharedMemory appShare ("ckb")

Referenced by isRunning().

Here is the caller graph for this function:



13.72.3.2 static bool isRunning (const char * *command*) [static]

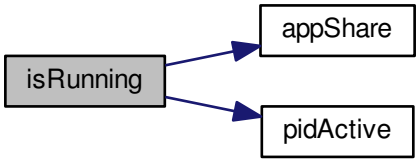
Definition at line 109 of file main.cpp.

References appShare(), pidActive(), SHMEM_SIZE, and SHMEM_SIZE_V015.

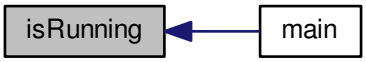
Referenced by main().

```
109                                         {
110     // Try to create shared memory (if successful, application was not already running)
111     if(!appShare.create(SHMEM_SIZE) || !appShare.lock()){
112         // Lock existing shared memory
113         if(!appShare.attach() || !appShare.lock())
114             return true;
115         bool running = false;
116         if(appShare.size() == SHMEM_SIZE_V015){
117             // Old shmem - no PID listed so assume the process is running, and print the command directly
    to the buffer
118             if(command){
119                 void* data = appShare.data();
120                 snprintf((char*)data, SHMEM_SIZE_V015, "%s", command);
121             }
122             running = true;
123         } else {
124             // New shmem. Scan the contents of the shared memory for a PID
125             QStringList lines = QString((const char*)appShare.constData()).split("\n");
126             if(pidActive(lines)){
127                 running = true;
128                 // Append the command
129                 if(command){
130                     lines.append(QString("Option ") + command);
131                     QByteArray newMem = lines.join("\n").left(SHMEM_SIZE).toLatin1();
132                     // Copy the NUL byte as well as the string
133                     memcpy(appShare.data(), newMem.constData(), newMem.length() + 1);
134                 }
135             }
136         }
137         if(running){
138             appShare.unlock();
139             return true;
140         }
141     }
142     // Not already running. Initialize the shared memory with our PID
143     snprintf((char*)appShare.data(), appShare.size(), "PID %ld", (long)getpid());
144     appShare.unlock();
145     return false;
146 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



13.72.3.3 int main (int *argc*, char * *argv[]*)

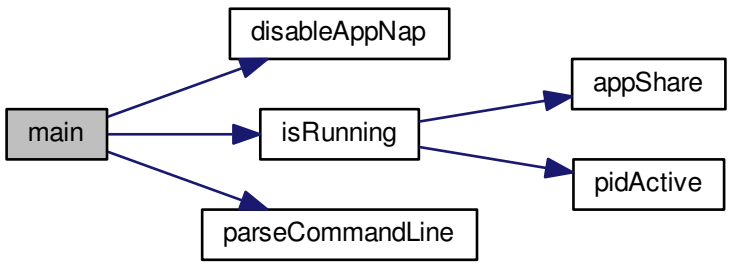
Definition at line 148 of file main.cpp.

References CommandLineBackground, CommandLineClose, CommandLineError, CommandLineHelpRequested, CommandLineOK, CommandLineVersionRequested, disableAppNap(), isRunning(), and parseCommandLine().

```
148                                {
149     // Setup main application
150     QApplication a(argc, argv);
151 
152     // Setup names and versions
153     QCoreApplication::setOrganizationName("ckb");
154     QCoreApplication::setApplicationVersion(CKB_VERSION_STR);
155     QCoreApplication::setApplicationName("ckb");
156 
157     // Setup argument parser
158     QCommandLineParser parser;
159     QString errorMessage;
160     parser.setApplicationDescription("Open Source Corsair Input Device Driver for Linux and OSX.");
161     bool background = 0;
162 
163     // Although the daemon runs as root, the GUI needn't and shouldn't be, as it has the potential to
      corrupt settings data.
164     if(getuid() == 0){
165         printf("The ckb GUI should not be run as root.\n");
166         return 0;
167     }
168 
169     // Seed the RNG for UsbIds
170     qsrand(QDateTime::currentMSecsSinceEpoch());
171 #ifdef Q_OS_MACX
172     disableAppNap();
173 
174     FILE *fp = fopen("/tmp/ckb", "w");
175     fprintf(fp, "%d", getpid());
176     fclose(fp);
177 #endif
178 #if QT_VERSION >= QT_VERSION_CHECK(5, 3, 0)
```

```
179     // Enable HiDPI support
180     qApp->setAttribute(Qt::AA_UseHighDpiPixmaps);
181 #endif
182
183     // Parse arguments
184     switch (parseCommandLine(parser, &errorMessage)) {
185     case CommandLineOK:
186         // If launched with no argument
187         break;
188     case CommandLineError:
189         fputs(qPrintable(errorMessage), stderr);
190         fputs("\n\n", stderr);
191         fputs(qPrintable(parser.helpText()), stderr);
192         return 1;
193     case CommandLineVersionRequested:
194         // If launched with --version, print version info and then quit
195         printf("%s %s\n", qPrintable(QCoreApplication::applicationName()),
196                 qPrintable(QCoreApplication::applicationVersion()));
197         return 0;
198     case CommandLineHelpRequested:
199         // If launched with --help, print help and then quit
200         parser.showHelp();
201         return 0;
202     case CommandLineClose:
203         // If launched with --close, kill existing app
204         if (isRunning("Close"))
205             printf("Asking existing instance to close.\n");
206         else
207             printf("ckb is not running.\n");
208         return 0;
209     case CommandLineBackground:
210         // If launched with --background, launch in background
211         background = 1;
212         break;
213     }
214
215     // Launch in background if requested, or if re-launching a previous session
216     if(qApp->isSessionRestored())
217             background = 1;
218     if(isRunning(background ? 0 : "Open")){
219         printf("ckb is already running. Exiting.\n");
220         return 0;
221     }
222     MainWindow w;
223     if(!background)
224         w.show();
225
226     return a.exec();
227 }
```

Here is the call graph for this function:



13.72.3.4 CommandLineParseResults parseCommandLine (QCommandLineParser & *parser*, QString * *errorMessage*)

Parameters

parser	parser instance to use for parse the arguments
errorMessage	argument parse error message

Returns

integer, representing the requested argument

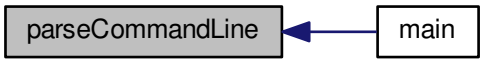
Definition at line 41 of file main.cpp.

References CommandLineBackground, CommandLineClose, CommandLineError, CommandLineHelpRequested, CommandLineOK, and CommandLineVersionRequested.

Referenced by main().

```
41                                                                                                  {
42     // setup parser to interpret -abc as -a -b -c
43     parser.setSingleDashWordOptionMode(QCommandLineParser::ParseAsCompactedShortOptions);
44
45     /* add command line options */
46     // add -v, --version
47     const QCommandLineOption versionOption = parser.addVersionOption();
48     // add -h, --help (/? on windows)
49     const QCommandLineOption helpOption = parser.addHelpOption();
50     // add -b, --background
51     const QCommandLineOption backgroundOption(QStringList() << "b" << "background",
52                                               "Starts in background, without displaying the main window.");
53     parser.addOption(backgroundOption);
54     // add -c, --close
55     const QCommandLineOption closeOption(QStringList() << "c" << "close",
56                                          "Causes already running instance (if any) to exit.");
57     parser.addOption(closeOption);
58
59     /* parse arguments */
60     if (!parser.parse(QCoreApplication::arguments())) {
61         // set error, if there already are some
62         *errorMessage = parser.errorText();
63         return CommandLineError;
64     }
65
66     /* return requested operation/setup */
67     if (parser.isSet(versionOption)) {
68         // print version and exit
69         return CommandLineVersionRequested;
70     }
71
72     if (parser.isSet(helpOption)) {
73         // print help and exit
74         return CommandLineHelpRequested;
75     }
76
77     if (parser.isSet(backgroundOption)) {
78         // open application in background
79         return CommandLineBackground;
80     }
81
82     if (parser.isSet(closeOption)) {
83         // close already running application instances, if any
84         return CommandLineClose;
85     }
86
87     /* no explicit argument was passed */
88     return CommandLineOK;
89 };
```

Here is the caller graph for this function:



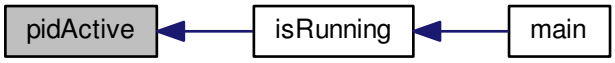
13.72.3.5 static bool pidActive (const QStringList & *lines*) [static]

Definition at line 92 of file main.cpp.

Referenced by isRunning().

```
92                                                {
93     foreach(const QString& line, lines){
94         if(line.startsWith("PID ")){
95             bool ok;
96             pid_t pid;
97             // Valid PID found?
98             if((pid = line.split(" ")[1].toLong(&ok)) > 0 && ok){
99                 // kill -0 does nothing to the application, but checks if it's running
100                 return (kill(pid, 0) == 0 || errno != ESRCH);
101             }
102         }
103     }
104     // If the PID wasn't listed in the shmem, assume it is running
105     return true;
106 }
```

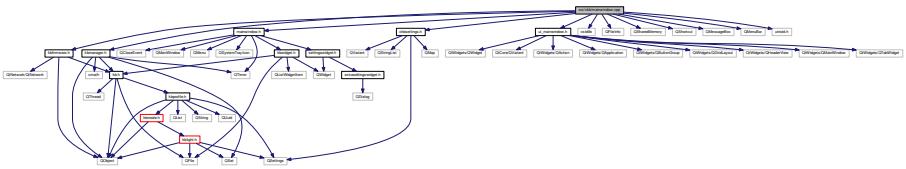
Here is the caller graph for this function:



13.73 src/ckb/mainwindow.cpp File Reference

```
#include "ckbsettings.h"
#include "kbmanager.h"
#include "kbfirmware.h"
#include "mainwindow.h"
#include "ui_mainwindow.h"
#include <cstdlib>
#include <QFileInfo>
#include <QSharedMemory>
#include <QShortcut>
#include <QMessageBox>
#include <QMenuBar>
#include <unistd.h>
```

Include dependency graph for mainwindow.cpp:



Variables

- QSharedMemory appShare
- QString devpath
- static const QString configLabel = "Settings"
- QString daemonDialogText

13.73.1 Variable Documentation

13.73.1.1 QSharedMemory appShare

Referenced by MainWindow::timerTick().

13.73.1.2 const QString configLabel = "Settings" [static]

Definition at line 17 of file mainwindow.cpp.

Referenced by MainWindow::MainWindow().

13.73.1.3 QString daemonDialogText

Initial value:

```
= QObject::tr("Start it once with:") +
    "<blockquote><code>sudo systemctl start ckb-daemon</code></blockquote>" +
    QObject::tr("Enable it for every boot:") +
    "<blockquote><code>sudo systemctl enable ckb-daemon</code></blockquote>"
```

Definition at line 20 of file mainwindow.cpp.

Referenced by MainWindow::MainWindow().

13.73.1.4 QString devpath

Definition at line 4 of file kbmanager.cpp.

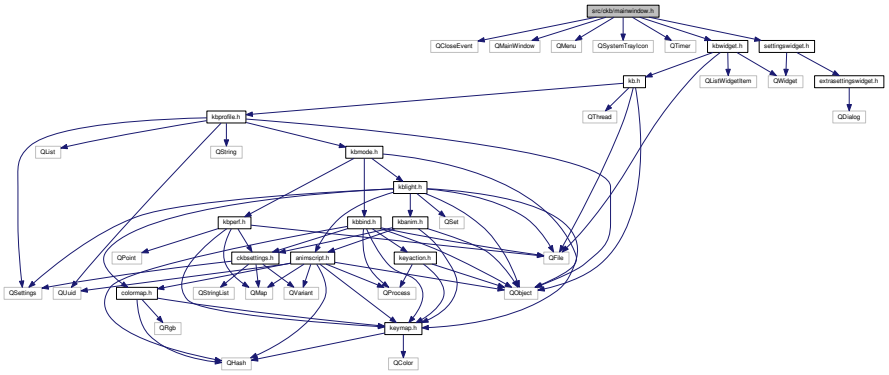
Referenced by MainWindow::MainWindow().

13.74 src/ckb/mainwindow.h File Reference

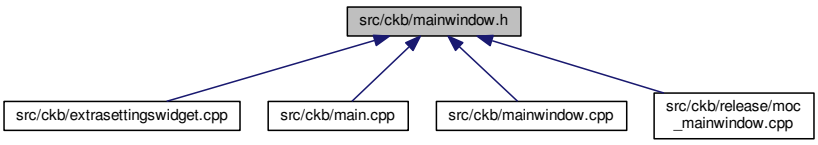
```
#include <QCloseEvent>
```

```
#include <QMainWindow>
#include <QMenu>
#include <QSystemTrayIcon>
#include <QTimer>
#include "kbwidget.h"
#include "settingswidget.h"
```

Include dependency graph for mainwindow.h:



This graph shows which files directly or indirectly include this file:



Data Structures

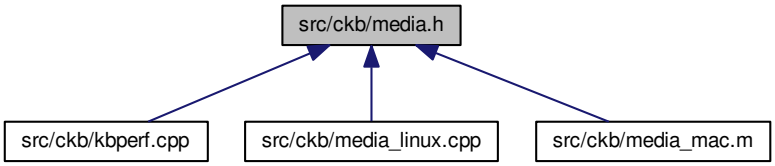
- class MainWindow

Namespaces

- Ui

13.75 src/ckb/media.h File Reference

This graph shows which files directly or indirectly include this file:



Macros

- #define EXTERN_C extern
- #define ENUM_C(name) typedef enum
- #define ENUM_END_C(name) name

Enumerations

- enum muteState { UNKNOWN = -1, UNMUTED, MUTED }

Functions

- muteState getMuteState ()

13.75.1 Macro Definition Documentation

13.75.1.1 #define ENUM_C(*name*) typedef enum

Definition at line 13 of file media.h.

13.75.1.2 #define ENUM_END_C(*name*) name

Definition at line 14 of file media.h.

13.75.1.3 #define EXTERN_C extern

Definition at line 12 of file media.h.

13.75.2 Enumeration Type Documentation

13.75.2.1 enum muteState

Enumerator

UNKNOWN

UNMUTED

MUTED

Definition at line 19 of file media.h.

```
19                  {
20     UNKNOWN = -1,
21     UNMUTED,
22     MUTED
23 } ENUM_END_C(muteState);
```

13.75.3 Function Documentation

13.75.3.1 muteState getMuteState ( )

Definition at line 102 of file media_linux.cpp.

References lastKnown, MUTED, UNKNOWN, and UNMUTED.

Referenced by KbPerf::applyIndicators().

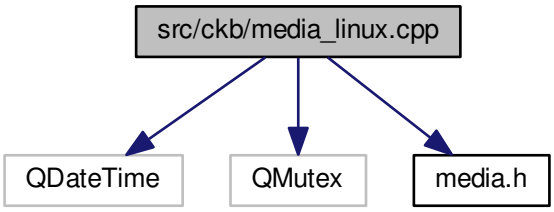
```
102                         {
103 #ifdef USE_LIBPULSE
104     static pa_threaded_mainloop* mainLoop = nullptr;
105
106     QMutexLocker locker(&mutex);
107
108     //Setup main loop thread used for communicating with Pulse Audio. All
109     //communication is done asynchronously.
110     if(mainLoop == nullptr){
111         mainLoop = pa_threaded_mainloop_new();
112         if(mainLoop == nullptr)
113             return lastKnown;
114
115         if(pa_threaded_mainloop_start(mainLoop) != 0){
116             pa_threaded_mainloop_free(mainLoop);
117             mainLoop = nullptr;
118             return lastKnown;
119         }
120     }
121
122     //Connect to the local Pulse Audio server. It's usually running but a
123     //reconnect is attempted periodically whenever the connection fails or is
124     //terminated.
125     if(paContext == nullptr && QDateTime::currentMSecsSinceEpoch() >= reconnectTime){
126         pa_threaded_mainloop_lock(mainLoop);
127         {
128             pa_mainloop_api* api = pa_threaded_mainloop_get_api(mainLoop);
129             Q_ASSERT(api != nullptr);
130
131             paContext = pa_context_new(api, "QPulse");
132             Q_ASSERT(paContext != nullptr);
133             pa_context_set_state_callback(paContext, &ContextStateCallback, nullptr);
134             pa_context_connect(paContext, nullptr, PA_CONTEXT_NOFAIL, nullptr);
135         }
136         pa_threaded_mainloop_unlock(mainLoop);
137     }
138 #endif
139
140     return lastKnown;
141 }
```

Here is the caller graph for this function:

13.76 src/ckb/media_linux.cpp File Reference

```
#include <QDateTime>
#include <QMutex>
#include "media.h"
```

Include dependency graph for media_linux.cpp:



Functions

- muteState getMuteState ()

Variables

- static muteState lastKnown = UNKNOWN

13.76.1 Function Documentation

13.76.1.1 muteState getMuteState ( )

Definition at line 102 of file media_linux.cpp.

References lastKnown.

```
102                         {
103 #ifdef USE_LIBPULSE
104     static pa_threaded_mainloop* mainLoop = nullptr;
105
106     QMutexLocker locker(&mutex);
107
108     //Setup main loop thread used for communicating with Pulse Audio. All
109     //communication is done asynchronously.
110     if(mainLoop == nullptr){
111         mainLoop = pa_threaded_mainloop_new();
112         if(mainLoop == nullptr)
113             return lastKnown;
114
115         if(pa_threaded_mainloop_start(mainLoop) != 0){
116             pa_threaded_mainloop_free(mainLoop);
117             mainLoop = nullptr;
118             return lastKnown;
119         }
120     }
121
122     //Connect to the local Pulse Audio server. It's usually running but a
123     //reconnect is attempted periodically whenever the connection fails or is
124     //terminated.
125     if(paContext == nullptr && QDateTime::currentMSecsSinceEpoch() >= reconnectTime){
126         pa_threaded_mainloop_lock(mainLoop);
127         {
128             pa_mainloop_api* api = pa_threaded_mainloop_get_api(mainLoop);
```

```
129             Q_ASSERT(api != nullptr);
130
131             paContext = pa_context_new(api, "QPulse");
132             Q_ASSERT(paContext != nullptr);
133             pa_context_set_state_callback(paContext, &ContextStateCallback, nullptr);
134             pa_context_connect(paContext, nullptr, PA_CONTEXT_NOFAIL, nullptr);
135         }
136         pa_threaded_mainloop_unlock(mainLoop);
137     }
138 #endif
139
140     return lastKnown;
141 }
```

13.76.2 Variable Documentation

13.76.2.1 muteState lastKnown = UNKNOWN [static]

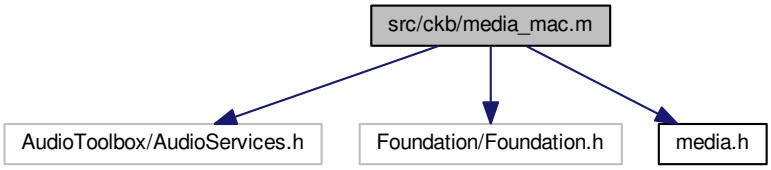
Definition at line 13 of file media_linux.cpp.

Referenced by getMuteState().

13.77 src/ckb/media_mac.m File Reference

```
#import <AudioToolbox/AudioServices.h>
#import <Foundation/Foundation.h>
#include "media.h"
```

Include dependency graph for media_mac.m:



Functions

- muteState getMuteState ()
- void disableAppNap ()

13.77.1 Function Documentation

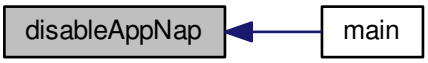
13.77.1.1 void disableAppNap ( )

Definition at line 34 of file media_mac.m.

Referenced by main().

```
34                  {
35     [[NSProcessInfo processInfo] beginActivityWithOptions:NSActivityUserInitiatedAllowingIdleSystemSleep
       reason:@"Keyboard animation"];
36 }
```

Here is the caller graph for this function:



13.77.1.2 muteState getMuteState ( )

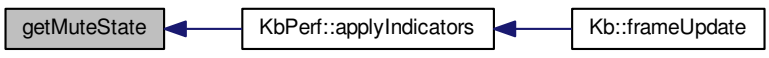
Definition at line 5 of file media_mac.m.

References MUTED, UNKNOWN, and UNMUTED.

Referenced by KbPerf::applyIndicators().

```
5                       {
6     // Get the system's default audio device
7     AudioObjectPropertyAddress propertyAddress = {
8         kAudioHardwarePropertyDefaultOutputDevice,
9         kAudioObjectPropertyScopeGlobal,
10         kAudioObjectPropertyElementMaster
11     };
12     AudioDeviceID deviceID = 0;
13     UInt32 dataSize = sizeof(deviceID);
14     if(AudioHardwareServiceGetPropertyData(kAudioObjectSystemObject, &propertyAddress, 0, NULL, &dataSize,
       &deviceID)
15             != kAudioHardwareNoError){
16         return UNKNOWN;
17     }
18     // Grab the mute property
19     AudioObjectPropertyAddress propertyAddress2 = {
20         kAudioDevicePropertyMute,
21         kAudioDevicePropertyScopeOutput,
22         kAudioObjectPropertyElementMaster
23     };
24     if(!AudioHardwareServiceHasProperty(deviceID, &propertyAddress2))
25         return UNKNOWN;
26     bool state = 0;
27     dataSize = sizeof(state);
28     if(AudioHardwareServiceGetPropertyData(deviceID, &propertyAddress2, 0, NULL, &dataSize, &state)
29             != kAudioHardwareNoError)
30         return UNKNOWN;
31     return state ? MUTED : UNMUTED;
32 }
```

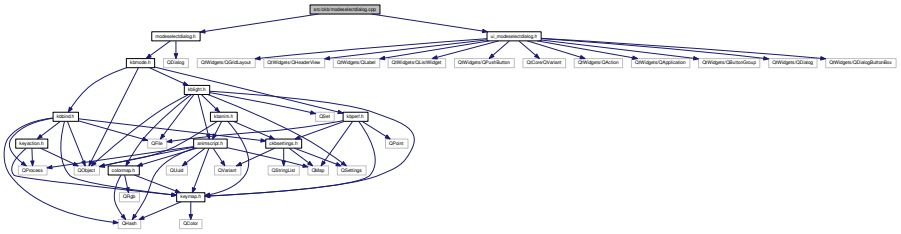
Here is the caller graph for this function:



13.78 src/ckb/modeselectdialog.cpp File Reference

```
#include "modeselectdialog.h"
#include "ui_modeselectdialog.h"
```

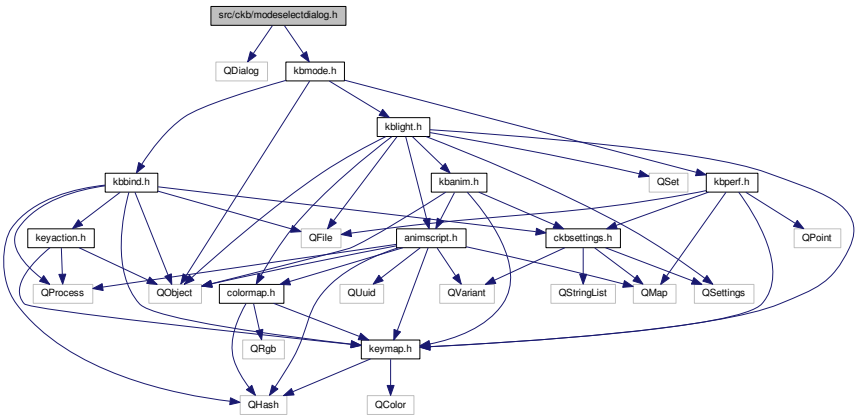
Include dependency graph for modeselectdialog.cpp:



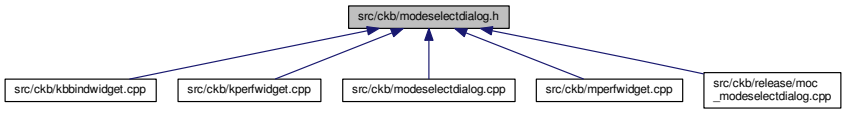
13.79 src/ckb/modeselectdialog.h File Reference

```
#include <QDialog>
#include "kbmode.h"
```

Include dependency graph for modeselectdialog.h:



This graph shows which files directly or indirectly include this file:



Data Structures

- class ModeSelectDialog

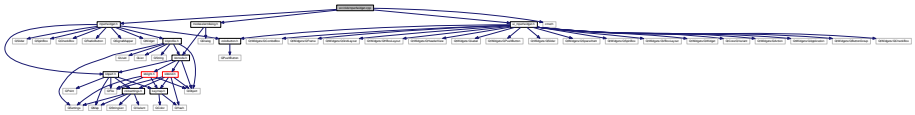
Namespaces

- Ui

13.80 src/ckb/mperfwidget.cpp File Reference

```
#include "mperfwidget.h"
#include "ui_mperfwidget.h"
#include "modeselectdialog.h"
#include <cmath>
```

Include dependency graph for mperfwidget.cpp:



Macros

- #define SET_START if(isSetting) return; isSetting = true
- #define SET_END isSetting = false

Functions

- int dpiExp (int value, int min, int max)
- int dpiLog (int value, int min, int max)
- int dpiRound (int value)

Variables

- static const QString xyLinkPath = "UI/DPI/UnlinkXY"
- static const double DPI_BASE = (double)KbPerf::DPI_MAX / (double)KbPerf::DPI_MIN

13.80.1 Macro Definition Documentation

13.80.1.1 #define SET_END isSetting = false

Definition at line 126 of file mperfwidget.cpp.

Referenced by MPerfWidget::boxXChanged(), MPerfWidget::boxYChanged(), MPerfWidget::sliderXMoved(), and MPerfWidget::sliderYMoved().

13.80.1.2 #define SET_START if(isSetting) return; isSetting = true

Definition at line 125 of file mperfwidget.cpp.

Referenced by MPerfWidget::boxXChanged(), MPerfWidget::boxYChanged(), MPerfWidget::sliderXMoved(), and MPerfWidget::sliderYMoved().

13.80.2 Function Documentation

13.80.2.1 int dpiExp (int *value*, int *min*, int *max*) [inline]

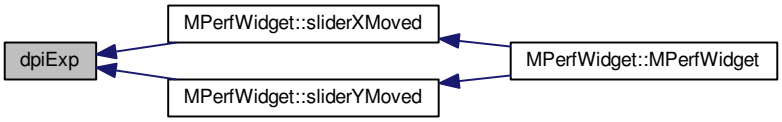
Definition at line 105 of file mperfwidget.cpp.

References DPI_BASE, and KbPerf::DPI_MIN.

Referenced by MPerfWidget::sliderXMoved(), and MPerfWidget::sliderYMoved().

```
105                                                 {
106     double val = (value - min) / (double)(max - min);
107     val = pow(DPI_BASE, val);
108     return round(KbPerf::DPI_MIN * val);
109 }
```

Here is the caller graph for this function:



13.80.2.2 int dpiLog (int *value*, int *min*, int *max*) [inline]

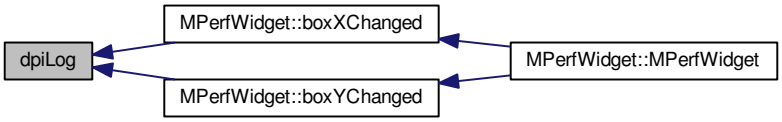
Definition at line 111 of file mperfwidget.cpp.

References DPI_BASE, and KbPerf::DPI_MIN.

Referenced by MPerfWidget::boxXChanged(), and MPerfWidget::boxYChanged().

```
111                                                 {
112     double val = value / (double)KbPerf::DPI_MIN;
113     val = log(val) / log(DPI_BASE);
114     return round(val * (max - min) + min);
115 }
```

Here is the caller graph for this function:



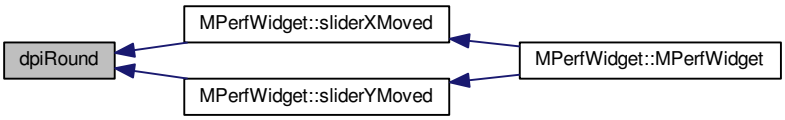
13.80.2.3 int dpiRound (int *value*) [inline]

Definition at line 117 of file mperfwidget.cpp.

Referenced by MPerfWidget::sliderXMoved(), and MPerfWidget::sliderYMoved().

```
117                                 {
118     if(value > 1000)
119         return round(value / 100.) * 100;
120     else
121         return round(value / 50.) * 50;
122 }
```

Here is the caller graph for this function:



13.80.3 Variable Documentation

13.80.3.1 const double DPI_BASE = (double)KbPerf::DPI_MAX / (double)KbPerf::DPI_MIN [static]

Definition at line 103 of file mperfwidget.cpp.

Referenced by dpiExp(), and dpiLog().

13.80.3.2 const QString xyLinkPath = "UI/DPI/UnlinkXY" [static]

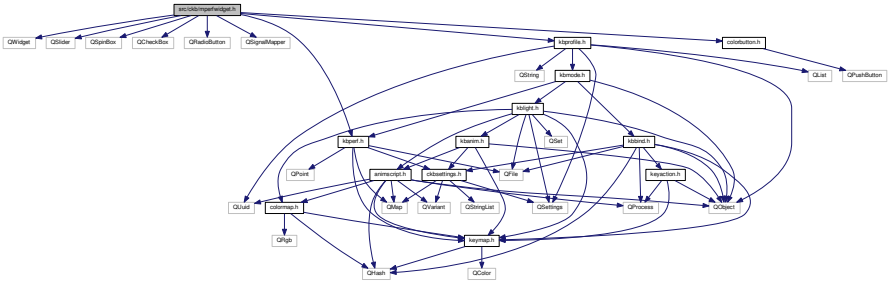
Definition at line 6 of file mperfwidget.cpp.

Referenced by MPerfWidget::on_xyBox_clicked().

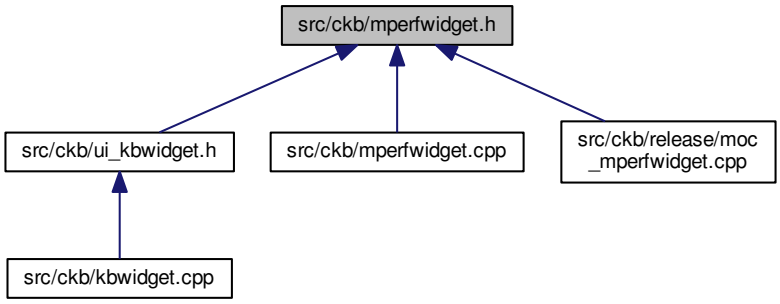
13.81 src/ckb/mperfwidget.h File Reference

```
#include <QWidget>
#include <QSlider>
#include <QSpinBox>
#include <QCheckBox>
#include <QRadioButton>
#include <QSignalMapper>
#include "kbperf.h"
#include "kbprofile.h"
#include "colorbutton.h"
```

Include dependency graph for mperfwidget.h:



This graph shows which files directly or indirectly include this file:



Data Structures

- class MPerfWidget
- struct MPerfWidget::DpiUi

Namespaces

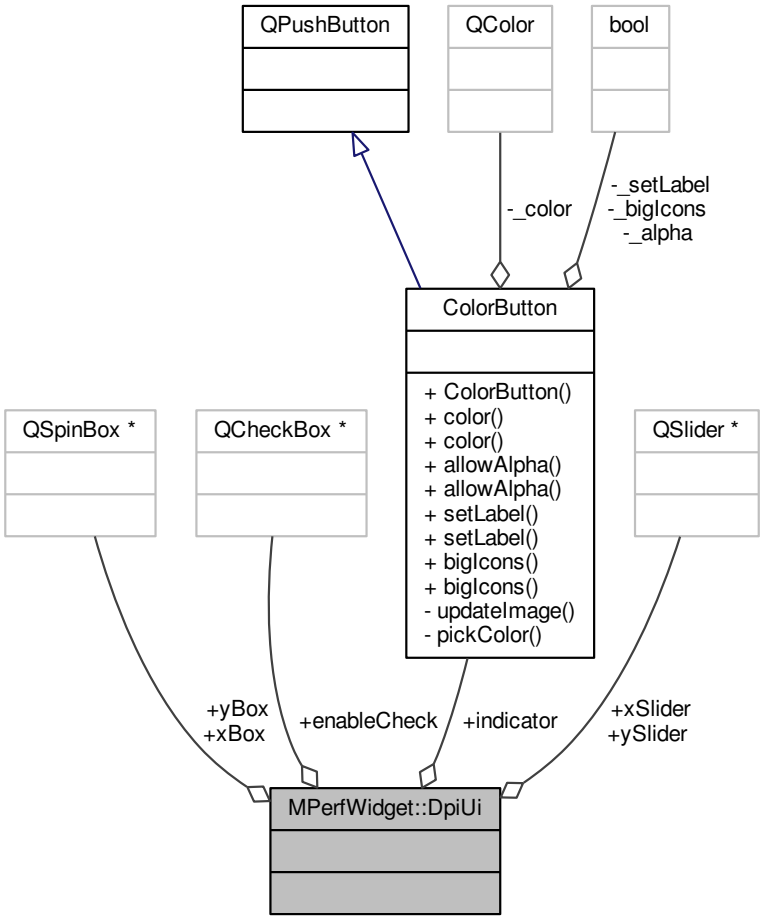
- Ui

13.81.1 Data Structure Documentation

13.81.1.1 struct MPerfWidget::DpiUi

Definition at line 37 of file mperfwidget.h.

Collaboration diagram for MPerfWidget::DpiUi:



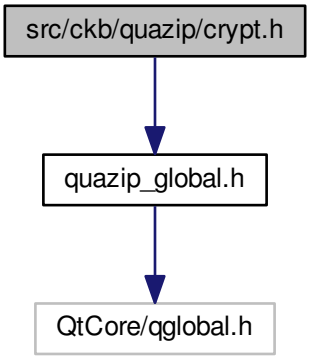
Data Fields

QCheckBox *	enableCheck	
ColorButton *	indicator	
QSpinBox *	xBox	
QSlider *	xSlider	
QSpinBox *	yBox	
QSlider *	ySlider	

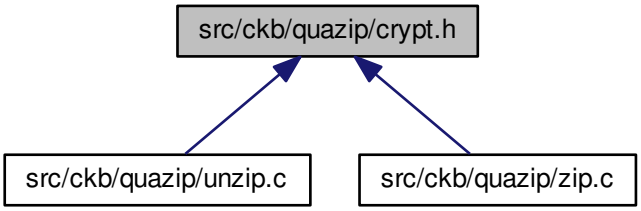
13.82 src/ckb/quazip/crypt.h File Reference

```
#include "quazip_global.h"
```

Include dependency graph for crypt.h:



This graph shows which files directly or indirectly include this file:



Macros

- #define CRC32(c, b) ((*(pcrc_32_tab+(((int)(c) ^ (b)) & 0xff))) ^ ((c) >> 8))
- #define zdecode(pkeys, pcrc_32_tab, c) (update_keys(pkeys,pcrc_32_tab,c ^= decrypt_byte(pkeys,pcrc_32_tab)))
- #define zencode(pkeys, pcrc_32_tab, c, t) (t=decrypt_byte(pkeys,pcrc_32_tab), update_keys(pkeys,pcrc_32_tab,c), t^(c))

Functions

- static int decrypt_byte (unsigned long *pkeys, const z_crc_t FAR *pcrc_32_tab)
- static int update_keys (unsigned long *pkeys, const z_crc_t FAR *pcrc_32_tab, int c)
- static void init_keys (const char *passwd, unsigned long *pkeys, const z_crc_t FAR *pcrc_32_tab)

13.82.1 Macro Definition Documentation

13.82.1.1 #define CRC32(c, b) ((*(pcrc_32_tab+(((int)(c) ^ (b)) & 0xff))) ^ ((c) >> 8))

Definition at line 32 of file crypt.h.

Referenced by update_keys().

13.82.1.2 #define zdecode(pkeys, pcrc_32_tab, c) (update_keys(pkeys,pcrc_32_tab,c ^= decrypt_byte(pkeys,pcrc_32_tab)))

Definition at line 79 of file crypt.h.

Referenced by unzOpenCurrentFile3(), and unzReadCurrentFile().

13.82.1.3 #define zencode(pkeys, pcrc_32_tab, c, t) (t=decrypt_byte(pkeys,pcrc_32_tab), update_keys(pkeys,pcrc_32_tab,c), t^(c))

Definition at line 82 of file crypt.h.

Referenced by zip64FlushWriteBuffer().

13.82.2 Function Documentation

13.82.2.1 static int decrypt_byte (unsigned long * pkeys, const z_crc_t FAR * pcrc_32_tab) [static]

Definition at line 37 of file crypt.h.

```
38 {
39     //(void) pcrc_32_tab; /* avoid "unused parameter" warning */
40     unsigned temp;  /* POTENTIAL BUG:  temp*(temp^1) may overflow in an
41                      * unpredictable manner on 16-bit systems; not a problem
42                      * with any known compiler so far, though */
43
44     temp = ((unsigned)(*(pkeys+2)) & 0xffff) | 2;
45     return (int)(((temp * (temp ^ 1)) >> 8) & 0xff);
46 }
```

13.82.2.2 static void init_keys (const char * passwd, unsigned long * pkeys, const z_crc_t FAR * pcrc_32_tab) [static]

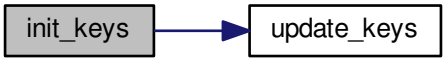
Definition at line 68 of file crypt.h.

References update_keys().

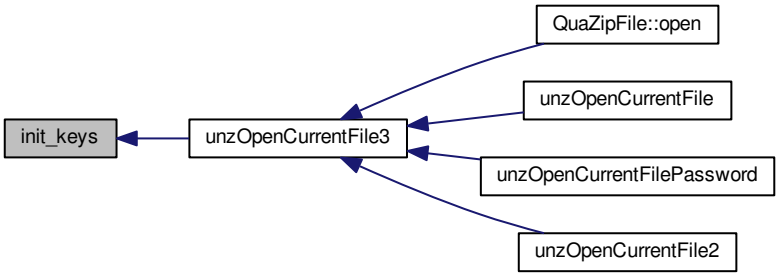
Referenced by unzOpenCurrentFile3().

```
69 {
70     *(pkeys+0) = 305419896L;
71     *(pkeys+1) = 591751049L;
72     *(pkeys+2) = 878082192L;
73     while (*passwd != '\0') {
74         update_keys(pkeys,pcrc_32_tab,(int)*passwd);
75         passwd++;
76     }
77 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



13.82.2.3 static int update_keys (unsigned long * *pkeys*, const z_crc_t FAR * *pcrc_32_tab*, int *c*) [static]

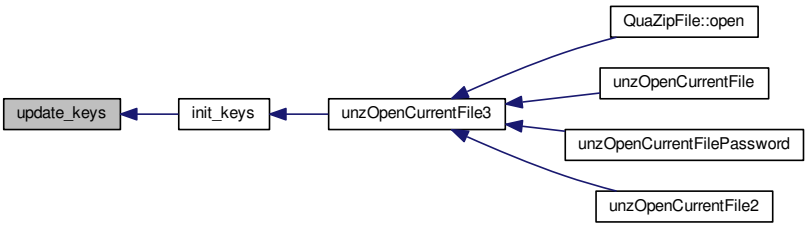
Definition at line 51 of file crypt.h.

References CRC32.

Referenced by init_keys().

```
52 {
53     (*(pkeys+0)) = CRC32((*(pkeys+0)), c);
54     (*(pkeys+1)) += (*(pkeys+0)) & 0xff;
55     (*(pkeys+1)) = (*(pkeys+1)) * 134775813L + 1;
56     {
57       register int keyshift = (int)((*(pkeys+1)) >> 24);
58       (*(pkeys+2)) = CRC32((*(pkeys+2)), keyshift);
59     }
60     return c;
61 }
```

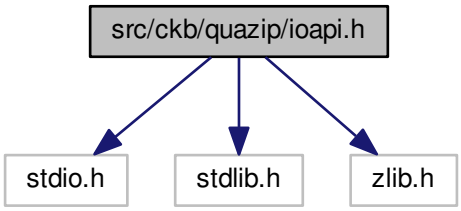
Here is the caller graph for this function:



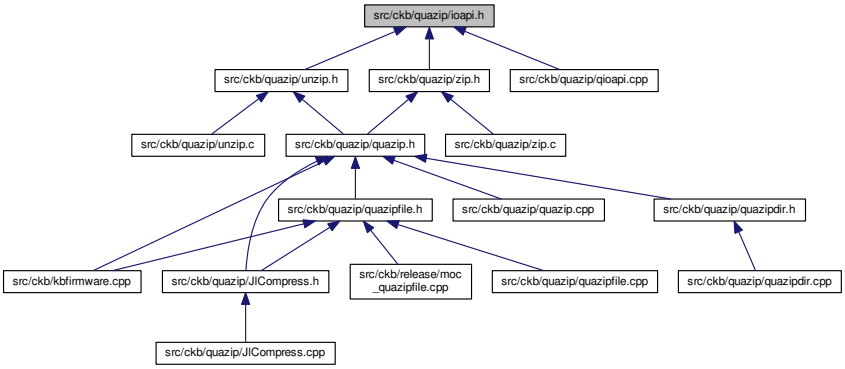
13.83 src/ckb/quazip/ioapi.h File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include "zlib.h"
```

Include dependency graph for ioapi.h:



This graph shows which files directly or indirectly include this file:



Data Structures

- struct zlib_filefunc_def_s
- struct zlib_filefunc64_def_s
- struct zlib_filefunc64_32_def_s

Macros

- #define __USE_FILE_OFFSET64
- #define __USE_LARGEFILE64
- #define _LARGEFILE64_SOURCE
- #define _FILE_OFFSET_BIT 64
- #define OF _Z_OF
- #define ZLIB_FILEFUNC_SEEK_CUR (1)
- #define ZLIB_FILEFUNC_SEEK_END (2)
- #define ZLIB_FILEFUNC_SEEK_SET (0)
- #define ZLIB_FILEFUNC_MODE_READ (1)
- #define ZLIB_FILEFUNC_MODE_WRITE (2)
- #define ZLIB_FILEFUNC_MODE_READWRITEFILTER (3)
- #define ZLIB_FILEFUNC_MODE_EXISTING (4)
- #define ZLIB_FILEFUNC_MODE_CREATE (8)
- #define ZCALLBACK
- #define ZREAD64(filefunc, filestream, buf, size) ((*((filefunc).zfile_func64.zread_file)) ((filefunc).zfile_-func64.opaque,filestream,buf,size))
- #define ZWRITE64(filefunc, filestream, buf, size) ((*((filefunc).zfile_func64.zwrite_file)) ((filefunc).zfile_-func64.opaque,filestream,buf,size))
- #define ZCLOSE64(filefunc, filestream) ((*((filefunc).zfile_func64.zclose_file)) ((filefunc).zfile_func64.-opaque,filestream))
- #define ZFAKECLOSE64(filefunc, filestream) ((*((filefunc).zfile_func64.zfakeclose_file)) ((filefunc).zfile_-func64.opaque,filestream))
- #define ZERROR64(filefunc, filestream) ((*((filefunc).zfile_func64.zerror_file)) ((filefunc).zfile_func64.-opaque,filestream))
- #define ZOPEN64(filefunc, filename, mode) (call_zopen64((&(filefunc)),(filename),(mode)))
- #define ZTELL64(filefunc, filestream) (call_ztell64((&(filefunc)),(filestream)))
- #define ZSEEK64(filefunc, filestream, pos, mode) (call_zseek64((&(filefunc)),(filestream),(pos),(mode)))

Typedefs

- typedef unsigned long long int ZPOS64_T
- typedef voidpf open_file_func _Z_OF ((voidpf opaque, voidpf file, int mode))

  Addition for GDAL : START.
- typedef struct zlib_filefunc_def_s zlib_filefunc_def
- typedef struct zlib_filefunc64_def_s zlib_filefunc64_def
- typedef struct zlib_filefunc64_32_def_s zlib_filefunc64_32_def

Functions

- void fill_qiodevice64_filefunc _Z_OF ((zlib_filefunc64_def *pzlib_filefunc_def))
- void fill_qiodevice_filefunc _Z_OF ((zlib_filefunc_def *pzlib_filefunc_def))
- voidpf call_zopen64 _Z_OF ((const zlib_filefunc64_32_def *pfilefunc, voidpf file, int mode))
- int call_zseek64 _Z_OF ((const zlib_filefunc64_32_def *pfilefunc, voidpf filestream, ZPOS64_T offset, int origin))

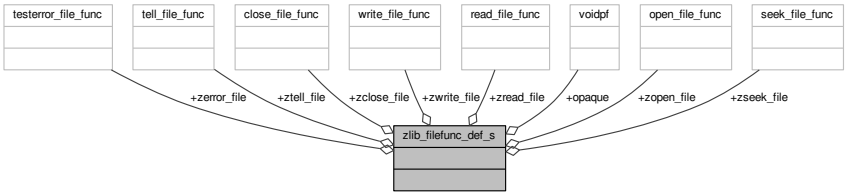
- ZPOS64_T call_ztell64 _Z_OF ((const zlib_filefunc64_32_def ∗pfilefunc, voidpf filestream))
- void fill_zlib_filefunc64_32_def_from_filefunc32 (zlib_filefunc64_32_def ∗p_filefunc64_32, const zlib_filefunc_def ∗p_filefunc32)

13.83.1 Data Structure Documentation

13.83.1.1 struct zlib_filefunc_def_s

Definition at line 143 of file ioapi.h.

Collaboration diagram for zlib_filefunc_def_s:



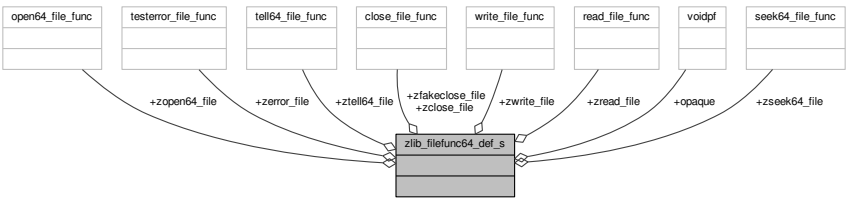
Data Fields

voidpf	opaque	
close_file_func	zclose_file	
testerror_file_func	zerror_file	
open_file_func	zopen_file	
read_file_func	zread_file	
seek_file_func	zseek_file	
tell_file_func	ztell_file	
write_file_func	zwrite_file	

13.83.1.2 struct zlib_filefunc64_def_s

Definition at line 159 of file ioapi.h.

Collaboration diagram for zlib_filefunc64_def_s:



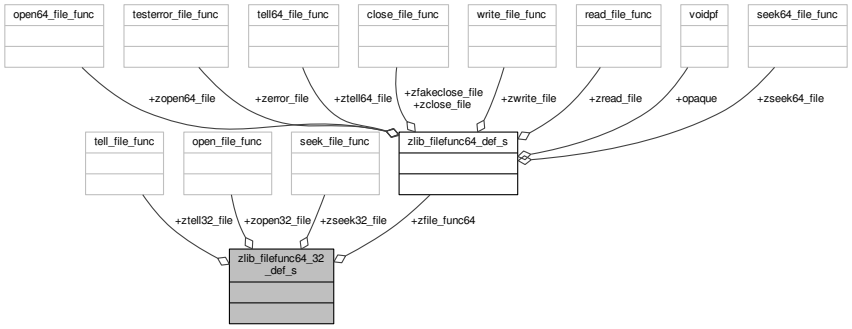
Data Fields

voidpf	opaque	
close_file_func	zclose_file	
testerror_file_-func	zerror_file	
close_file_func	zfakeclose_file	
open64_file_-func	zopen64_file	
read_file_func	zread_file	
seek64_file_func	zseek64_file	
tell64_file_func	ztell64_file	
write_file_func	zwrite_file	

13.83.1.3 struct zlib_filefunc64_32_def_s

Definition at line 176 of file ioapi.h.

Collaboration diagram for zlib_filefunc64_32_def_s:



Data Fields

zlib_filefunc64_-def	zfile_func64	
open_file_func	zopen32_file	
seek_file_func	zseek32_file	
tell_file_func	ztell32_file	

13.83.2 Macro Definition Documentation

13.83.2.1 #define __USE_FILE_OFFSET64

Definition at line 32 of file ioapi.h.

13.83.2.2 #define __USE_LARGEFILE64

Definition at line 35 of file ioapi.h.

13.83.2.3 #define _FILE_OFFSET_BIT 64

Definition at line 41 of file ioapi.h.

13.83.2.4 #define _LARGEFILE64_SOURCE

Definition at line 38 of file ioapi.h.

13.83.2.5 #define OF _Z_OF

Definition at line 106 of file ioapi.h.

13.83.2.6 #define ZCALLBACK

Definition at line 125 of file ioapi.h.

13.83.2.7 #define ZCLOSE64(*filefunc, filestream*) ((∗((filefunc).zfile_func64.zclose_file)) ((filefunc).zfile_func64.opaque,filestream))

Definition at line 189 of file ioapi.h.

Referenced by LoadCentralDirectoryRecord(), unzClose(), unzOpenInternal(), zipClose(), and zipOpen3().

13.83.2.8 #define ZERROR64(*filefunc, filestream*) ((∗((filefunc).zfile_func64.zerror_file)) ((filefunc).zfile_func64.opaque,filestream))

Definition at line 191 of file ioapi.h.

Referenced by unz64local_getByte(), and zip64local_getByte().

13.83.2.9 #define ZFAKECLOSE64(*filefunc, filestream*) ((∗((filefunc).zfile_func64.zfakeclose_file)) ((filefunc).zfile_func64.opaque,filestream))

Definition at line 190 of file ioapi.h.

Referenced by LoadCentralDirectoryRecord(), unzClose(), unzOpenInternal(), zipClose(), and zipOpen3().

13.83.2.10 #define ZLIB_FILEFUNC_MODE_CREATE (8)

Definition at line 118 of file ioapi.h.

Referenced by qiodevice_open_file_func(), and zipOpen3().

13.83.2.11 #define ZLIB_FILEFUNC_MODE_EXISTING (4)

Definition at line 117 of file ioapi.h.

Referenced by qiodevice_open_file_func(), unzOpenInternal(), and zipOpen3().

13.83.2.12 #define ZLIB_FILEFUNC_MODE_READ (1)

Definition at line 113 of file ioapi.h.

Referenced by qiodevice_open_file_func(), unzOpenInternal(), and zipOpen3().

13.83.2.13 #define ZLIB_FILEFUNC_MODE_READWRITEFILTER (3)

Definition at line 115 of file ioapi.h.

Referenced by qiodevice_open_file_func().

13.83.2.14 #define ZLIB_FILEFUNC_MODE_WRITE (2)

Definition at line 114 of file ioapi.h.

Referenced by zipOpen3().

13.83.2.15 #define ZLIB_FILEFUNC_SEEK_CUR (1)

Definition at line 109 of file ioapi.h.

Referenced by qiodevice64_seek_file_func(), qiodevice_seek_file_func(), and unz64local_GetCurrentFileInfoInternal().

13.83.2.16 #define ZLIB_FILEFUNC_SEEK_END (2)

Definition at line 110 of file ioapi.h.

Referenced by qiodevice64_seek_file_func(), qiodevice_seek_file_func(), unz64local_SearchCentralDir(), unz64local_SearchCentralDir64(), zip64local_SearchCentralDir(), and zip64local_SearchCentralDir64().

13.83.2.17 #define ZLIB_FILEFUNC_SEEK_SET (0)

Definition at line 111 of file ioapi.h.

Referenced by LoadCentralDirectoryRecord(), qiodevice64_seek_file_func(), qiodevice_seek_file_func(), unz64local_CheckCurrentFileCoherencyHeader(), unz64local_GetCurrentFileInfoInternal(), unz64local_SearchCentralDir(), unz64local_SearchCentralDir64(), unzGetGlobalComment(), unzGetLocalExtrafield(), unzOpenInternal(), unzReadCurrentFile(), zip64local_SearchCentralDir(), zip64local_SearchCentralDir64(), and zipCloseFileInZipRaw64().

13.83.2.18 #define ZOPEN64(*filefunc, filename, mode*) (call_zopen64((&(filefunc)),(filename),(mode)))

Definition at line 199 of file ioapi.h.

Referenced by unzOpenInternal(), and zipOpen3().

13.83.2.19 #define ZREAD64(*filefunc, filestream, buf, size*) ((∗((filefunc).zfile_func64.zread_file)) ((filefunc).zfile_func64.opaque,filestream,buf,size))

Definition at line 185 of file ioapi.h.

Referenced by LoadCentralDirectoryRecord(), unz64local_getByte(), unz64local_GetCurrentFileInfoInternal(), unz64local_SearchCentralDir(), unz64local_SearchCentralDir64(), unzGetGlobalComment(), unzGetLocalExtrafield(), unzOpenCurrentFile3(), unzReadCurrentFile(), zip64local_getByte(), zip64local_SearchCentralDir(), and zip64local_SearchCentralDir64().

13.83.2.20 #define ZSEEK64(*filefunc, filestream, pos, mode*) (call_zseek64((&(filefunc)),(filestream),(pos),(mode)))

Definition at line 201 of file ioapi.h.

Referenced by LoadCentralDirectoryRecord(), unz64local_CheckCurrentFileCoherencyHeader(), unz64local_GetCurrentFileInfoInternal(), unz64local_SearchCentralDir(), unz64local_SearchCentralDir64(), unzGetGlobalComment(), unzGetLocalExtrafield(), unzOpenCurrentFile3(), unzOpenInternal(), unzReadCurrentFile(), zip64local_SearchCentralDir(), zip64local_SearchCentralDir64(), zipCloseFileInZipRaw64(), and zipOpen3().

13.83.2.21 #define ZTELL64(*filefunc, filestream*) (call_ztell64((&(filefunc)),(filestream)))

Definition at line 200 of file ioapi.h.

Referenced by unz64local_SearchCentralDir(), unz64local_SearchCentralDir64(), Write_LocalFileHeader(), zip64local_SearchCentralDir(), zip64local_SearchCentralDir64(), zipClose(), zipCloseFileInZipRaw64(), zipOpen3(), and zipOpenNewFileInZip4_64().

13.83.2.22 #define ZWRITE64(*filefunc, filestream, buf, size*) ((∗((filefunc).zfile_func64.zwrite_file)) ((filefunc).zfile_func64.opaque,filestream,buf,size))

Definition at line 186 of file ioapi.h.

Referenced by Write_GlobalComment(), Write_LocalFileHeader(), zip64FlushWriteBuffer(), zip64local_putValue(), zipClose(), and zipOpenNewFileInZip4_64().

13.83.3 Typedef Documentation

13.83.3.1 zipFile ZEXPORT zipOpen64 _Z_OF ((voidpf opaque, voidpf file, int mode))

Addition for GDAL : END.

Definition at line 132 of file ioapi.h.

13.83.3.2 typedef struct zlib_filefunc64_32_def_s zlib_filefunc64_32_def

13.83.3.3 typedef struct zlib_filefunc64_def_s zlib_filefunc64_def

13.83.3.4 typedef struct zlib_filefunc_def_s zlib_filefunc_def

13.83.3.5 typedef unsigned long long int ZPOS64_T

Definition at line 94 of file ioapi.h.

13.83.4 Function Documentation

13.83.4.1 void fill_qiodevice64_filefunc _Z_OF ((zlib_filefunc64_def ∗pzlib_filefunc_def))

13.83.4.2 void fill_qiodevice_filefunc _Z_OF ((zlib_filefunc_def ∗pzlib_filefunc_def))

13.83.4.3 voidpf call_zopen64 _Z_OF ((const zlib_filefunc64_32_def ∗pfilefunc, voidpf file, int mode))

13.83.4.4 int call_zseek64 _Z_OF ((const zlib_filefunc64_32_def ∗pfilefunc, voidpf filestream, ZPOS64_T offset, int origin))

13.83.4.5 ZPOS64_T call_ztell64 _Z_OF ((const zlib_filefunc64_32_def ∗pfilefunc, voidpf filestream))

13.83.4.6 void fill_zlib_filefunc64_32_def_from_filefunc32 (zlib_filefunc64_32_def * *p_filefunc64_32,* const zlib_filefunc_def * *p_filefunc32*)

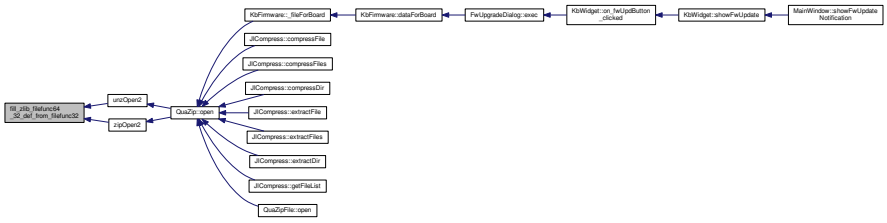
Definition at line 346 of file qioapi.cpp.

References zlib_filefunc_def_s::opaque, zlib_filefunc64_def_s::opaque, zlib_filefunc_def_s::zclose_file, zlib_filefunc64_def_s::zclose_file, zlib_filefunc_def_s::zerror_file, zlib_filefunc64_def_s::zerror_file, zlib_filefunc64_def_s::zfakeclose_file, zlib_filefunc64_32_def_s::zfile_func64, zlib_filefunc64_32_def_s::zopen32_file, zlib_filefunc64_def_s::zopen64_file, zlib_filefunc_def_s::zopen_file, zlib_filefunc_def_s::zread_file, zlib_filefunc64_def_s::zread_file, zlib_filefunc64_32_def_s::zseek32_file, zlib_filefunc64_def_s::zseek64_file, zlib_filefunc_def_s::zseek_file, zlib_filefunc64_32_def_s::ztell32_file, zlib_filefunc64_def_s::ztell64_file, zlib_filefunc_def_s::ztell_file, zlib_filefunc_def_s::zwrite_file, and zlib_filefunc64_def_s::zwrite_file.

Referenced by unzOpen2(), and zipOpen2().

```
347 {
348     p_filefunc64_32->zfile_func64.zopen64_file = NULL;
349     p_filefunc64_32->zopen32_file = p_filefunc32->zopen_file;
350     p_filefunc64_32->zfile_func64.zerror_file = p_filefunc32->
    zerror_file;
351     p_filefunc64_32->zfile_func64.zread_file = p_filefunc32->
    zread_file;
352     p_filefunc64_32->zfile_func64.zwrite_file = p_filefunc32->
    zwrite_file;
353     p_filefunc64_32->zfile_func64.ztell64_file = NULL;
354     p_filefunc64_32->zfile_func64.zseek64_file = NULL;
355     p_filefunc64_32->zfile_func64.zclose_file = p_filefunc32->
    zclose_file;
356     p_filefunc64_32->zfile_func64.zerror_file = p_filefunc32->
    zerror_file;
357     p_filefunc64_32->zfile_func64.opaque = p_filefunc32->
    opaque;
358     p_filefunc64_32->zfile_func64.zfakeclose_file = NULL;
359     p_filefunc64_32->zseek32_file = p_filefunc32->zseek_file;
360     p_filefunc64_32->ztell32_file = p_filefunc32->ztell_file;
361 }
```

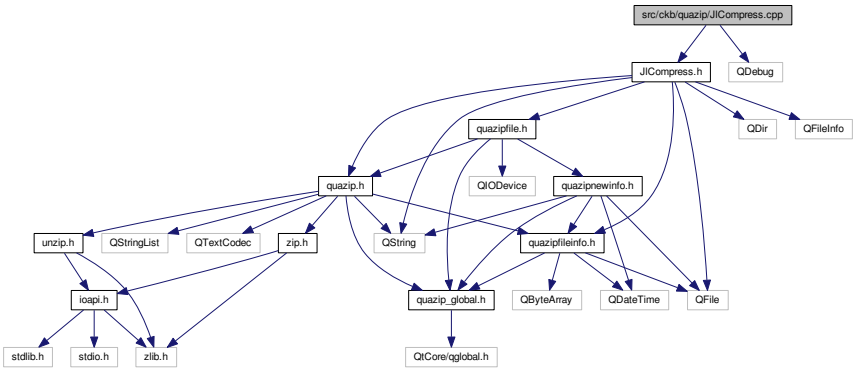
Here is the caller graph for this function:



13.84 src/ckb/quazip/JlCompress.cpp File Reference

```
#include "JlCompress.h"
#include <QDebug>
```

Include dependency graph for JlCompress.cpp:



Functions

- static bool copyData (QIODevice &inFile, QIODevice &outFile)

13.84.1 Function Documentation

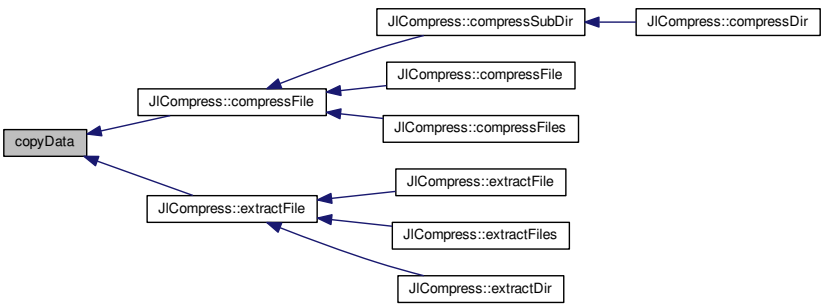
13.84.1.1 static bool copyData (QIODevice & *inFile,* QIODevice & *outFile*) [static]

Definition at line 29 of file JlCompress.cpp.

Referenced by JlCompress::compressFile(), and JlCompress::extractFile().

```
30 {
31     while (!inFile.atEnd()) {
32         char buf[4096];
33         qint64 readLen = inFile.read(buf, 4096);
34         if (readLen <= 0)
35             return false;
36         if (outFile.write(buf, readLen) != readLen)
37             return false;
38     }
39     return true;
40 }
```

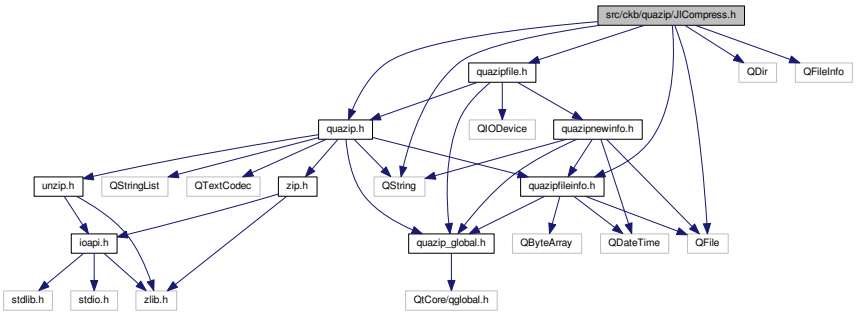
Here is the caller graph for this function:



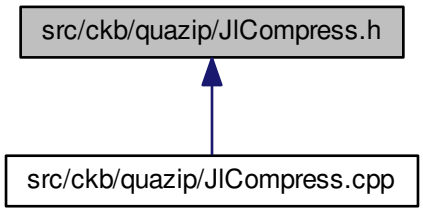
13.85 src/ckb/quazip/JlCompress.h File Reference

```
#include "quazip.h"
#include "quazipfile.h"
#include "quazipfileinfo.h"
#include <QString>
#include <QDir>
#include <QFileInfo>
#include <QFile>
```

Include dependency graph for JlCompress.h:



This graph shows which files directly or indirectly include this file:



Data Structures

- class JlCompress

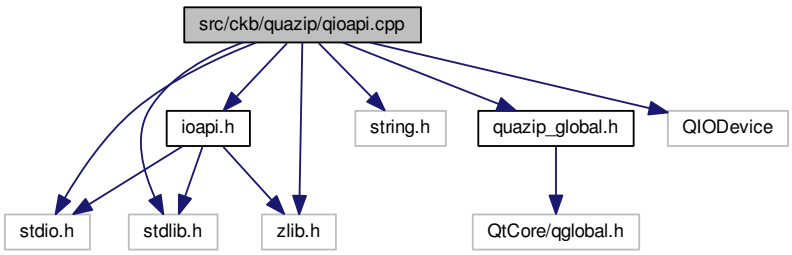
  Utility class for typical operations.

13.86 src/ckb/quazip/qioapi.cpp File Reference

```
#include <stdio.h>
```

```
#include <stdlib.h>
#include <string.h>
#include "zlib.h"
#include "ioapi.h"
#include "quazip_global.h"
#include <QIODevice>
```

Include dependency graph for qioapi.cpp:



Data Structures

- struct QIODevice_descriptor

Macros

- #define SEEK_CUR 1
- #define SEEK_END 2
- #define SEEK_SET 0

Functions

- voidpf call_zopen64 (const zlib_filefunc64_32_def *pfilefunc, voidpf file, int mode)
- int call_zseek64 (const zlib_filefunc64_32_def *pfilefunc, voidpf filestream, ZPOS64_T offset, int origin)
- ZPOS64_T call_ztell64 (const zlib_filefunc64_32_def *pfilefunc, voidpf filestream)
- voidpf qiodevice_open_file_func (voidpf opaque, voidpf file, int mode)
- uLong qiodevice_read_file_func (voidpf opaque, voidpf stream, void *buf, uLong size)
- uLong qiodevice_write_file_func (voidpf opaque, voidpf stream, const void *buf, uLong size)
- uLong qiodevice_tell_file_func (voidpf opaque, voidpf stream)
- ZPOS64_T qiodevice64_tell_file_func (voidpf opaque, voidpf stream)
- int qiodevice_seek_file_func (voidpf, voidpf stream, uLong offset, int origin)
- int qiodevice64_seek_file_func (voidpf, voidpf stream, ZPOS64_T offset, int origin)
- int qiodevice_close_file_func (voidpf opaque, voidpf stream)
- int qiodevice_fakeclose_file_func (voidpf opaque, voidpf)
- int qiodevice_error_file_func (voidpf, voidpf)
- void fill_qiodevice_filefunc (zlib_filefunc_def *pzlib_filefunc_def)
- void fill_qiodevice64_filefunc (zlib_filefunc64_def *pzlib_filefunc_def)
- void fill_zlib_filefunc64_32_def_from_filefunc32 (zlib_filefunc64_32_def *p_filefunc64_32, const zlib_filefunc_def *p_filefunc32)

13.86.1 Macro Definition Documentation

13.86.1.1 #define SEEK_CUR 1

Definition at line 29 of file qioapi.cpp.

13.86.1.2 #define SEEK_END 2

Definition at line 33 of file qioapi.cpp.

13.86.1.3 #define SEEK_SET 0

Definition at line 37 of file qioapi.cpp.

Referenced by unzOpenCurrentFile3().

13.86.2 Function Documentation

13.86.2.1 voidpf call_zopen64 (const zlib_filefunc64_32_def * *pfilefunc,* voidpf *file,* int *mode*)

Definition at line 40 of file qioapi.cpp.

References zlib_filefunc64_def_s::opaque, zlib_filefunc64_32_def_s::zfile_func64, zlib_filefunc64_32_def_s::zopen32_file, and zlib_filefunc64_def_s::zopen64_file.

```
41 {
42     if (pfilefunc->zfile_func64.zopen64_file != NULL)
43         return (*(pfilefunc->zfile_func64.zopen64_file)) (pfilefunc->
      zfile_func64.opaque,file,mode);
44     else
45     {
46         return (*(pfilefunc->zopen32_file))(pfilefunc->zfile_func64.
      opaque,file,mode);
47     }
48 }
```

13.86.2.2 int call_zseek64 (const zlib_filefunc64_32_def * *pfilefunc,* voidpf *filestream,* ZPOS64_T *offset,* int *origin*)

Definition at line 50 of file qioapi.cpp.

References zlib_filefunc64_def_s::opaque, zlib_filefunc64_32_def_s::zfile_func64, zlib_filefunc64_32_def_s::zseek32_file, and zlib_filefunc64_def_s::zseek64_file.

```
51 {
52     if (pfilefunc->zfile_func64.zseek64_file != NULL)
53         return (*(pfilefunc->zfile_func64.zseek64_file)) (pfilefunc->
      zfile_func64.opaque,filestream,offset,origin);
54     else
55     {
56         uLong offsetTruncated = (uLong)offset;
57         if (offsetTruncated != offset)
58             return -1;
59         else
60             return (*(pfilefunc->zseek32_file))(pfilefunc->
      zfile_func64.opaque,filestream,offsetTruncated,origin);
61     }
62 }
```

13.86.2.3 ZPOS64_T call_ztell64 (const zlib_filefunc64_32_def * *pfilefunc,* voidpf *filestream*)

Definition at line 64 of file qioapi.cpp.

References zlib_filefunc64_def_s::opaque, zlib_filefunc64_32_def_s::zfile_func64, zlib_filefunc64_def_s::zseek64_file, zlib_filefunc64_32_def_s::ztell32_file, and zlib_filefunc64_def_s::ztell64_file.

```
65 {
66     if (pfilefunc->zfile_func64.zseek64_file != NULL)
67         return (*(pfilefunc->zfile_func64.ztell64_file)) (pfilefunc->
      zfile_func64.opaque,filestream);
68     else
69     {
70         uLong tell_uLong = (*(pfilefunc->ztell32_file))(pfilefunc->
      zfile_func64.opaque,filestream);
71         if ((tell_uLong) == ((uLong)-1))
72             return (ZPOS64_T)-1;
73         else
74             return tell_uLong;
75     }
76 }
```

13.86.2.4 void fill_qiodevice64_filefunc (zlib_filefunc64_def * *pzlib_filefunc_def*)

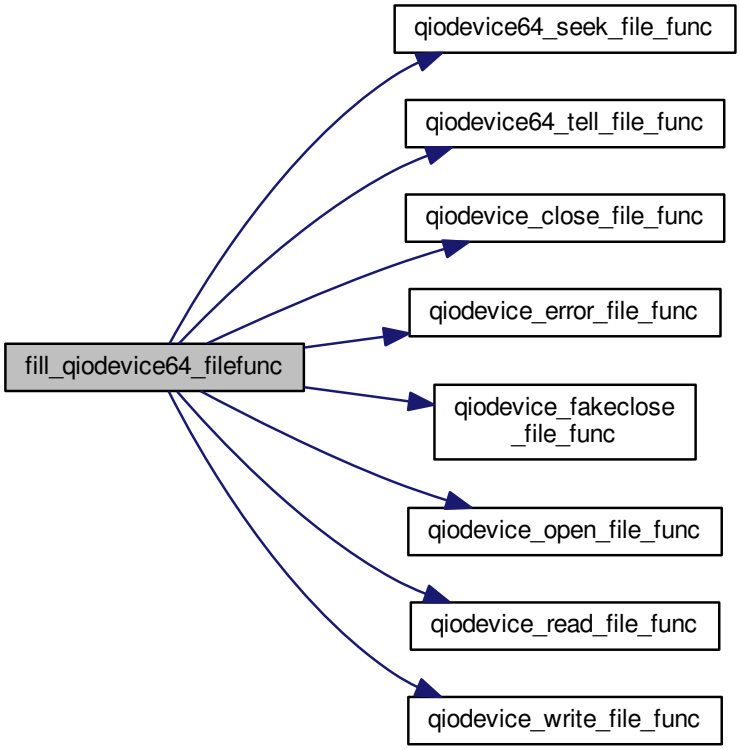
Definition at line 331 of file qioapi.cpp.

References zlib_filefunc64_def_s::opaque, qiodevice64_seek_file_func(), qiodevice64_tell_file_func(), qiodevice_close_file_func(), qiodevice_error_file_func(), qiodevice_fakeclose_file_func(), qiodevice_open_file_func(), qiodevice_read_file_func(), qiodevice_write_file_func(), zlib_filefunc64_def_s::zclose_file, zlib_filefunc64_def_s::zerror_file, zlib_filefunc64_def_s::zfakeclose_file, zlib_filefunc64_def_s::zopen64_file, zlib_filefunc64_def_s::zread_file, zlib_filefunc64_def_s::zseek64_file, zlib_filefunc64_def_s::ztell64_file, and zlib_filefunc64_def_s::zwrite_file.

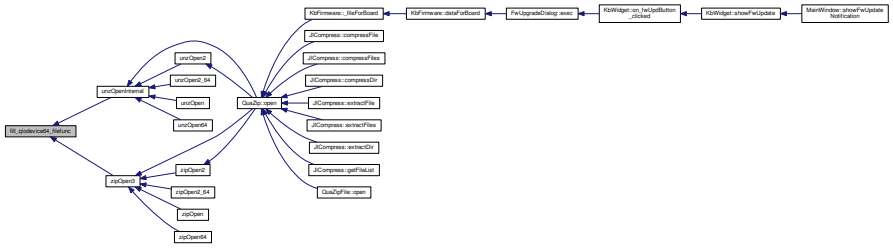
Referenced by unzOpenInternal(), and zipOpen3().

```
333 {
334     // Open functions are the same for Qt.
335     pzlib_filefunc_def->zopen64_file = qiodevice_open_file_func;
336     pzlib_filefunc_def->zread_file = qiodevice_read_file_func;
337     pzlib_filefunc_def->zwrite_file = qiodevice_write_file_func;
338     pzlib_filefunc_def->ztell64_file = qiodevice64_tell_file_func;
339     pzlib_filefunc_def->zseek64_file = qiodevice64_seek_file_func;
340     pzlib_filefunc_def->zclose_file = qiodevice_close_file_func;
341     pzlib_filefunc_def->zerror_file = qiodevice_error_file_func;
342     pzlib_filefunc_def->opaque = new QIODevice_descriptor;
343     pzlib_filefunc_def->zfakeclose_file =
      qiodevice_fakeclose_file_func;
344 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



13.86.2.5 void fill_qiodevice_filefunc (zlib_filefunc_def * *pzlib_filefunc_def*)

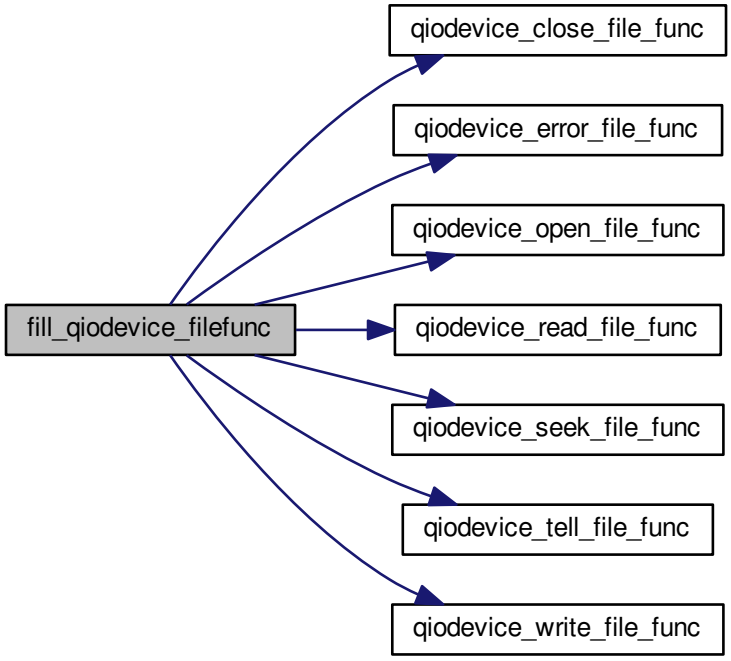
Definition at line 318 of file qioapi.cpp.

References zlib_filefunc_def_s::opaque, qiodevice_close_file_func(), qiodevice_error_file_func(), qiodevice_open_file_func(), qiodevice_read_file_func(), qiodevice_seek_file_func(), qiodevice_tell_file_func(), qiodevice_write_file_func(), zlib_filefunc_def_s::zclose_file, zlib_filefunc_def_s::zerror_file, zlib_filefunc_def_s::zopen_file, zlib_filefunc_def_s::zread_file, zlib_filefunc_def_s::zseek_file, zlib_filefunc_def_s::ztell_file, and zlib_filefunc_def_s::zwrite_file.

```
320 {
```

```
321     pzlib_filefunc_def->zopen_file = qiodevice_open_file_func;
322     pzlib_filefunc_def->zread_file = qiodevice_read_file_func;
323     pzlib_filefunc_def->zwrite_file = qiodevice_write_file_func;
324     pzlib_filefunc_def->ztell_file = qiodevice_tell_file_func;
325     pzlib_filefunc_def->zseek_file = qiodevice_seek_file_func;
326     pzlib_filefunc_def->zclose_file = qiodevice_close_file_func;
327     pzlib_filefunc_def->zerror_file = qiodevice_error_file_func;
328     pzlib_filefunc_def->opaque = new QIODevice_descriptor;
329 }
```

Here is the call graph for this function:



13.86.2.6 void fill_zlib_filefunc64_32_def_from_filefunc32 (zlib_filefunc64_32_def * *p_filefunc64_32*, const zlib_filefunc_def * *p_filefunc32*)

Definition at line 346 of file qioapi.cpp.

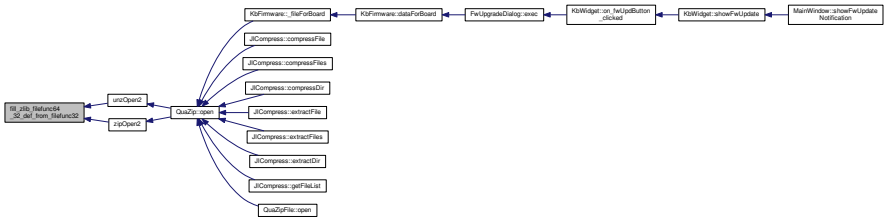
References zlib_filefunc_def_s::opaque, zlib_filefunc64_def_s::opaque, zlib_filefunc_def_s::zclose_file, zlib_filefunc64_def_s::zclose_file, zlib_filefunc_def_s::zerror_file, zlib_filefunc64_def_s::zerror_file, zlib_filefunc64_def_s::zfakeclose_file, zlib_filefunc64_32_def_s::zfile_func64, zlib_filefunc64_32_def_s::zopen32_file, zlib_filefunc64_def_s::zopen64_file, zlib_filefunc_def_s::zopen_file, zlib_filefunc_def_s::zread_file, zlib_filefunc64_def_s::zread_file, zlib_filefunc64_32_def_s::zseek32_file, zlib_filefunc64_def_s::zseek64_file, zlib_filefunc_def_s::zseek_file, zlib_filefunc64_32_def_s::ztell32_file, zlib_filefunc64_def_s::ztell64_file, zlib_filefunc_def_s::ztell_file, zlib_filefunc_def_s::zwrite_file, and zlib_filefunc64_def_s::zwrite_file.

Referenced by unzOpen2(), and zipOpen2().

```
347 {
348     p_filefunc64_32->zfile_func64.zopen64_file = NULL;
349     p_filefunc64_32->zopen32_file = p_filefunc32->zopen_file;
350     p_filefunc64_32->zfile_func64.zerror_file = p_filefunc32->
      zerror_file;
351     p_filefunc64_32->zfile_func64.zread_file = p_filefunc32->
```

```
    zread_file;
352   p_filefunc64_32->zfile_func64.zwrite_file = p_filefunc32->
    zwrite_file;
353   p_filefunc64_32->zfile_func64.ztell64_file = NULL;
354   p_filefunc64_32->zfile_func64.zseek64_file = NULL;
355   p_filefunc64_32->zfile_func64.zclose_file = p_filefunc32->
    zclose_file;
356   p_filefunc64_32->zfile_func64.zerror_file = p_filefunc32->
    zerror_file;
357   p_filefunc64_32->zfile_func64.opaque = p_filefunc32->
    opaque;
358   p_filefunc64_32->zfile_func64.zfakeclose_file = NULL;
359   p_filefunc64_32->zseek32_file = p_filefunc32->zseek_file;
360   p_filefunc64_32->ztell32_file = p_filefunc32->ztell_file;
361 }
```

Here is the caller graph for this function:



13.86.2.7 int qiodevice64_seek_file_func (voidpf , voidpf *stream*, ZPOS64_T *offset*, int *origin*)

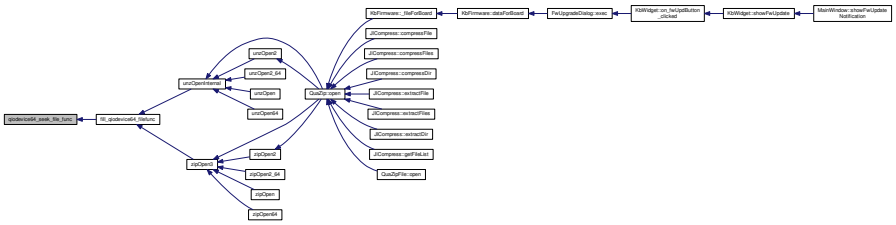
Definition at line 244 of file qioapi.cpp.

References ZLIB_FILEFUNC_SEEK_CUR, ZLIB_FILEFUNC_SEEK_END, and ZLIB_FILEFUNC_SEEK_SET.

Referenced by fill_qiodevice64_filefunc().

```
249 {
250     QIODevice *iodevice = reinterpret_cast<QIODevice*>(stream);
251     if (iodevice->isSequential()) {
252         if (origin == ZLIB_FILEFUNC_SEEK_END
253                 && offset == 0) {
254             // sequential devices are always at end (needed in mdAppend)
255             return 0;
256         } else {
257             qWarning("qiodevice_seek_file_func() called for sequential device");
258             return -1;
259         }
260     }
261     qint64 qiodevice_seek_result=0;
262     int ret;
263     switch (origin)
264     {
265     case ZLIB_FILEFUNC_SEEK_CUR :
266         qiodevice_seek_result = ((QIODevice*)stream)->pos() + offset;
267         break;
268     case ZLIB_FILEFUNC_SEEK_END :
269         qiodevice_seek_result = ((QIODevice*)stream)->size() - offset;
270         break;
271     case ZLIB_FILEFUNC_SEEK_SET :
272         qiodevice_seek_result = offset;
273         break;
274     default:
275         return -1;
276     }
277     ret = !iodevice->seek(qiodevice_seek_result);
278     return ret;
279 }
```

Here is the caller graph for this function:



13.86.2.8 ZPOS64_T qiodevice64_tell_file_func (voidpf *opaque*, voidpf *stream*)

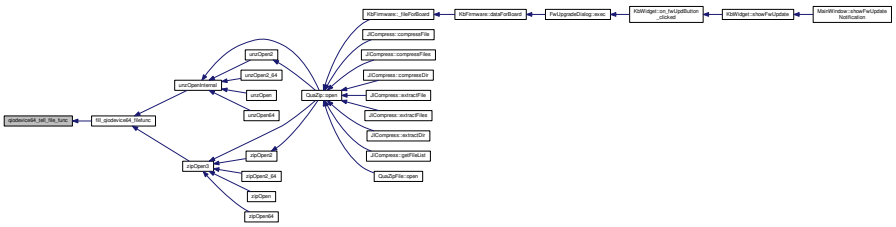
Definition at line 192 of file qioapi.cpp.

References QIODevice_descriptor::pos.

Referenced by fill_qiodevice64_filefunc().

```
195 {
196     QIODevice_descriptor *d = reinterpret_cast<
      QIODevice_descriptor*>(opaque);
197     QIODevice *iodevice = reinterpret_cast<QIODevice*>(stream);
198     qint64 ret;
199     if (iodevice->isSequential()) {
200         ret = d->pos;
201     } else {
202         ret = iodevice->pos();
203     }
204     return static_cast<ZPOS64_T>(ret);
205 }
```

Here is the caller graph for this function:



13.86.2.9 int qiodevice_close_file_func (voidpf *opaque*, voidpf *stream*)

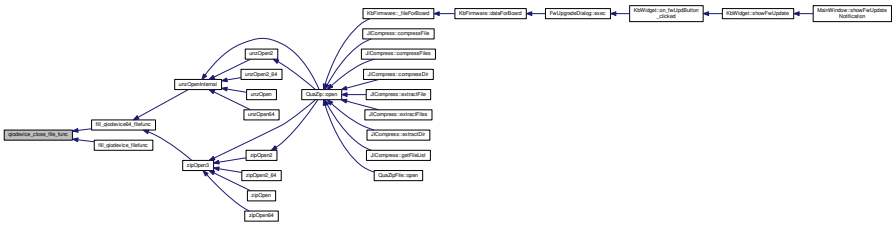
Definition at line 281 of file qioapi.cpp.

Referenced by fill_qiodevice64_filefunc(), and fill_qiodevice_filefunc().

```
284 {
285     QIODevice_descriptor *d = reinterpret_cast<
      QIODevice_descriptor*>(opaque);
286     delete d;
287     QIODevice *device = reinterpret_cast<QIODevice*>(stream);
288 #ifdef QUAZIP_QSAVEFILE_BUG_WORKAROUND
289     // QSaveFile terribly breaks the is-a idiom:
290     // it IS a QIODevice, but it is NOT compatible with it: close() is private
291     QSaveFile *file = qobject_cast<QSaveFile*>(device);
292     if (file != NULL) {
293         // We have to call the ugly commit() instead:
294         return file->commit() ? 0 : -1;
295     }
```

```
296 #endif
297     device->close();
298     return 0;
299 }
```

Here is the caller graph for this function:



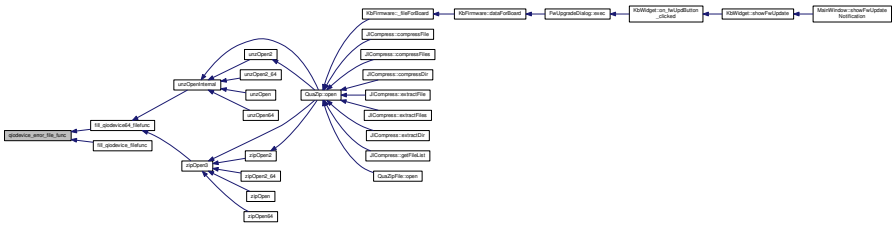
13.86.2.10 int qiodevice_error_file_func (voidpf , voidpf)

Definition at line 310 of file qioapi.cpp.

Referenced by fill_qiodevice64_filefunc(), and fill_qiodevice_filefunc().

```
313 {
314     // can't check for error due to the QIODevice API limitation
315     return 0;
316 }
```

Here is the caller graph for this function:



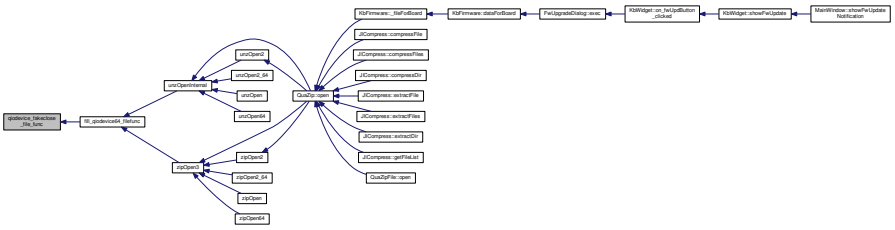
13.86.2.11 int qiodevice_fakeclose_file_func (voidpf *opaque*, voidpf)

Definition at line 301 of file qioapi.cpp.

Referenced by fill_qiodevice64_filefunc().

```
304 {
305     QIODevice_descriptor *d = reinterpret_cast<
    QIODevice_descriptor*>(opaque);
306     delete d;
307     return 0;
308 }
```

Here is the caller graph for this function:



13.86.2.12 voidpf qiodevice_open_file_func (voidpf *opaque*, voidpf *file*, int *mode*)

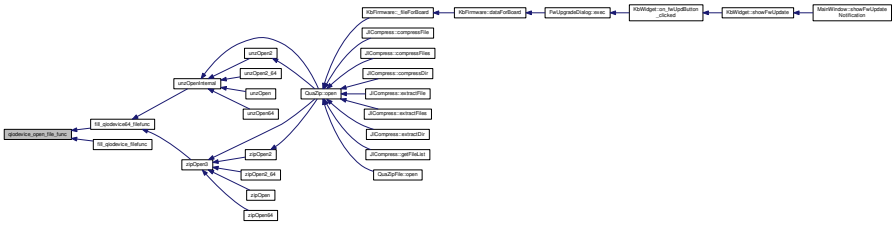
Definition at line 86 of file qioapi.cpp.

References QIODevice_descriptor::pos, ZLIB_FILEFUNC_MODE_CREATE, ZLIB_FILEFUNC_MODE_EXISTING, ZLIB_FILEFUNC_MODE_READ, and ZLIB_FILEFUNC_MODE_READWRITEFILTER.

Referenced by fill_qiodevice64_filefunc(), and fill_qiodevice_filefunc().

```
90 {
91     QIODevice_descriptor *d = reinterpret_cast<
      QIODevice_descriptor*>(opaque);
92     QIODevice *iodevice = reinterpret_cast<QIODevice*>(file);
93     QIODevice::OpenMode desiredMode;
94     if ((mode & ZLIB_FILEFUNC_MODE_READWRITEFILTER)==
      ZLIB_FILEFUNC_MODE_READ)
95         desiredMode = QIODevice::ReadOnly;
96     else if (mode & ZLIB_FILEFUNC_MODE_EXISTING)
97         desiredMode = QIODevice::ReadWrite;
98     else if (mode & ZLIB_FILEFUNC_MODE_CREATE)
99         desiredMode = QIODevice::WriteOnly;
100     if (iodevice->isOpen()) {
101         if ((iodevice->openMode() & desiredMode) == desiredMode) {
102             if (desiredMode != QIODevice::WriteOnly
103                     && iodevice->isSequential()) {
104                 // We can use sequential devices only for writing.
105                 delete d;
106                 return NULL;
107             } else {
108                 if ((desiredMode & QIODevice::WriteOnly) != 0) {
109                     // open for writing, need to seek existing device
110                     if (!iodevice->isSequential()) {
111                         iodevice->seek(0);
112                     } else {
113                         d->pos = iodevice->pos();
114                     }
115                 }
116             }
117             return iodevice;
118         } else {
119             delete d;
120             return NULL;
121         }
122     }
123     iodevice->open(desiredMode);
124     if (iodevice->isOpen()) {
125         if (desiredMode != QIODevice::WriteOnly && iodevice->isSequential()) {
126             // We can use sequential devices only for writing.
127             iodevice->close();
128             delete d;
129             return NULL;
130         } else {
131             return iodevice;
132         }
133     } else {
134         delete d;
135         return NULL;
136     }
137 }
```

Here is the caller graph for this function:



13.86.2.13 uLong qiodevice_read_file_func (voidpf *opaque,* voidpf *stream,* void * *buf,* uLong *size*)

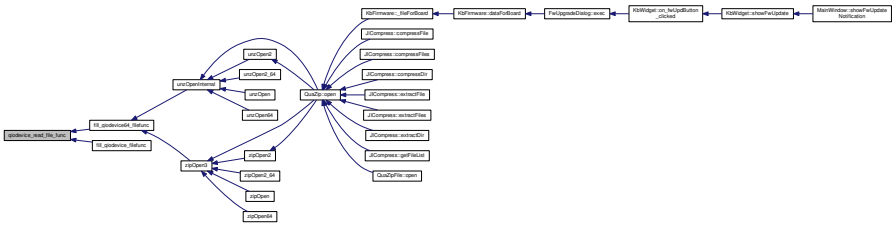
Definition at line 140 of file qioapi.cpp.

References QIODevice_descriptor::pos.

Referenced by fill_qiodevice64_filefunc(), and fill_qiodevice_filefunc().

```
145 {
146     QIODevice_descriptor *d = reinterpret_cast<
    QIODevice_descriptor*>(opaque);
147     QIODevice *iodevice = reinterpret_cast<QIODevice*>(stream);
148     qint64 ret64 = iodevice->read((char*)buf,size);
149     uLong ret;
150     ret = (uLong) ret64;
151     if (ret64 != -1) {
152         d->pos += ret64;
153     }
154     return ret;
155 }
```

Here is the caller graph for this function:



13.86.2.14 int qiodevice_seek_file_func (voidpf , voidpf *stream,* uLong *offset,* int *origin*)

Definition at line 207 of file qioapi.cpp.

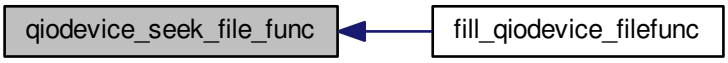
References ZLIB_FILEFUNC_SEEK_CUR, ZLIB_FILEFUNC_SEEK_END, and ZLIB_FILEFUNC_SEEK_SET.

Referenced by fill_qiodevice_filefunc().

```
212 {
213     QIODevice *iodevice = reinterpret_cast<QIODevice*>(stream);
214     if (iodevice->isSequential()) {
215         if (origin == ZLIB_FILEFUNC_SEEK_END
216                 && offset == 0) {
217             // sequential devices are always at end (needed in mdAppend)
218             return 0;
219         } else {
220             qWarning("qiodevice_seek_file_func() called for sequential device");
221             return -1;
```

```
222         }
223     }
224     uLong qiodevice_seek_result=0;
225     int ret;
226     switch (origin)
227     {
228     case ZLIB_FILEFUNC_SEEK_CUR :
229         qiodevice_seek_result = ((QIODevice*)stream)->pos() + offset;
230         break;
231     case ZLIB_FILEFUNC_SEEK_END :
232         qiodevice_seek_result = ((QIODevice*)stream)->size() - offset;
233         break;
234     case ZLIB_FILEFUNC_SEEK_SET :
235         qiodevice_seek_result = offset;
236         break;
237     default:
238         return -1;
239     }
240     ret = !iodevice->seek(qiodevice_seek_result);
241     return ret;
242 }
```

Here is the caller graph for this function:



13.86.2.15 uLong qiodevice_tell_file_func (voidpf *opaque*, voidpf *stream*)

Definition at line 175 of file qioapi.cpp.

References QIODevice_descriptor::pos.

Referenced by fill_qiodevice_filefunc().

```
178 {
179     QIODevice_descriptor *d = reinterpret_cast<
      QIODevice_descriptor*>(opaque);
180     QIODevice *iodevice = reinterpret_cast<QIODevice*>(stream);
181     uLong ret;
182     qint64 ret64;
183     if (iodevice->isSequential()) {
184         ret64 = d->pos;
185     } else {
186         ret64 = iodevice->pos();
187     }
188     ret = static_cast<uLong>(ret64);
189     return ret;
190 }
```

Here is the caller graph for this function:

13.86.2.16 uLong qiodevice_write_file_func (voidpf *opaque*, voidpf *stream*, const void * *buf*, uLong *size*)

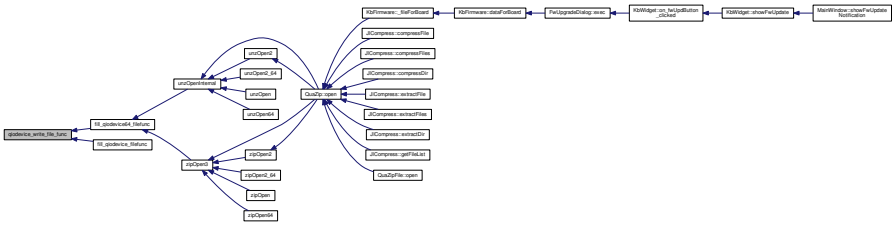
Definition at line 158 of file qioapi.cpp.

References QIODevice_descriptor::pos.

Referenced by fill_qiodevice64_filefunc(), and fill_qiodevice_filefunc().

```
163 {
164     QIODevice_descriptor *d = reinterpret_cast<
      QIODevice_descriptor*>(opaque);
165     QIODevice *iodevice = reinterpret_cast<QIODevice*>(stream);
166     uLong ret;
167     qint64 ret64 = iodevice->write((char*)buf,size);
168     if (ret64 != -1) {
169         d->pos += ret64;
170     }
171     ret = (uLong) ret64;
172     return ret;
173 }
```

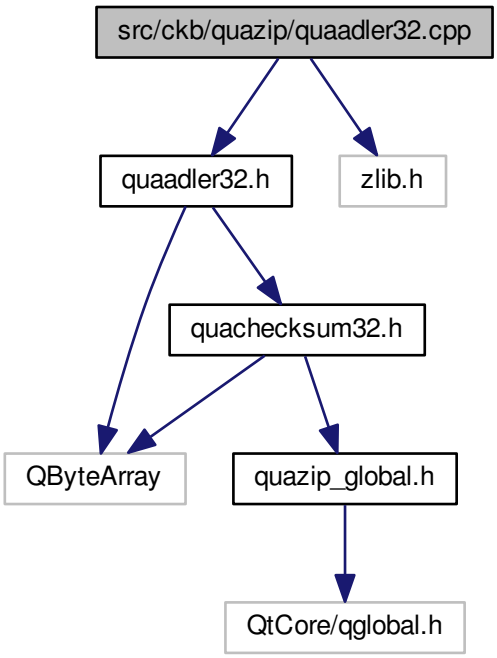
Here is the caller graph for this function:



13.87 src/ckb/quazip/quaadler32.cpp File Reference

```
#include "quaadler32.h"
#include "zlib.h"
```

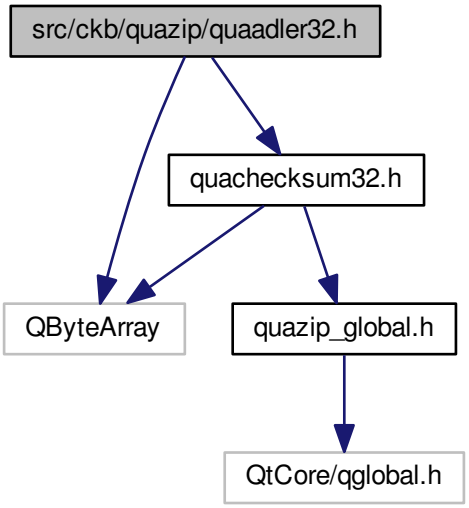
Include dependency graph for quaadler32.cpp:



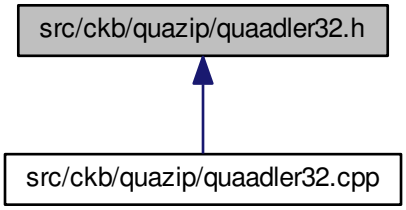
13.88 src/ckb/quazip/quaadler32.h File Reference

```
#include <QByteArray>
#include "quachecksum32.h"
```

Include dependency graph for quaadler32.h:



This graph shows which files directly or indirectly include this file:



Data Structures

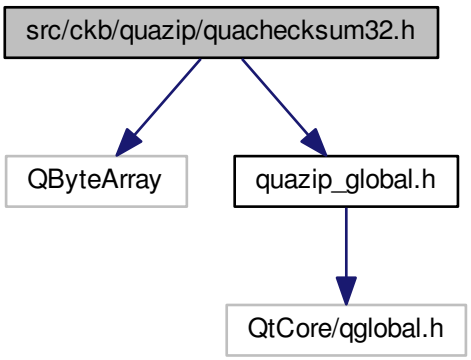
- class QuaAdler32

  Adler32 checksum.

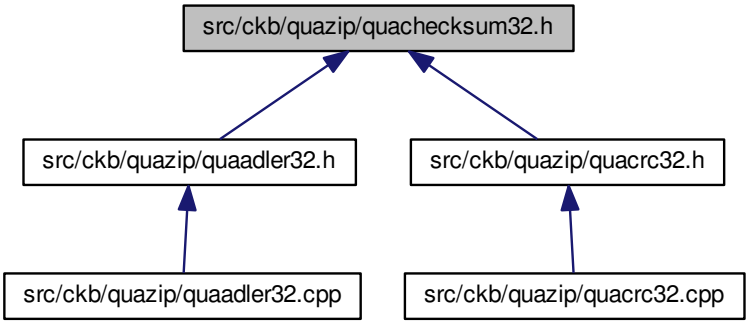
13.89 src/ckb/quazip/quachecksum32.h File Reference

```
#include <QByteArray>
#include "quazip_global.h"
```

Include dependency graph for quachecksum32.h:



This graph shows which files directly or indirectly include this file:



Data Structures

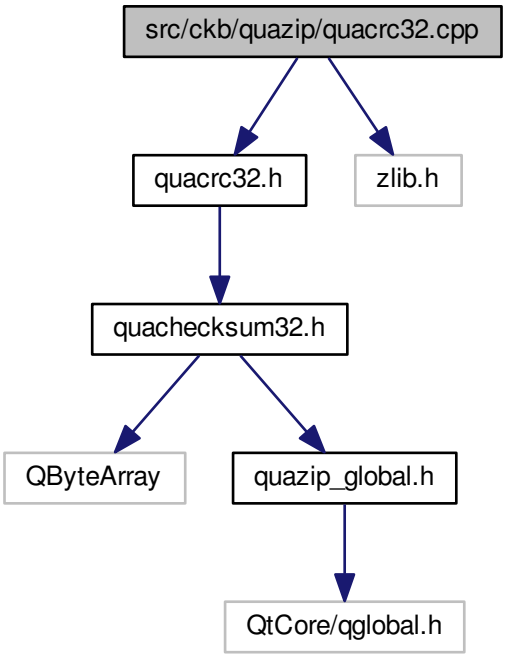
- class QuaChecksum32

  Checksum interface.

13.90 src/ckb/quazip/quacrc32.cpp File Reference

```
#include "quacrc32.h"
#include "zlib.h"
```

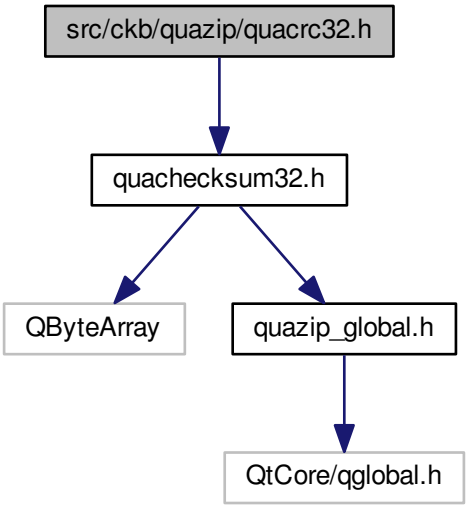
Include dependency graph for quacrc32.cpp:



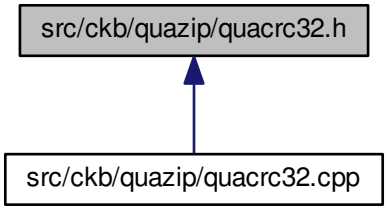
13.91 src/ckb/quazip/quacrc32.h File Reference

```
#include "quachecksum32.h"
```

Include dependency graph for quacrc32.h:



This graph shows which files directly or indirectly include this file:



Data Structures

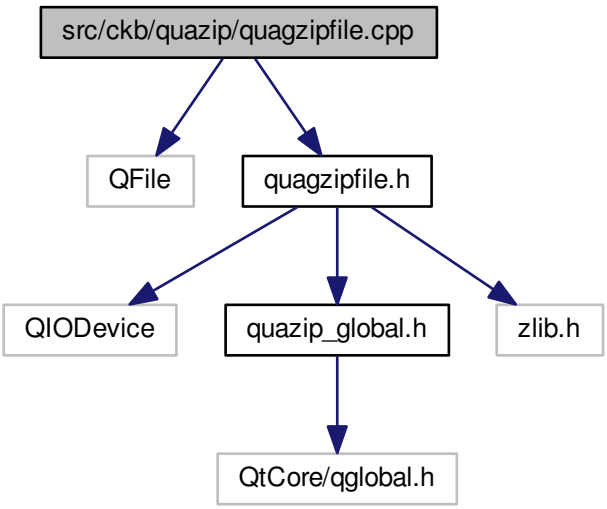
- class QuaCrc32

  CRC32 checksum.

13.92 src/ckb/quazip/quagzipfile.cpp File Reference

```
#include <QFile>
#include "quagzipfile.h"
```

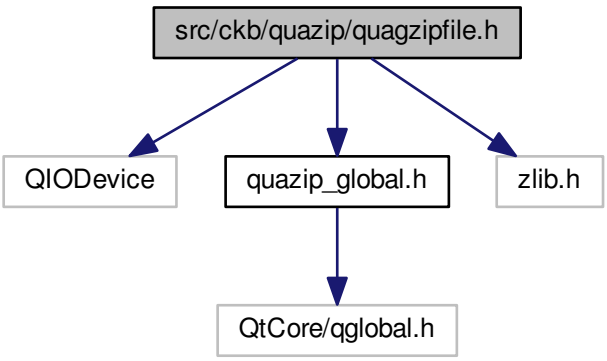
Include dependency graph for quagzipfile.cpp:



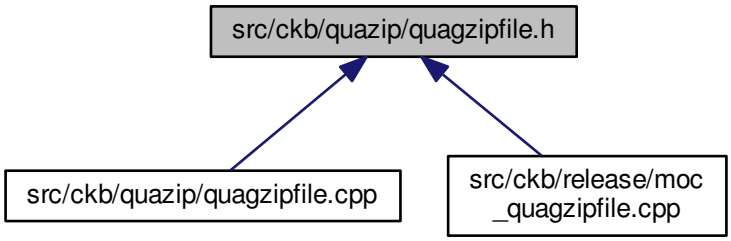
13.93 src/ckb/quazip/quagzipfile.h File Reference

```
#include <QIODevice>
#include "quazip_global.h"
#include <zlib.h>
```

Include dependency graph for quagzipfile.h:



This graph shows which files directly or indirectly include this file:

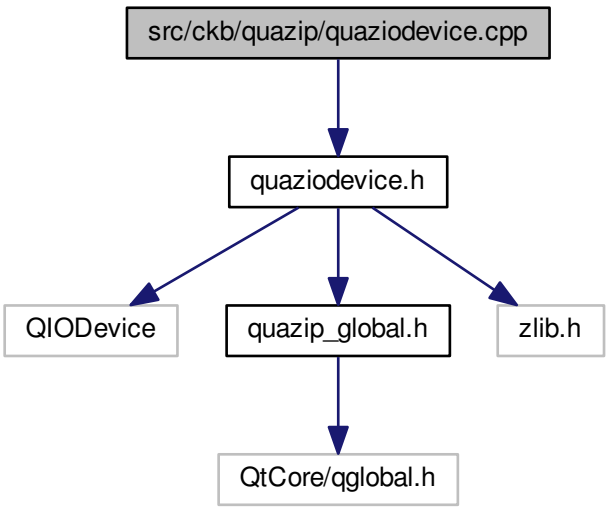


Data Structures

- class QuaGzipFile
  GZIP file.

13.94 src/ckb/quazip/quaziodevice.cpp File Reference

```
#include "quaziodevice.h"
```
Include dependency graph for quaziodevice.cpp:



Macros

- #define QUAZIO_INBUFSIZE 4096

- #define QUAZIO_OUTBUFSIZE 4096

13.94.1 Macro Definition Documentation

13.94.1.1 #define QUAZIO_INBUFSIZE 4096

Definition at line 27 of file quaziodevice.cpp.

Referenced by QuaZIODevice::readData().

13.94.1.2 #define QUAZIO_OUTBUFSIZE 4096

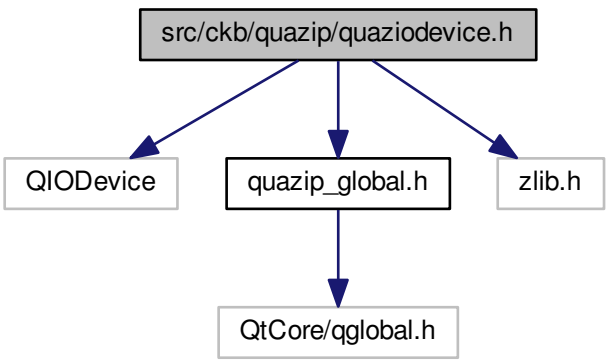
Definition at line 28 of file quaziodevice.cpp.

Referenced by QuaZIODevice::flush(), and QuaZIODevice::writeData().

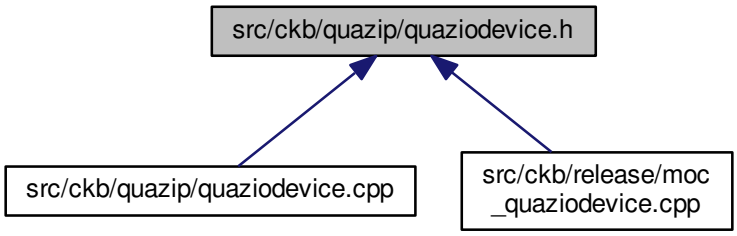
13.95 src/ckb/quazip/quaziodevice.h File Reference

```
#include <QIODevice>
#include "quazip_global.h"
#include <zlib.h>
```

Include dependency graph for quaziodevice.h:



This graph shows which files directly or indirectly include this file:



Data Structures

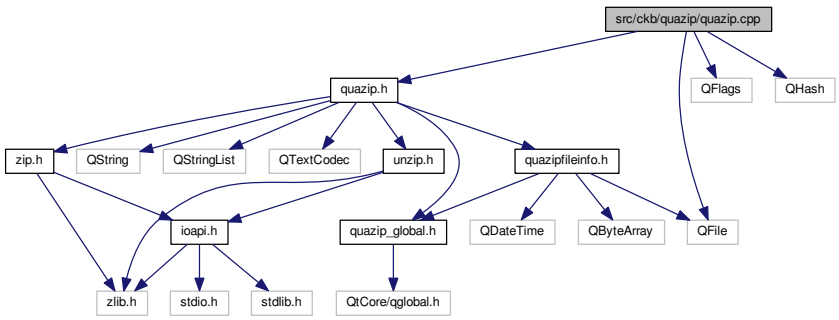
- class QuaZIODevice

  A class to compress/decompress QIODevice.

13.96 src/ckb/quazip/quazip.cpp File Reference

```
#include <QFile>
#include <QFlags>
#include <QHash>
#include "quazip.h"
```

Include dependency graph for quazip.cpp:



Data Structures

- class QuaZipPrivate

  All the internal stuff for the QuaZip class.

Functions

- template<typename TFileInfo >
  TFileInfo QuaZip_getFileInfo (QuaZip *zip, bool *ok)

- template<>
  QuaZipFileInfo QuaZip_getFileInfo (QuaZip *zip, bool *ok)
- template<>
  QuaZipFileInfo64 QuaZip_getFileInfo (QuaZip *zip, bool *ok)
- template<>
  QString QuaZip_getFileInfo (QuaZip *zip, bool *ok)

13.96.1 Function Documentation

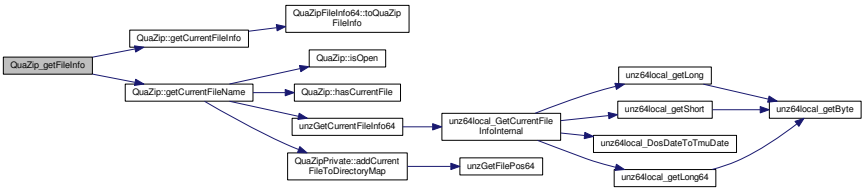
13.96.1.1 template<typename TFileInfo > TFileInfo QuaZip_getFileInfo (QuaZip * *zip*, bool * *ok*)

Definition at line 663 of file quazip.cpp.

References QuaZip::getCurrentFileInfo(), and QuaZip::getCurrentFileName().

```
664 {
665     QuaZipFileInfo info;
666     *ok = zip->getCurrentFileInfo(&info);
667     return info;
668 }
```

Here is the call graph for this function:



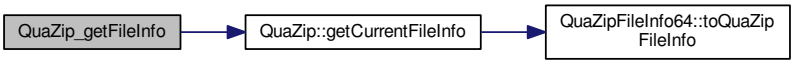
13.96.1.2 template<> QuaZipFileInfo QuaZip_getFileInfo (QuaZip * *zip*, bool * *ok*)

Definition at line 663 of file quazip.cpp.

References QuaZip::getCurrentFileInfo().

```
664 {
665     QuaZipFileInfo info;
666     *ok = zip->getCurrentFileInfo(&info);
667     return info;
668 }
```

Here is the call graph for this function:



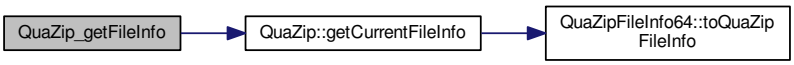
13.96.1.3 template<> QuaZipFileInfo64 QuaZip_getFileInfo (QuaZip * *zip*, bool * *ok*)

Definition at line 671 of file quazip.cpp.

References QuaZip::getCurrentFileInfo().

```
672 {
673     QuaZipFileInfo64 info;
674     *ok = zip->getCurrentFileInfo(&info);
675     return info;
676 }
```

Here is the call graph for this function:



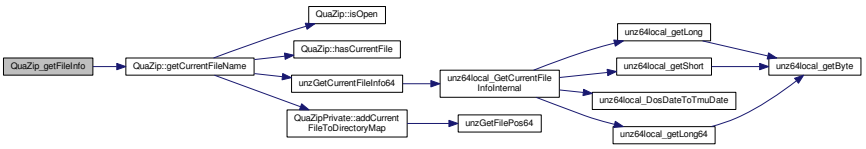
13.96.1.4 template<> QString QuaZip_getFileInfo (QuaZip * *zip*, bool * *ok*)

Definition at line 679 of file quazip.cpp.

References QuaZip::getCurrentFileName().

```
680 {
681     QString name = zip->getCurrentFileName();
682     *ok = !name.isEmpty();
683     return name;
684 }
```

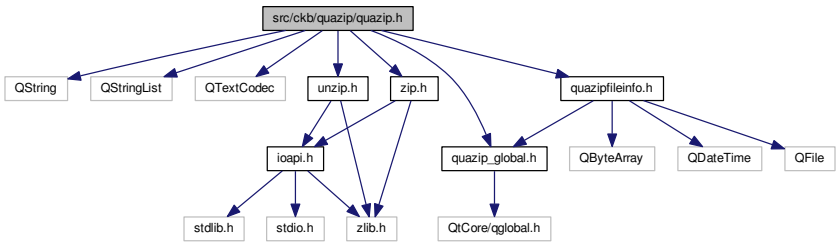
Here is the call graph for this function:



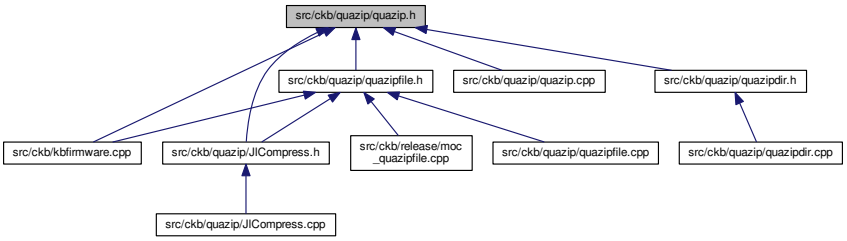
13.97 src/ckb/quazip/quazip.h File Reference

```
#include <QString>
#include <QStringList>
#include <QTextCodec>
#include "zip.h"
#include "unzip.h"
#include "quazip_global.h"
#include "quazipfileinfo.h"
```

Include dependency graph for quazip.h:



This graph shows which files directly or indirectly include this file:



Data Structures

- class QuaZip

  ZIP archive.

Macros

- #define UNZ_OPENERROR -1000

13.97.1 Macro Definition Documentation

13.97.1.1 #define UNZ_OPENERROR -1000

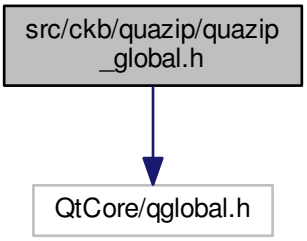
Definition at line 41 of file quazip.h.

Referenced by QuaZip::open().

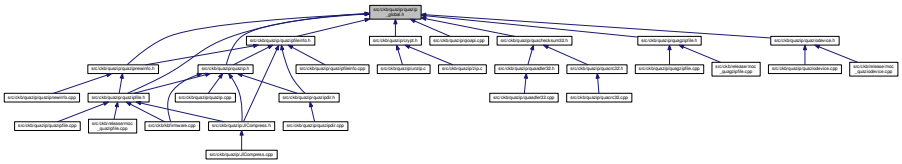
13.98 src/ckb/quazip/quazip_global.h File Reference

```
#include <QtCore/qglobal.h>
```

Include dependency graph for quazip_global.h:



This graph shows which files directly or indirectly include this file:



Macros

- #define QUAZIP_EXPORT Q_DECL_IMPORT

  This is automatically defined when building a static library, but when including QuaZip sources directly into a project, QUAZIP_STATIC should be defined explicitly to avoid possible troubles with unnecessary importing/exporting.
- #define UNUSED
- #define QUAZIP_EXTRA_NTFS_MAGIC 0x000Au
- #define QUAZIP_EXTRA_NTFS_TIME_MAGIC 0x0001u

13.98.1 Macro Definition Documentation

13.98.1.1 #define QUAZIP_EXPORT Q_DECL_IMPORT

When building a DLL with MSVC, QUAZIP_BUILD must be defined. qglobal.h takes care of defining Q_DECL_∗ correctly for msvc/gcc.

Definition at line 46 of file quazip_global.h.

13.98.1.2 #define QUAZIP_EXTRA_NTFS_MAGIC 0x000Au

Definition at line 56 of file quazip_global.h.

Referenced by getNTFSTime(), and setNTFSTime().

13.98.1.3 #define QUAZIP_EXTRA_NTFS_TIME_MAGIC 0x0001u

Definition at line 57 of file quazip_global.h.

Referenced by getNTFSTime(), and setNTFSTime().

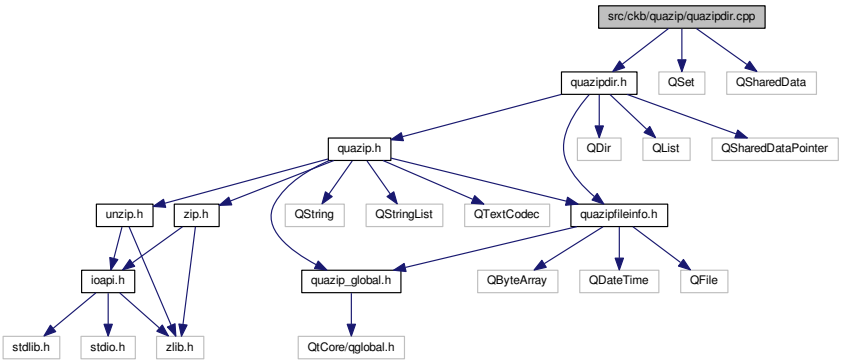
13.98.1.4 #define UNUSED

Definition at line 53 of file quazip_global.h.

13.99 src/ckb/quazip/quazipdir.cpp File Reference

```
#include "quazipdir.h"
#include <QSet>
#include <QSharedData>
```

Include dependency graph for quazipdir.cpp:



Functions

- QuaZipFileInfo64 QuaZipDir_getFileInfo (QuaZip *zip, bool *ok, const QString &relativeName, bool isReal)
- static void QuaZipDir_convertInfoList (const QList< QuaZipFileInfo64 > &from, QList< QuaZipFileInfo64 > &to)
- static void QuaZipDir_convertInfoList (const QList< QuaZipFileInfo64 > &from, QStringList &to)
- static void QuaZipDir_convertInfoList (const QList< QuaZipFileInfo64 > &from, QList< QuaZipFileInfo > &to)

13.99.1 Function Documentation

13.99.1.1 static void QuaZipDir_convertInfoList (const QList< QuaZipFileInfo64 > & *from*, QList< QuaZipFileInfo64 > & *to*) [static]

Definition at line 188 of file quazipdir.cpp.

```
190 {
191     to = from;
192 }
```

13.99.1.2 static void QuaZipDir_convertInfoList (const QList< QuaZipFileInfo64 > & *from*, QStringList & *to*) [static]

Definition at line 194 of file quazipdir.cpp.

```
196 {
197     to.clear();
198     for (QList<QuaZipFileInfo64>::const_iterator i = from.constBegin();
199             i != from.constEnd();
200             ++i) {
201         to.append(i->name);
202     }
203 }
```

13.99.1.3 static void QuaZipDir_convertInfoList (const QList< QuaZipFileInfo64 > & *from*, QList< QuaZipFileInfo > & *to*) [static]

Definition at line 205 of file quazipdir.cpp.

```
207 {
208     to.clear();
209     for (QList<QuaZipFileInfo64>::const_iterator i = from.constBegin();
210             i != from.constEnd();
211             ++i) {
212         QuaZipFileInfo info32;
213         i->toQuaZipFileInfo(info32);
214         to.append(info32);
215     }
216 }
```

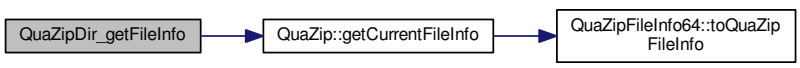
13.99.1.4 QuaZipFileInfo64 QuaZipDir_getFileInfo (QuaZip * *zip*, bool * *ok*, const QString & *relativeName*, bool *isReal*)

Definition at line 165 of file quazipdir.cpp.

References QuaZipFileInfo64::compressedSize, QuaZipFileInfo64::crc, QuaZipFileInfo64::diskNumberStart, QuaZipFileInfo64::externalAttr, QuaZipFileInfo64::flags, QuaZip::getCurrentFileInfo(), QuaZipFileInfo64::internalAttr, QuaZipFileInfo64::method, QuaZipFileInfo64::name, QuaZipFileInfo64::uncompressedSize, QuaZipFileInfo64::versionCreated, and QuaZipFileInfo64::versionNeeded.

```
168 {
169     QuaZipFileInfo64 info;
170     if (isReal) {
171         *ok = zip->getCurrentFileInfo(&info);
172     } else {
173         *ok = true;
174         info.compressedSize = 0;
175         info.crc = 0;
176         info.diskNumberStart = 0;
177         info.externalAttr = 0;
178         info.flags = 0;
179         info.internalAttr = 0;
180         info.method = 0;
181         info.uncompressedSize = 0;
182         info.versionCreated = info.versionNeeded = 0;
183     }
184     info.name = relativeName;
185     return info;
186 }
```

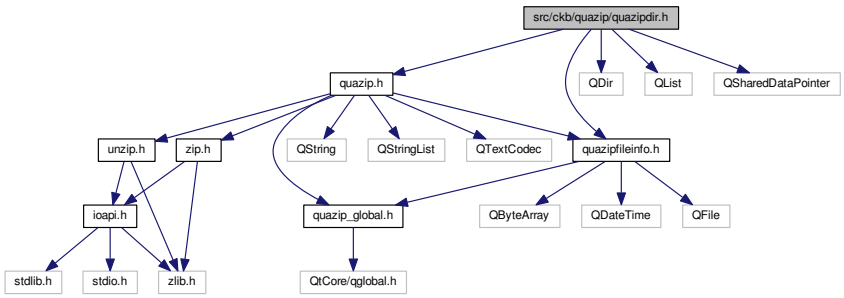
Here is the call graph for this function:



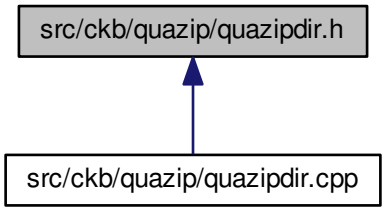
13.100 src/ckb/quazip/quazipdir.h File Reference

```
#include "quazip.h"
#include "quazipfileinfo.h"
#include <QDir>
#include <QList>
#include <QSharedDataPointer>
```

Include dependency graph for quazipdir.h:



This graph shows which files directly or indirectly include this file:



Data Structures

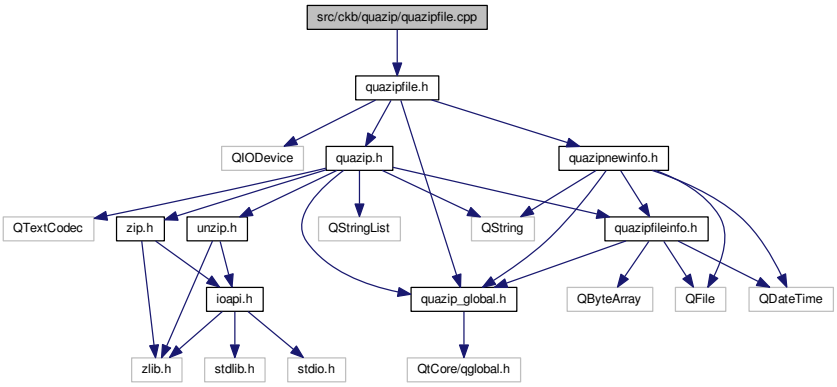
- class QuaZipDir

  Provides ZIP archive navigation.

13.101 src/ckb/quazip/quazipfile.cpp File Reference

```
#include "quazipfile.h"
```

Include dependency graph for quazipfile.cpp:



Data Structures

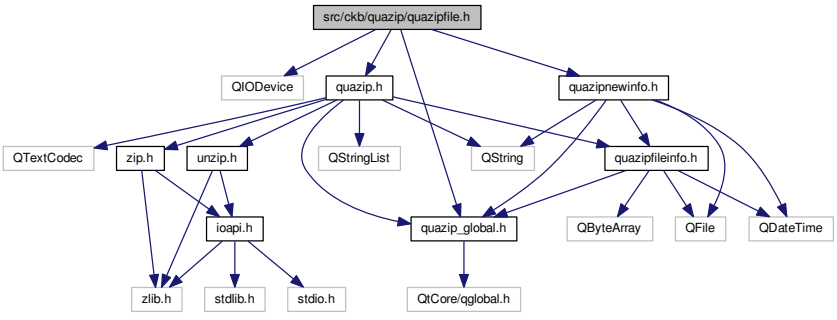
- class QuaZipFilePrivate

  The implementation class for QuaZip.

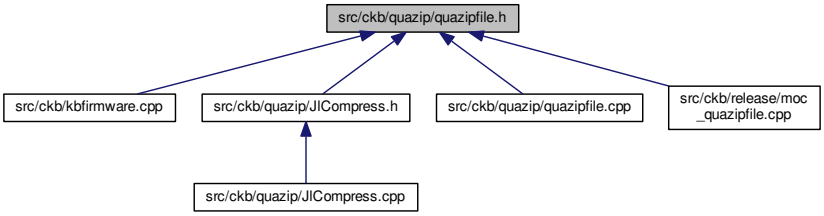
13.102 src/ckb/quazip/quazipfile.h File Reference

```
#include <QIODevice>
#include "quazip_global.h"
#include "quazip.h"
#include "quazipnewinfo.h"
```

Include dependency graph for quazipfile.h:



This graph shows which files directly or indirectly include this file:



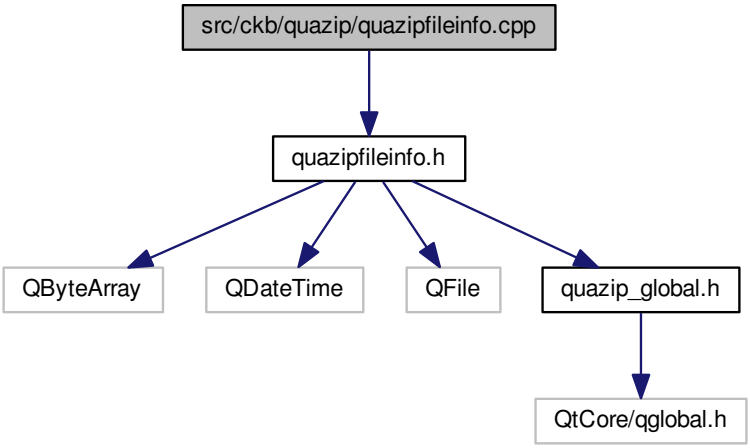
Data Structures

- class QuaZipFile

  A file inside ZIP archive.

13.103 src/ckb/quazip/quazipfileinfo.cpp File Reference

`#include "quazipfileinfo.h"`
Include dependency graph for quazipfileinfo.cpp:



Functions

- static QFile::Permissions permissionsFromExternalAttr (quint32 externalAttr)
- static QDateTime getNTFSTime (const QByteArray &extra, int position, int *fineTicks)

13.103.1 Function Documentation

13.103.1.1 static QDateTime getNTFSTime (const QByteArray & *extra*, int *position*, int * *fineTicks*) [static]

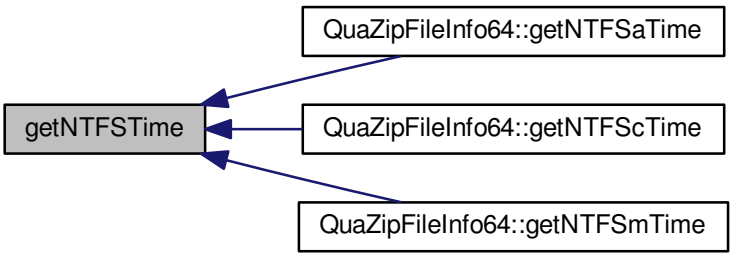
Definition at line 92 of file quazipfileinfo.cpp.

References QUAZIP_EXTRA_NTFS_MAGIC, and QUAZIP_EXTRA_NTFS_TIME_MAGIC.

Referenced by QuaZipFileInfo64::getNTFSaTime(), QuaZipFileInfo64::getNTFScTime(), and QuaZipFileInfo64::getNTFSmTime().

```
94 {
95     QDateTime dateTime;
96     for (int i = 0; i <= extra.size() - 4; ) {
97         unsigned type = static_cast<unsigned>(static_cast<unsigned char>(
98                                                   extra.at(i)))
99                 | (static_cast<unsigned>(static_cast<unsigned char>(
100                                                    extra.at(i + 1))) << 8);
101         i += 2;
102         unsigned length = static_cast<unsigned>(static_cast<unsigned char>(
103                                                    extra.at(i)))
104                 | (static_cast<unsigned>(static_cast<unsigned char>(
105                                                    extra.at(i + 1))) << 8);
106         i += 2;
107         if (type == QUAZIP_EXTRA_NTFS_MAGIC && length >= 32) {
108             i += 4; // reserved
109             while (i <= extra.size() - 4) {
110                 unsigned tag = static_cast<unsigned>(
111                             static_cast<unsigned char>(extra.at(i)))
112                         | (static_cast<unsigned>(
113                                 static_cast<unsigned char>(extra.at(i + 1)))
114                             << 8);
115                 i += 2;
116                 int tagsize = static_cast<unsigned>(
117                             static_cast<unsigned char>(extra.at(i)))
118                         | (static_cast<unsigned>(
119                                 static_cast<unsigned char>(extra.at(i + 1)))
120                             << 8);
121                 i += 2;
122                 if (tag == QUAZIP_EXTRA_NTFS_TIME_MAGIC
123                         && tagsize >= position + 8) {
124                     i += position;
125                     quint64 mtime = static_cast<quint64>(
126                                 static_cast<unsigned char>(extra.at(i)))
127                         | (static_cast<quint64>(static_cast<unsigned char>(
128                                                    extra.at(i + 1))) << 8)
129                         | (static_cast<quint64>(static_cast<unsigned char>(
130                                                    extra.at(i + 2))) << 16)
131                         | (static_cast<quint64>(static_cast<unsigned char>(
132                                                    extra.at(i + 3))) << 24)
133                         | (static_cast<quint64>(static_cast<unsigned char>(
134                                                    extra.at(i + 4))) << 32)
135                         | (static_cast<quint64>(static_cast<unsigned char>(
136                                                    extra.at(i + 5))) << 40)
137                         | (static_cast<quint64>(static_cast<unsigned char>(
138                                                    extra.at(i + 6))) << 48)
139                         | (static_cast<quint64>(static_cast<unsigned char>(
140                                                    extra.at(i + 7))) << 56);
141                     // the NTFS time is measured from 1601 for whatever reason
142                     QDateTime base(QDate(1601, 1, 1), QTime(0, 0), Qt::UTC);
143                     dateTime = base.addMSecs(mtime / 10000);
144                     if (fineTicks != NULL) {
145                         *fineTicks = static_cast<int>(mtime % 10000);
146                     }
147                     i += tagsize - position;
148                 } else {
149                     i += tagsize;
150                 }
151 
152             }
153         } else {
154             i += length;
155         }
156     }
157     if (fineTicks != NULL && dateTime.isNull()) {
158         *fineTicks = 0;
159     }
160     return dateTime;
161 }
```

Here is the caller graph for this function:



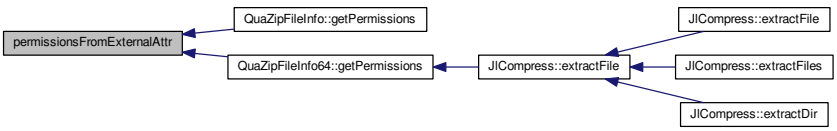
13.103.1.2 static QFile::Permissions permissionsFromExternalAttr (quint32 *externalAttr*) [static]

Definition at line 27 of file quazipfileinfo.cpp.

Referenced by QuaZipFileInfo::getPermissions(), and QuaZipFileInfo64::getPermissions().

```
27                                                                      {
28     quint32 uPerm = (externalAttr & 0xFFFF0000u) >> 16;
29     QFile::Permissions perm = 0;
30     if ((uPerm & 0400) != 0)
31         perm |= QFile::ReadOwner;
32     if ((uPerm & 0200) != 0)
33         perm |= QFile::WriteOwner;
34     if ((uPerm & 0100) != 0)
35         perm |= QFile::ExeOwner;
36     if ((uPerm & 0040) != 0)
37         perm |= QFile::ReadGroup;
38     if ((uPerm & 0020) != 0)
39         perm |= QFile::WriteGroup;
40     if ((uPerm & 0010) != 0)
41         perm |= QFile::ExeGroup;
42     if ((uPerm & 0004) != 0)
43         perm |= QFile::ReadOther;
44     if ((uPerm & 0002) != 0)
45         perm |= QFile::WriteOther;
46     if ((uPerm & 0001) != 0)
47         perm |= QFile::ExeOther;
48     return perm;
49
50 }
```

Here is the caller graph for this function:

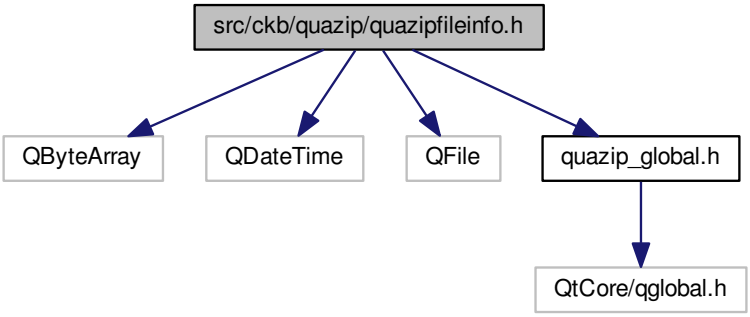


13.104 src/ckb/quazip/quazipfileinfo.h File Reference

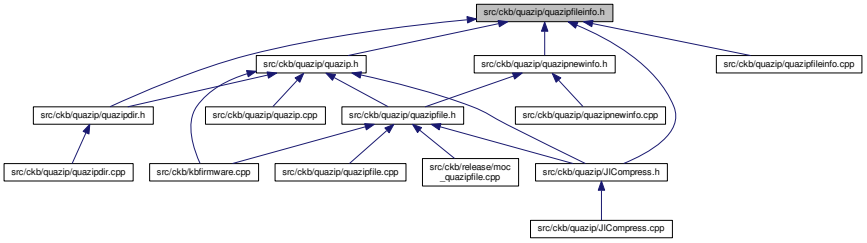
```
#include <QByteArray>
```

```
#include <QDateTime>
#include <QFile>
#include "quazip_global.h"
```

Include dependency graph for quazipfileinfo.h:



This graph shows which files directly or indirectly include this file:



Data Structures

- struct QuaZipFileInfo

  Information about a file inside archive.

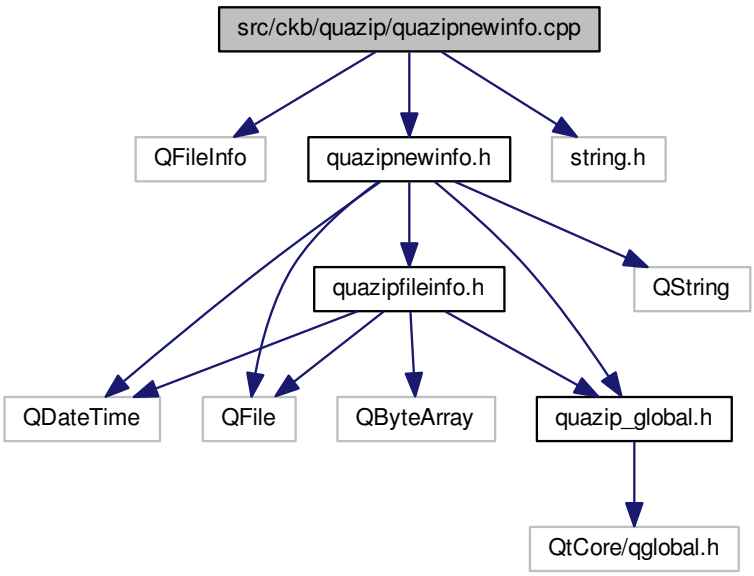
- struct QuaZipFileInfo64

  Information about a file inside archive (with zip64 support).

13.105 src/ckb/quazip/quazipnewinfo.cpp File Reference

```
#include <QFileInfo>
#include "quazipnewinfo.h"
#include <string.h>
```

Include dependency graph for quazipnewinfo.cpp:



Functions

- static void QuaZipNewInfo_setPermissions (QuaZipNewInfo *info, QFile::Permissions perm, bool isDir)
- template<typename FileInfo >
  void QuaZipNewInfo_init (QuaZipNewInfo &self, const FileInfo &existing)
- static void setNTFSTime (QByteArray &extra, const QDateTime &time, int position, int fineTicks)

13.105.1 Function Documentation

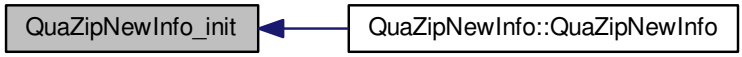
13.105.1.1 template<typename FileInfo > void QuaZipNewInfo_init (QuaZipNewInfo & *self*, const FileInfo & *existing*)

Definition at line 57 of file quazipnewinfo.cpp.

Referenced by QuaZipNewInfo::QuaZipNewInfo().

```
58 {
59     self.name = existing.name;
60     self.dateTime = existing.dateTime;
61     self.internalAttr = existing.internalAttr;
62     self.externalAttr = existing.externalAttr;
63     self.comment = existing.comment;
64     self.extraLocal = existing.extra;
65     self.extraGlobal = existing.extra;
66     self.uncompressedSize = existing.uncompressedSize;
67 }
```

Here is the caller graph for this function:



13.105.1.2 static void QuaZipNewInfo_setPermissions (QuaZipNewInfo * *info*, QFile::Permissions *perm*, bool *isDir*) [static]

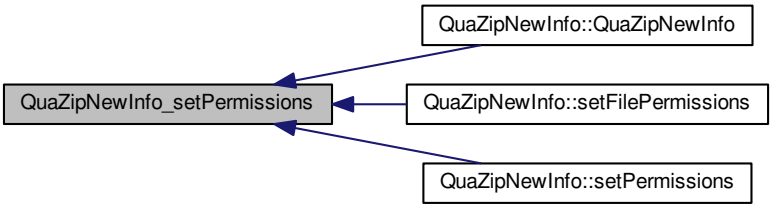
Definition at line 31 of file quazipnewinfo.cpp.

References QuaZipNewInfo::externalAttr.

Referenced by QuaZipNewInfo::QuaZipNewInfo(), QuaZipNewInfo::setFilePermissions(), and QuaZipNewInfo::setPermissions().

```
33 {
34     quint32 uPerm = isDir ? 0040000 : 0100000;
35     if ((perm & QFile::ReadOwner) != 0)
36         uPerm |= 0400;
37     if ((perm & QFile::WriteOwner) != 0)
38         uPerm |= 0200;
39     if ((perm & QFile::ExeOwner) != 0)
40         uPerm |= 0100;
41     if ((perm & QFile::ReadGroup) != 0)
42         uPerm |= 0040;
43     if ((perm & QFile::WriteGroup) != 0)
44         uPerm |= 0020;
45     if ((perm & QFile::ExeGroup) != 0)
46         uPerm |= 0010;
47     if ((perm & QFile::ReadOther) != 0)
48         uPerm |= 0004;
49     if ((perm & QFile::WriteOther) != 0)
50         uPerm |= 0002;
51     if ((perm & QFile::ExeOther) != 0)
52         uPerm |= 0001;
53     info->externalAttr = (info->externalAttr & ~0xFFFF0000u) | (uPerm << 16);
54 }
```

Here is the caller graph for this function:



13.105.1.3 static void setNTFSTime (QByteArray & *extra*, const QDateTime & *time*, int *position*, int *fineTicks*) [static]

Definition at line 131 of file quazipnewinfo.cpp.

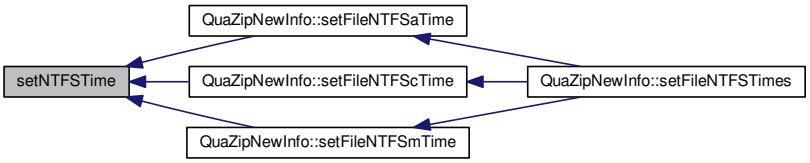
References QUAZIP_EXTRA_NTFS_MAGIC, and QUAZIP_EXTRA_NTFS_TIME_MAGIC.

Referenced by QuaZipNewInfo::setFileNTFSaTime(), QuaZipNewInfo::setFileNTFScTime(), and QuaZipNewInfo::setFileNTFSmTime().

```
132                                     {
133     int ntfsPos = -1, timesPos = -1;
134     unsigned ntfsLength = 0, ntfsTimesLength = 0;
135     for (int i = 0; i <= extra.size() - 4; ) {
136         unsigned type = static_cast<unsigned>(static_cast<unsigned char>(
137                                                 extra.at(i)))
138                 | (static_cast<unsigned>(static_cast<unsigned char>(
139                                                 extra.at(i + 1))) << 8);
140         i += 2;
141         unsigned length = static_cast<unsigned>(static_cast<unsigned char>(
142                                                 extra.at(i)))
143                 | (static_cast<unsigned>(static_cast<unsigned char>(
144                                                 extra.at(i + 1))) << 8);
145         i += 2;
146         if (type == QUAZIP_EXTRA_NTFS_MAGIC) {
147             ntfsPos = i - 4; // the beginning of the NTFS record
148             ntfsLength = length;
149             if (length <= 4) {
150                 break; // no times in the NTFS record
151             }
152             i += 4; // reserved
153             while (i <= extra.size() - 4) {
154                 unsigned tag = static_cast<unsigned>(
155                             static_cast<unsigned char>(extra.at(i)))
156                         | (static_cast<unsigned>(
157                                 static_cast<unsigned char>(extra.at(i + 1)))
158                             << 8);
159                 i += 2;
160                 unsigned tagsize = static_cast<unsigned>(
161                             static_cast<unsigned char>(extra.at(i)))
162                         | (static_cast<unsigned>(
163                                 static_cast<unsigned char>(extra.at(i + 1)))
164                             << 8);
165                 i += 2;
166                 if (tag == QUAZIP_EXTRA_NTFS_TIME_MAGIC) {
167                     timesPos = i - 4; // the beginning of the NTFS times tag
168                     ntfsTimesLength = tagsize;
169                     break;
170                 } else {
171                     i += tagsize;
172                 }
173             }
174             break; // I ain't going to search for yet another NTFS record!
175         } else {
176             i += length;
177         }
178     }
179     if (ntfsPos == -1) {
180         // No NTFS record, need to create one.
181         ntfsPos = extra.size();
182         ntfsLength = 32;
183         extra.resize(extra.size() + 4 + ntfsLength);
184         // the NTFS record header
185         extra[ntfsPos] = static_cast<char>(QUAZIP_EXTRA_NTFS_MAGIC);
186         extra[ntfsPos + 1] = static_cast<char>(QUAZIP_EXTRA_NTFS_MAGIC >> 8);
187         extra[ntfsPos + 2] = 32; // the 2-byte size in LittleEndian
188         extra[ntfsPos + 3] = 0;
189         // zero the record
190         memset(extra.data() + ntfsPos + 4, 0, 32);
191         timesPos = ntfsPos + 8;
192         // now set the tag data
193         extra[timesPos] = static_cast<char>(QUAZIP_EXTRA_NTFS_TIME_MAGIC);
194         extra[timesPos + 1] = static_cast<char>(QUAZIP_EXTRA_NTFS_TIME_MAGIC
195                                                 >> 8);
196         // the size:
197         extra[timesPos + 2] = 24;
198         extra[timesPos + 3] = 0;
199         ntfsTimesLength = 24;
200     }
201     if (timesPos == -1) {
202         // No time tag in the NTFS record, need to add one.
203         timesPos = ntfsPos + 4 + ntfsLength;
204         extra.resize(extra.size() + 28);
205         // Now we need to move the rest of the field
206         // (possibly zero bytes, but memmove() is OK with that).
207         // 0 ......... ntfsPos .. ntfsPos + 4   ... timesPos
208         // <some data> <header>   <NTFS record>     <need-to-move data>    <end>
209         memmove(extra.data() + timesPos + 28, extra.data() + timesPos,
210                 extra.size() - 28 - timesPos);
211         ntfsLength += 28;
212         // now set the tag data
```

```
213         extra[timesPos] = static_cast<char>(QUAZIP_EXTRA_NTFS_TIME_MAGIC);
214         extra[timesPos + 1] = static_cast<char>(QUAZIP_EXTRA_NTFS_TIME_MAGIC
215                                                 >> 8);
216         // the size:
217         extra[timesPos + 2] = 24;
218         extra[timesPos + 3] = 0;
219         // zero the record
220         memset(extra.data() + timesPos + 4, 0, 24);
221         ntfsTimesLength = 24;
222     }
223     if (ntfsTimesLength < 24) {
224         // Broken times field. OK, this is really unlikely, but just in case...
225         size_t timesEnd = timesPos + 4 + ntfsTimesLength;
226         extra.resize(extra.size() + (24 - ntfsTimesLength));
227         // Move it!
228         // 0 ......... timesPos .... timesPos + 4 .. timesEnd
229         // <some data> <time header> <broken times> <need-to-move data> <end>
230         memmove(extra.data() + timesEnd + (24 - ntfsTimesLength),
231                 extra.data() + timesEnd,
232                 extra.size() - (24 - ntfsTimesLength) - timesEnd);
233         // Now we have to increase the NTFS record and time tag lengths.
234         ntfsLength += (24 - ntfsTimesLength);
235         ntfsTimesLength = 24;
236         extra[ntfsPos + 2] = static_cast<char>(ntfsLength);
237         extra[ntfsPos + 3] = static_cast<char>(ntfsLength >> 8);
238         extra[timesPos + 2] = static_cast<char>(ntfsTimesLength);
239         extra[timesPos + 3] = static_cast<char>(ntfsTimesLength >> 8);
240     }
241     QDateTime base(QDate(1601, 1, 1), QTime(0, 0), Qt::UTC);
242 #if (QT_VERSION >= 0x040700)
243     quint64 ticks = base.msecsTo(time) * 10000 + fineTicks;
244 #else
245     QDateTime utc = time.toUTC();
246     quint64 ticks = (static_cast<qint64>(base.date().daysTo(utc.date()))
247             * Q_INT64_C(86400000)
248             + static_cast<qint64>(base.time().msecsTo(utc.time())))
249         * Q_INT64_C(10000) + fineTicks;
250 #endif
251     extra[timesPos + 4 + position] = static_cast<char>(ticks);
252     extra[timesPos + 5 + position] = static_cast<char>(ticks >> 8);
253     extra[timesPos + 6 + position] = static_cast<char>(ticks >> 16);
254     extra[timesPos + 7 + position] = static_cast<char>(ticks >> 24);
255     extra[timesPos + 8 + position] = static_cast<char>(ticks >> 32);
256     extra[timesPos + 9 + position] = static_cast<char>(ticks >> 40);
257     extra[timesPos + 10 + position] = static_cast<char>(ticks >> 48);
258     extra[timesPos + 11 + position] = static_cast<char>(ticks >> 56);
259 }
```

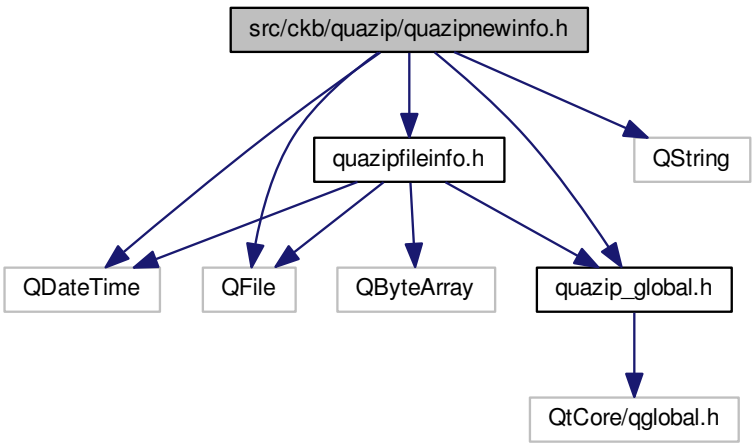
Here is the caller graph for this function:



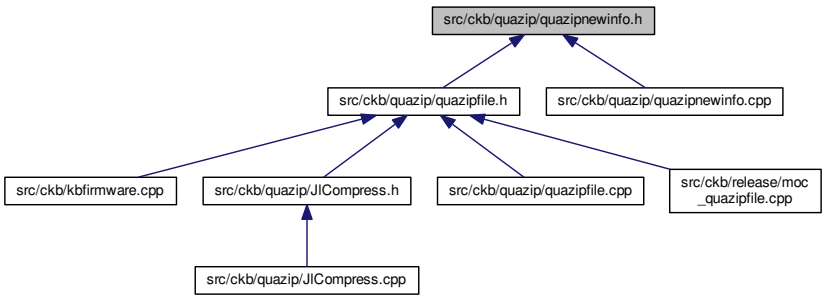
13.106 src/ckb/quazip/quazipnewinfo.h File Reference

```
#include <QDateTime>
#include <QFile>
#include <QString>
#include "quazip_global.h"
#include "quazipfileinfo.h"
```

Include dependency graph for quazipnewinfo.h:



This graph shows which files directly or indirectly include this file:



Data Structures

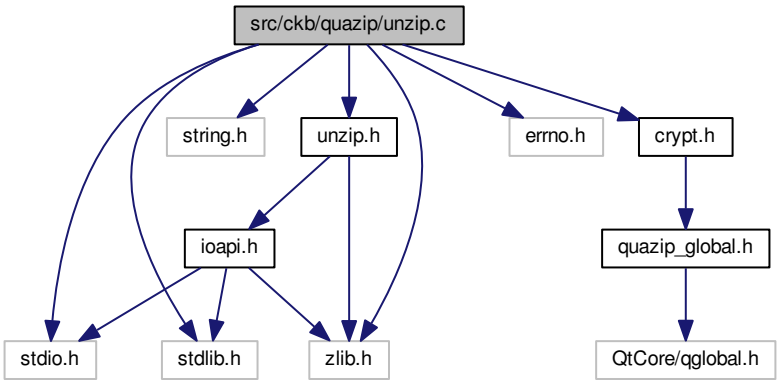
- struct QuaZipNewInfo

  Information about a file to be created.

13.107 src/ckb/quazip/unzip.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include "zlib.h"
#include "unzip.h"
#include <errno.h>
#include "crypt.h"
```

Include dependency graph for unzip.c:



Data Structures

- struct unz_file_info64_internal_s
- struct file_in_zip64_read_info_s
- struct unz64_s

Macros

- #define local static
- #define CASESENSITIVITYDEFAULT_NO
- #define UNZ_BUFSIZE (16384)
- #define UNZ_MAXFILENAMEINZIP (256)
- #define ALLOC(size) (malloc(size))
- #define TRYFREE(p) {if (p) free(p);}
- #define SIZECENTRALDIRITEM (0x2e)
- #define SIZEZIPLOCALHEADER (0x1e)
- #define CASESENSITIVITYDEFAULTVALUE 2
- #define STRCMPCASENOSENTIVEFUNCTION strcmpcasenosensitive_internal
- #define BUFREADCOMMENT (0x400)

Typedefs

- typedef uLongf z_crc_t
- typedef struct unz_file_info64_internal_s unz_file_info64_internal

Functions

- static int unz64local_getByte _Z_OF ((const zlib_filefunc64_32_def *pzlib_filefunc_def, voidpf filestream, int *pi))
- static int unz64local_getByte (const zlib_filefunc64_32_def *pzlib_filefunc_def, voidpf filestream, int *pi)
- static int unz64local_getShort _Z_OF ((const zlib_filefunc64_32_def *pzlib_filefunc_def, voidpf filestream, uLong *pX))

- static int unz64local_getShort (const zlib_filefunc64_32_def *pzlib_filefunc_def, voidpf filestream, uLong *pX)
- static int unz64local_getLong (const zlib_filefunc64_32_def *pzlib_filefunc_def, voidpf filestream, uLong *pX)
- static int unz64local_getLong64 _Z_OF ((const zlib_filefunc64_32_def *pzlib_filefunc_def, voidpf filestream, ZPOS64_T *pX))
- static int unz64local_getLong64 (const zlib_filefunc64_32_def *pzlib_filefunc_def, voidpf filestream, ZPOS64_T *pX)
- static int strcmpcasenosensitive_internal (const char *fileName1, const char *fileName2)
- int ZEXPORT unzStringFileNameCompare (const char *fileName1, const char *fileName2, int iCaseSensitivity)
- static ZPOS64_T unz64local_SearchCentralDir _Z_OF ((const zlib_filefunc64_32_def *pzlib_filefunc_def, voidpf filestream))
- static ZPOS64_T unz64local_SearchCentralDir (const zlib_filefunc64_32_def *pzlib_filefunc_def, voidpf filestream)
- static ZPOS64_T unz64local_SearchCentralDir64 (const zlib_filefunc64_32_def *pzlib_filefunc_def, voidpf filestream)
- unzFile unzOpenInternal (voidpf file, zlib_filefunc64_32_def *pzlib_filefunc64_32_def, int is64bitOpenFunction, unsigned flags)
- unzFile ZEXPORT unzOpen2 (voidpf file, zlib_filefunc_def *pzlib_filefunc32_def)
- unzFile ZEXPORT unzOpen2_64 (voidpf file, zlib_filefunc64_def *pzlib_filefunc_def)
- unzFile ZEXPORT unzOpen (voidpf file)
- unzFile ZEXPORT unzOpen64 (voidpf file)
- int ZEXPORT unzClose (unzFile file)
- int ZEXPORT unzGetGlobalInfo64 (unzFile file, unz_global_info64 *pglobal_info)
- int ZEXPORT unzGetGlobalInfo (unzFile file, unz_global_info *pglobal_info32)
- static void unz64local_DosDateToTmuDate (ZPOS64_T ulDosDate, tm_unz *ptm)
- static int unz64local_GetCurrentFileInfoInternal _Z_OF ((unzFile file, unz_file_info64 *pfile_info, unz_file_info64_internal *pfile_info_internal, char *szFileName, uLong fileNameBufferSize, void *extraField, uLong extraFieldBufferSize, char *szComment, uLong commentBufferSize))
- static int unz64local_GetCurrentFileInfoInternal (unzFile file, unz_file_info64 *pfile_info, unz_file_info64_internal *pfile_info_internal, char *szFileName, uLong fileNameBufferSize, void *extraField, uLong extraFieldBufferSize, char *szComment, uLong commentBufferSize)
- int ZEXPORT unzGetCurrentFileInfo64 (unzFile file, unz_file_info64 *pfile_info, char *szFileName, uLong fileNameBufferSize, void *extraField, uLong extraFieldBufferSize, char *szComment, uLong commentBufferSize)
- int ZEXPORT unzGetCurrentFileInfo (unzFile file, unz_file_info *pfile_info, char *szFileName, uLong fileNameBufferSize, void *extraField, uLong extraFieldBufferSize, char *szComment, uLong commentBufferSize)
- int ZEXPORT unzGoToFirstFile (unzFile file)
- int ZEXPORT unzGoToNextFile (unzFile file)
- int ZEXPORT unzLocateFile (unzFile file, const char *szFileName, int iCaseSensitivity)
- int ZEXPORT unzGetFilePos64 (unzFile file, unz64_file_pos *file_pos)
- int ZEXPORT unzGetFilePos (unzFile file, unz_file_pos *file_pos)
- int ZEXPORT unzGoToFilePos64 (unzFile file, const unz64_file_pos *file_pos)
- int ZEXPORT unzGoToFilePos (unzFile file, unz_file_pos *file_pos)
- static int unz64local_CheckCurrentFileCoherencyHeader (unz64_s *s, uInt *piSizeVar, ZPOS64_T *poffset_local_extrafield, uInt *psize_local_extrafield)
- int ZEXPORT unzOpenCurrentFile3 (unzFile file, int *method, int *level, int raw, const char *password)
- int ZEXPORT unzOpenCurrentFile (unzFile file)
- int ZEXPORT unzOpenCurrentFilePassword (unzFile file, const char *password)
- int ZEXPORT unzOpenCurrentFile2 (unzFile file, int *method, int *level, int raw)
- ZPOS64_T ZEXPORT unzGetCurrentFileZStreamPos64 (unzFile file)

  Addition for GDAL : START.
- int ZEXPORT unzReadCurrentFile (unzFile file, voidp buf, unsigned len)

Addition for GDAL : END.

- z_off_t ZEXPORT unztell (unzFile file)
- ZPOS64_T ZEXPORT unztell64 (unzFile file)
- int ZEXPORT unzeof (unzFile file)
- int ZEXPORT unzGetLocalExtrafield (unzFile file, voidp buf, unsigned len)
- int ZEXPORT unzCloseCurrentFile (unzFile file)
- int ZEXPORT unzGetGlobalComment (unzFile file, char *szComment, uLong uSizeBuf)
- ZPOS64_T ZEXPORT unzGetOffset64 (unzFile file)
- uLong ZEXPORT unzGetOffset (unzFile file)
- int ZEXPORT unzSetOffset64 (unzFile file, ZPOS64_T pos)
- int ZEXPORT unzSetOffset (unzFile file, uLong pos)
- int ZEXPORT unzSetFlags (unzFile file, unsigned flags)
- int ZEXPORT unzClearFlags (unzFile file, unsigned flags)

Variables

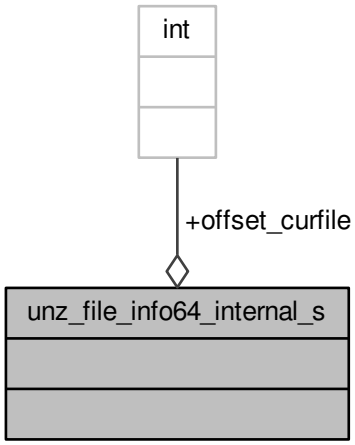
- const char unz_copyright []

13.107.1 Data Structure Documentation

13.107.1.1 struct unz_file_info64_internal_s

Definition at line 128 of file unzip.c.

Collaboration diagram for unz_file_info64_internal_s:



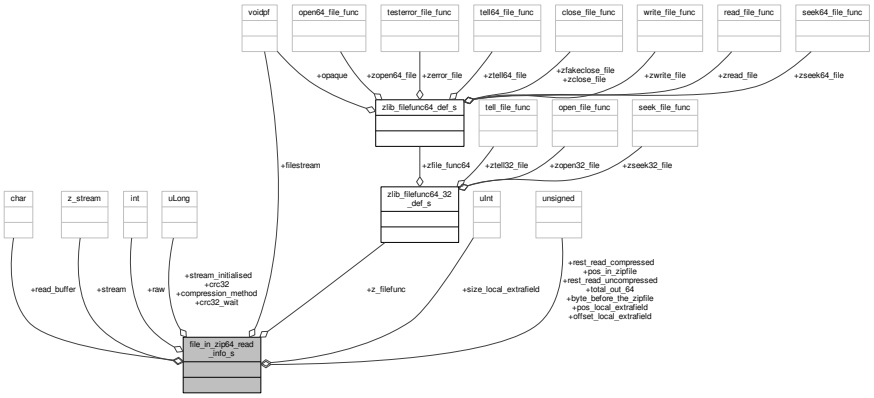
Data Fields

ZPOS64_T	offset_curfile	

13.107.1.2 struct file_in_zip64_read_info_s

Definition at line 136 of file unzip.c.

Collaboration diagram for file_in_zip64_read_info_s:



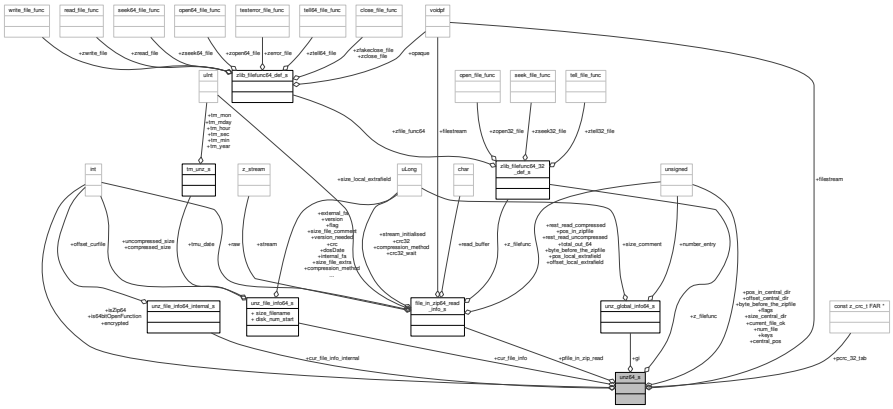
Data Fields

ZPOS64_T	byte_before_the-_zipfile	
uLong	compression_-method	
uLong	crc32	
uLong	crc32_wait	
voidpf	filestream	
ZPOS64_T	offset_local_-extrafield	
ZPOS64_T	pos_in_zipfile	
ZPOS64_T	pos_local_-extrafield	
int	raw	
char *	read_buffer	
ZPOS64_T	rest_read_-compressed	
ZPOS64_T	rest_read_-uncompressed	
uInt	size_local_-extrafield	
z_stream	stream	
uLong	stream_-initialised	
ZPOS64_T	total_out_64	
zlib_filefunc64_-32_def	z_filefunc	

13.107.1.3 struct unz64_s

Definition at line 167 of file unzip.c.

Collaboration diagram for unz64_s:



Data Fields

ZPOS64_T	byte_before_the-_zipfile	
ZPOS64_T	central_pos	
unz_file_info64	cur_file_info	
unz_file_info64_-internal	cur_file_info_-internal	
ZPOS64_T	current_file_ok	
int	encrypted	
voidpf	filestream	
unsigned	flags	
unz_global_-info64	gi	
int	is64bitOpen-Function	
int	isZip64	
unsigned long	keys[3]	
ZPOS64_T	num_file	
ZPOS64_T	offset_central_-dir	
const z_crc_t FAR *	pcrc_32_tab	
file_in_zip64_-read_info_s *	pfile_in_zip_read	
ZPOS64_T	pos_in_central_-dir	
ZPOS64_T	size_central_dir	
zlib_filefunc64_-32_def	z_filefunc	

13.107.2 Macro Definition Documentation

13.107.2.1 #define ALLOC(*size*) (malloc(size))

Definition at line 114 of file unzip.c.

Referenced by unz64local_SearchCentralDir(), unz64local_SearchCentralDir64(), unzOpenCurrentFile3(), and

unzOpenInternal().

13.107.2.2 #define BUFREADCOMMENT (0x400)

Definition at line 408 of file unzip.c.

Referenced by unz64local_SearchCentralDir(), and unz64local_SearchCentralDir64().

13.107.2.3 #define CASESENSITIVITYDEFAULT_NO

Definition at line 100 of file unzip.c.

13.107.2.4 #define CASESENSITIVITYDEFAULTVALUE 2

Definition at line 375 of file unzip.c.

Referenced by unzStringFileNameCompare().

13.107.2.5 #define local static

Definition at line 93 of file unzip.c.

13.107.2.6 #define SIZECENTRALDIRITEM (0x2e)

Definition at line 120 of file unzip.c.

Referenced by unzGoToNextFile().

13.107.2.7 #define SIZEZIPLOCALHEADER (0x1e)

Definition at line 121 of file unzip.c.

Referenced by unz64local_CheckCurrentFileCoherencyHeader(), and unzOpenCurrentFile3().

13.107.2.8 #define STRCMPCASENOSENTIVEFUNCTION strcmpcasenosensitive_internal

Definition at line 381 of file unzip.c.

Referenced by unzStringFileNameCompare().

13.107.2.9 #define TRYFREE(*p*) {if (p) free(p);}

Definition at line 117 of file unzip.c.

Referenced by unz64local_SearchCentralDir(), unz64local_SearchCentralDir64(), unzClose(), unzCloseCurrentFile(), and unzOpenCurrentFile3().

13.107.2.10 #define UNZ_BUFSIZE (16384)

Definition at line 106 of file unzip.c.

Referenced by unzOpenCurrentFile3(), and unzReadCurrentFile().

13.107.2.11 #define UNZ_MAXFILENAMEINZIP (256)

Definition at line 110 of file unzip.c.

Referenced by unzLocateFile().

13.107.3 Typedef Documentation

13.107.3.1 typedef struct unz_file_info64_internal_s unz_file_info64_internal

13.107.3.2 typedef uLongf z_crc_t

Definition at line 76 of file unzip.c.

13.107.4 Function Documentation

13.107.4.1 static int unz64local_getByte _Z_OF ((const zlib_filefunc64_32_def *pzlib_filefunc_def, voidpf filestream, int *pi)) [static]

13.107.4.2 static int unz64local_getShort _Z_OF ((const zlib_filefunc64_32_def *pzlib_filefunc_def, voidpf filestream, uLong *pX)) [static]

13.107.4.3 static int unz64local_getLong64 _Z_OF ((const zlib_filefunc64_32_def *pzlib_filefunc_def, voidpf filestream, ZPOS64_T *pX)) [static]

13.107.4.4 static ZPOS64_T unz64local_SearchCentralDir _Z_OF ((const zlib_filefunc64_32_def *pzlib_filefunc_def, voidpf filestream)) [static]

13.107.4.5 static int unz64local_GetCurrentFileInfoInternal _Z_OF ((unzFile file, unz_file_info64 *pfile_info, unz_file_info64_internal *pfile_info_internal, char *szFileName, uLong fileNameBufferSize, void *extraField, uLong extraFieldBufferSize, char *szComment, uLong commentBufferSize)) [static]

13.107.4.6 static int strcmpcasenosensitive_internal (const char * *fileName1*, const char * *fileName2*) [static]

Definition at line 352 of file unzip.c.

```
353 {
354     for (;;)
355     {
356         char c1=*(fileName1++);
357         char c2=*(fileName2++);
358         if ((c1>='a') && (c1<='z'))
359             c1 -= 0x20;
360         if ((c2>='a') && (c2<='z'))
361             c2 -= 0x20;
362         if (c1=='\0')
363             return ((c2=='\0') ? 0 : -1);
364         if (c2=='\0')
365             return 1;
366         if (c1<c2)
367             return -1;
368         if (c1>c2)
369             return 1;
370     }
371 }
```

13.107.4.7 static int unz64local_CheckCurrentFileCoherencyHeader (unz64_s * *s*, uInt * *piSizeVar*, ZPOS64_T * *poffset_local_extrafield*, uInt * *psize_local_extrafield*) [static]

Definition at line 1393 of file unzip.c.

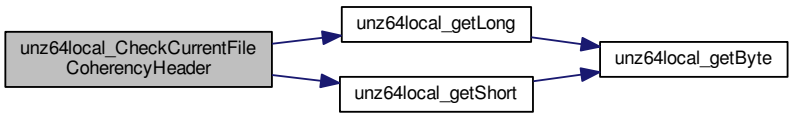
References unz64_s::byte_before_the_zipfile, unz_file_info64_s::compressed_size, unz_file_info64_s::compression_method, unz_file_info64_s::crc, unz64_s::cur_file_info, unz64_s::cur_file_info_internal, unz64_s::filestream, unz_file_info64_internal_s::offset_curfile, unz_file_info64_s::size_filename, SIZEZIPLOCALHEADER, unz_file_info64_s::uncompressed_size, unz64local_getLong(), unz64local_getShort(), UNZ_BADZIPFILE, UNZ_ERRNO, UNZ_OK, Z_BZIP2ED, unz64_s::z_filefunc, ZLIB_FILEFUNC_SEEK_SET, and ZSEEK64.

Referenced by unzOpenCurrentFile3().

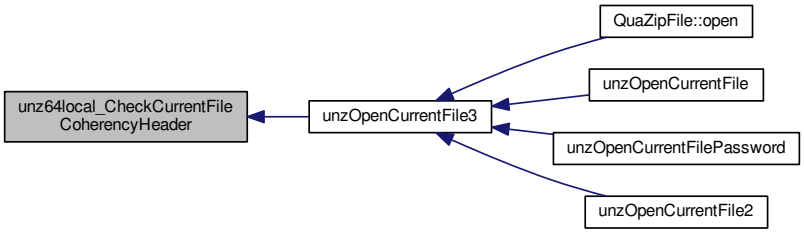
```
1396 {
1397     uLong uMagic,uData,uFlags;
1398     uLong size_filename;
1399     uLong size_extra_field;
1400     int err=UNZ_OK;
1401
1402     *piSizeVar = 0;
1403     *poffset_local_extrafield = 0;
1404     *psize_local_extrafield = 0;
1405
1406     if (ZSEEK64(s->z_filefunc, s->filestream,s->
     cur_file_info_internal.offset_curfile +
1407                                 s->byte_before_the_zipfile,
     ZLIB_FILEFUNC_SEEK_SET)!=0)
1408         return UNZ_ERRNO;
1409
1410
1411     if (err==UNZ_OK)
1412     {
1413         if (unz64local_getLong(&s->z_filefunc, s->
     filestream,&uMagic) != UNZ_OK)
1414             err=UNZ_ERRNO;
1415         else if (uMagic!=0x04034b50)
1416             err=UNZ_BADZIPFILE;
1417     }
1418
1419     if (unz64local_getShort(&s->z_filefunc, s->
     filestream,&uData) != UNZ_OK)
1420         err=UNZ_ERRNO;
1421 /*
1422     else if ((err==UNZ_OK) && (uData!=s->cur_file_info.wVersion))
1423         err=UNZ_BADZIPFILE;
1424 */
1425     if (unz64local_getShort(&s->z_filefunc, s->
     filestream,&uFlags) != UNZ_OK)
1426         err=UNZ_ERRNO;
1427
1428     if (unz64local_getShort(&s->z_filefunc, s->
     filestream,&uData) != UNZ_OK)
1429         err=UNZ_ERRNO;
1430     else if ((err==UNZ_OK) && (uData!=s->cur_file_info.
     compression_method))
1431         err=UNZ_BADZIPFILE;
1432
1433     if ((err==UNZ_OK) && (s->cur_file_info.compression_method!=0) &&
1434 /* #ifdef HAVE_BZIP2 */
1435                         (s->cur_file_info.compression_method!=
     Z_BZIP2ED) &&
1436 /* #endif */
1437                         (s->cur_file_info.compression_method!=Z_DEFLATED))
1438         err=UNZ_BADZIPFILE;
1439
1440     if (unz64local_getLong(&s->z_filefunc, s->
     filestream,&uData) != UNZ_OK) /* date/time */
1441         err=UNZ_ERRNO;
1442
1443     if (unz64local_getLong(&s->z_filefunc, s->
     filestream,&uData) != UNZ_OK) /* crc */
1444         err=UNZ_ERRNO;
1445     else if ((err==UNZ_OK) && (uData!=s->cur_file_info.crc) && ((uFlags & 8)==0))
1446         err=UNZ_BADZIPFILE;
1447
1448     if (unz64local_getLong(&s->z_filefunc, s->
     filestream,&uData) != UNZ_OK) /* size compr */
1449         err=UNZ_ERRNO;
1450     else if (uData != 0xFFFFFFFF && (err==UNZ_OK) && (uData!=s->
     cur_file_info.compressed_size) && ((uFlags & 8)==0))
1451         err=UNZ_BADZIPFILE;
1452
1453     if (unz64local_getLong(&s->z_filefunc, s->
     filestream,&uData) != UNZ_OK) /* size uncompr */
1454         err=UNZ_ERRNO;
1455     else if (uData != 0xFFFFFFFF && (err==UNZ_OK) && (uData!=s->
     cur_file_info.uncompressed_size) && ((uFlags & 8)==0))
1456         err=UNZ_BADZIPFILE;
1457
1458     if (unz64local_getShort(&s->z_filefunc, s->
```

```
      filestream,&size_filename) != UNZ_OK)
1459        err=UNZ_ERRNO;
1460    else if ((err==UNZ_OK) && (size_filename!=s->cur_file_info.
      size_filename))
1461        err=UNZ_BADZIPFILE;
1462
1463    *piSizeVar += (uInt)size_filename;
1464
1465    if (unz64local_getShort(&s->z_filefunc, s->
      filestream,&size_extra_field) != UNZ_OK)
1466        err=UNZ_ERRNO;
1467    *poffset_local_extrafield= s->cur_file_info_internal.
      offset_curfile +
1468                                SIZEZIPLOCALHEADER + size_filename;
1469    *psize_local_extrafield = (uInt)size_extra_field;
1470
1471    *piSizeVar += (uInt)size_extra_field;
1472
1473    return err;
1474 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



13.107.4.8 static void unz64local_DosDateToTmuDate (ZPOS64_T *ulDosDate*, tm_unz * *ptm*) [static]

Definition at line 862 of file unzip.c.

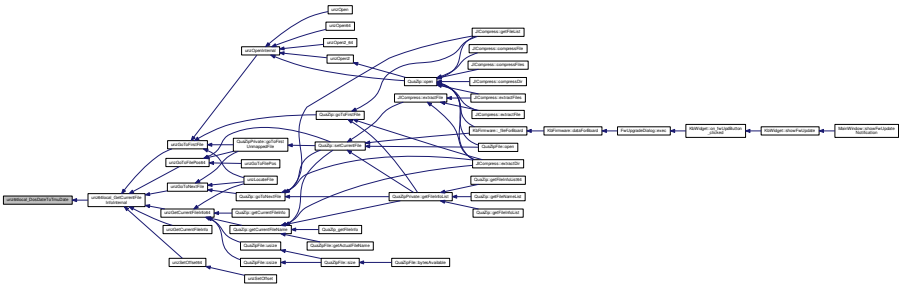
References tm_unz_s::tm_hour, tm_unz_s::tm_mday, tm_unz_s::tm_min, tm_unz_s::tm_mon, tm_unz_s::tm_sec, and tm_unz_s::tm_year.

Referenced by unz64local_GetCurrentFileInfoInternal().

```
863 {
864     ZPOS64_T uDate;
865     uDate = (ZPOS64_T)(ulDosDate>>16);
866     ptm->tm_mday = (uInt)(uDate&0x1f) ;
867     ptm->tm_mon =  (uInt)((((uDate)&0x1E0)/0x20)-1) ;
868     ptm->tm_year = (uInt)(((uDate&0x0FE00)/0x0200)+1980) ;
869
870     ptm->tm_hour = (uInt) ((ulDosDate &0xF800)/0x800);
871     ptm->tm_min =  (uInt) ((ulDosDate&0x7E0)/0x20) ;
```

```
872     ptm->tm_sec =  (uInt) (2*(ulDosDate&0x1f)) ;
873 }
```

Here is the caller graph for this function:



13.107.4.9 static int unz64local_getByte (const zlib_filefunc64_32_def * *pzlib_filefunc_def,* voidpf *filestream,* int * *pi*) [static]

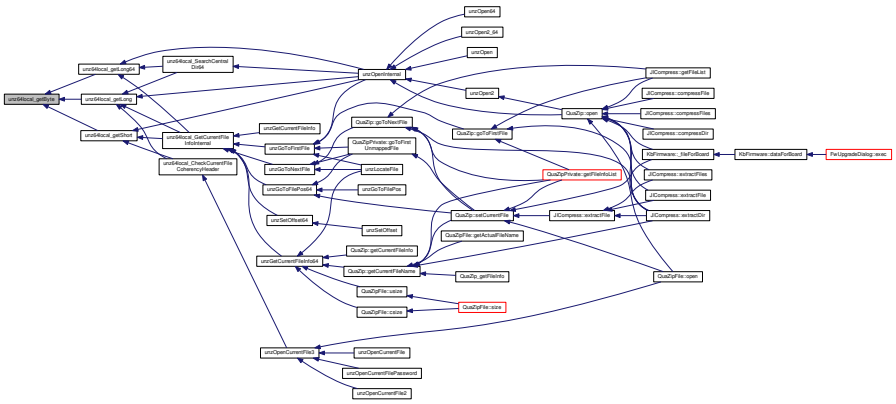
Definition at line 215 of file unzip.c.

References UNZ_EOF, UNZ_ERRNO, UNZ_OK, ZERROR64, and ZREAD64.

Referenced by unz64local_getLong(), unz64local_getLong64(), and unz64local_getShort().

```
216 {
217     unsigned char c;
218     int err = (int)ZREAD64(*pzlib_filefunc_def,filestream,&c,1);
219     if (err==1)
220     {
221         *pi = (int)c;
222         return UNZ_OK;
223     }
224     else
225     {
226         if (ZERROR64(*pzlib_filefunc_def,filestream))
227             return UNZ_ERRNO;
228         else
229             return UNZ_EOF;
230     }
231 }
```

Here is the caller graph for this function:



13.107.4.10 static int unz64local_GetCurrentFileInfoInternal (unzFile *file*, unz_file_info64 * *pfile_info*, unz_file_info64_internal * *pfile_info_internal*, char * *szFileName*, uLong *fileNameBufferSize*, void * *extraField*, uLong *extraFieldBufferSize*, char * *szComment*, uLong *commentBufferSize*) [static]

Definition at line 889 of file unzip.c.

References unz64_s::byte_before_the_zipfile, unz_file_info64_s::compressed_size, unz_file_info64_s::compression_method, unz_file_info64_s::crc, unz_file_info64_s::disk_num_start, unz_file_info64_s::dosDate, unz_file_info64_s::external_fa, unz64_s::filestream, unz_file_info64_s::flag, unz_file_info64_s::internal_fa, unz_file_info64_internal_s::offset_curfile, unz64_s::pos_in_central_dir, unz_file_info64_s::size_file_comment, unz_file_info64_s::size_file_extra, unz_file_info64_s::size_filename, unz_file_info64_s::tmu_date, unz_file_info64_s::uncompressed_size, unz64local_DosDateToTmuDate(), unz64local_getLong(), unz64local_getLong64(), unz64local_getShort(), UNZ_BADZIPFILE, UNZ_ERRNO, UNZ_OK, UNZ_PARAMERROR, unz_file_info64_s::version, unz_file_info64_s::version_needed, unz64_s::z_filefunc, ZLIB_FILEFUNC_SEEK_CUR, ZLIB_FILEFUNC_SEEK_SET, ZREAD64, and ZSEEK64.

Referenced by unzGetCurrentFileInfo(), unzGetCurrentFileInfo64(), unzGoToFilePos64(), unzGoToFirstFile(), unzGoToNextFile(), and unzSetOffset64().

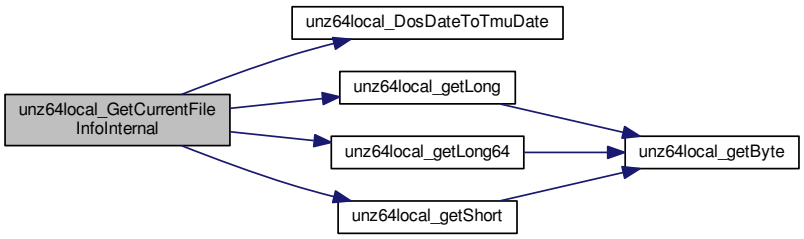
```
899 {
900     unz64_s* s;
901     unz_file_info64 file_info;
902     unz_file_info64_internal file_info_internal;
903     int err=UNZ_OK;
904     uLong uMagic;
905     ZPOS64_T llSeek=0;
906     uLong uL;
907
908     if (file==NULL)
909         return UNZ_PARAMERROR;
910     s=(unz64_s*)file;
911     if (ZSEEK64(s->z_filefunc, s->filestream,
912               s->pos_in_central_dir+s->byte_before_the_zipfile,
913               ZLIB_FILEFUNC_SEEK_SET)!=0)
914         err=UNZ_ERRNO;
915
916
917     /* we check the magic */
918     if (err==UNZ_OK)
919     {
920         if (unz64local_getLong(&s->z_filefunc, s->
    filestream,&uMagic) != UNZ_OK)
921             err=UNZ_ERRNO;
922         else if (uMagic!=0x02014b50)
923             err=UNZ_BADZIPFILE;
924     }
925
926     if (unz64local_getShort(&s->z_filefunc, s->
    filestream,&file_info.version) != UNZ_OK)
927         err=UNZ_ERRNO;
928
929     if (unz64local_getShort(&s->z_filefunc, s->
    filestream,&file_info.version_needed) != UNZ_OK)
930         err=UNZ_ERRNO;
931
932     if (unz64local_getShort(&s->z_filefunc, s->
    filestream,&file_info.flag) != UNZ_OK)
933         err=UNZ_ERRNO;
934
935     if (unz64local_getShort(&s->z_filefunc, s->
    filestream,&file_info.compression_method) != UNZ_OK)
936         err=UNZ_ERRNO;
937
938     if (unz64local_getLong(&s->z_filefunc, s->
    filestream,&file_info.dosDate) != UNZ_OK)
939         err=UNZ_ERRNO;
940
941     unz64local_DosDateToTmuDate(file_info.dosDate,&file_info.
    tmu_date);
942
943     if (unz64local_getLong(&s->z_filefunc, s->
    filestream,&file_info.crc) != UNZ_OK)
944         err=UNZ_ERRNO;
945
946     if (unz64local_getLong(&s->z_filefunc, s->
    filestream,&uL) != UNZ_OK)
947         err=UNZ_ERRNO;
948     file_info.compressed_size = uL;
949
950     if (unz64local_getLong(&s->z_filefunc, s->
```

```
    filestream,&uL) != UNZ_OK)
951         err=UNZ_ERRNO;
952       file_info.uncompressed_size = uL;
953
954       if (unz64local_getShort(&s->z_filefunc, s->
    filestream,&file_info.size_filename) != UNZ_OK)
955         err=UNZ_ERRNO;
956
957       if (unz64local_getShort(&s->z_filefunc, s->
    filestream,&file_info.size_file_extra) != UNZ_OK)
958         err=UNZ_ERRNO;
959
960       if (unz64local_getShort(&s->z_filefunc, s->
    filestream,&file_info.size_file_comment) != UNZ_OK)
961         err=UNZ_ERRNO;
962
963       if (unz64local_getShort(&s->z_filefunc, s->
    filestream,&file_info.disk_num_start) != UNZ_OK)
964         err=UNZ_ERRNO;
965
966       if (unz64local_getShort(&s->z_filefunc, s->
    filestream,&file_info.internal_fa) != UNZ_OK)
967         err=UNZ_ERRNO;
968
969       if (unz64local_getLong(&s->z_filefunc, s->
    filestream,&file_info.external_fa) != UNZ_OK)
970         err=UNZ_ERRNO;
971
972                 /* relative offset of local header */
973       if (unz64local_getLong(&s->z_filefunc, s->
    filestream,&uL) != UNZ_OK)
974         err=UNZ_ERRNO;
975       file_info_internal.offset_curfile = uL;
976
977       llSeek+=file_info.size_filename;
978       if ((err==UNZ_OK) && (szFileName!=NULL))
979       {
980           uLong uSizeRead ;
981           if (file_info.size_filename<fileNameBufferSize)
982           {
983               *(szFileName+file_info.size_filename)='\0';
984               uSizeRead = file_info.size_filename;
985           }
986           else
987               uSizeRead = fileNameBufferSize;
988
989           if ((file_info.size_filename>0) && (fileNameBufferSize>0))
990               if (ZREAD64(s->z_filefunc, s->filestream,szFileName,uSizeRead)!=
    uSizeRead)
991                   err=UNZ_ERRNO;
992           llSeek -= uSizeRead;
993       }
994
995       /* Read extrafield */
996       if ((err==UNZ_OK) && (extraField!=NULL))
997       {
998           ZPOS64_T uSizeRead ;
999           if (file_info.size_file_extra<extraFieldBufferSize)
1000              uSizeRead = file_info.size_file_extra;
1001          else
1002              uSizeRead = extraFieldBufferSize;
1003
1004          if (llSeek!=0)
1005          {
1006              if (ZSEEK64(s->z_filefunc, s->filestream,llSeek,
    ZLIB_FILEFUNC_SEEK_CUR)==0)
1007                  llSeek=0;
1008              else
1009                  err=UNZ_ERRNO;
1010          }
1011
1012          if ((file_info.size_file_extra>0) && (extraFieldBufferSize>0))
1013              if (ZREAD64(s->z_filefunc, s->filestream,extraField,(uLong)uSizeRead
    )!=uSizeRead)
1014                  err=UNZ_ERRNO;
1015
1016          llSeek += file_info.size_file_extra - (uLong)uSizeRead;
1017      }
1018      else
1019          llSeek += file_info.size_file_extra;
1020
1021
1022      if ((err==UNZ_OK) && (file_info.size_file_extra != 0))
1023      {
1024                                  uLong acc = 0;
1025
1026          /* since lSeek now points to after the extra field we need to move back */
```

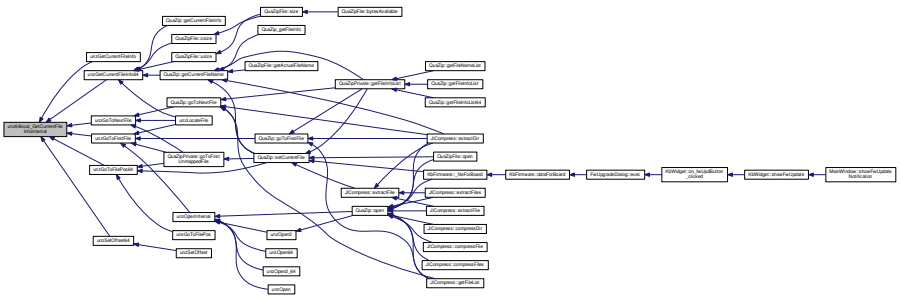
```
1027         llSeek -= file_info.size_file_extra;
1028
1029         if (llSeek!=0)
1030         {
1031             if (ZSEEK64(s->z_filefunc, s->filestream,llSeek,
     ZLIB_FILEFUNC_SEEK_CUR)==0)
1032                 llSeek=0;
1033             else
1034                 err=UNZ_ERRNO;
1035         }
1036
1037         while(acc < file_info.size_file_extra)
1038         {
1039             uLong headerId;
1040                                                 uLong dataSize;
1041
1042             if (unz64local_getShort(&s->z_filefunc, s->
     filestream,&headerId) != UNZ_OK)
1043                 err=UNZ_ERRNO;
1044
1045             if (unz64local_getShort(&s->z_filefunc, s->
     filestream,&dataSize) != UNZ_OK)
1046                 err=UNZ_ERRNO;
1047
1048             /* ZIP64 extra fields */
1049             if (headerId == 0x0001)
1050             {
1051                 uLong uL;
1052
1053                 if(file_info.uncompressed_size == (ZPOS64_T)0xFFFFFFFFu)
1054                 {
1055                     if (unz64local_getLong64(&s->z_filefunc, s->
     filestream,&file_info.uncompressed_size) != UNZ_OK)
1056                         err=UNZ_ERRNO;
1057                 }
1058
1059                 if(file_info.compressed_size == (ZPOS64_T)0xFFFFFFFFu)
1060                 {
1061                     if (unz64local_getLong64(&s->z_filefunc, s->
     filestream,&file_info.compressed_size) != UNZ_OK)
1062                         err=UNZ_ERRNO;
1063                 }
1064
1065                 if(file_info_internal.offset_curfile == (ZPOS64_T)0xFFFFFFFFu)
1066                 {
1067                     /* Relative Header offset */
1068                     if (unz64local_getLong64(&s->z_filefunc, s->
     filestream,&file_info_internal.offset_curfile) != UNZ_OK)
1069                         err=UNZ_ERRNO;
1070                 }
1071
1072                 if(file_info.disk_num_start == 0xFFFFFFFFu)
1073                 {
1074                     /* Disk Start Number */
1075                     if (unz64local_getLong(&s->z_filefunc, s->
     filestream,&uL) != UNZ_OK)
1076                         err=UNZ_ERRNO;
1077                 }
1078
1079             }
1080             else
1081             {
1082                 if (ZSEEK64(s->z_filefunc, s->filestream,dataSize,
     ZLIB_FILEFUNC_SEEK_CUR)!=0)
1083                     err=UNZ_ERRNO;
1084             }
1085
1086             acc += 2 + 2 + dataSize;
1087         }
1088     }
1089
1090     if ((err==UNZ_OK) && (szComment!=NULL))
1091     {
1092         uLong uSizeRead ;
1093         if (file_info.size_file_comment<commentBufferSize)
1094         {
1095             *(szComment+file_info.size_file_comment)='\0';
1096             uSizeRead = file_info.size_file_comment;
1097         }
1098         else
1099             uSizeRead = commentBufferSize;
1100
1101         if (llSeek!=0)
1102         {
1103             if (ZSEEK64(s->z_filefunc, s->filestream,llSeek,
     ZLIB_FILEFUNC_SEEK_CUR)==0)
1104                 llSeek=0;
```

```
1105             else
1106                 err=UNZ_ERRNO;
1107         }
1108
1109         if ((file_info.size_file_comment>0) && (commentBufferSize>0))
1110             if (ZREAD64(s->z_filefunc, s->filestream,szComment,uSizeRead)!=
     uSizeRead)
1111                 err=UNZ_ERRNO;
1112         llSeek+=file_info.size_file_comment - uSizeRead;
1113     }
1114     else
1115         llSeek+=file_info.size_file_comment;
1116
1117
1118     if ((err==UNZ_OK) && (pfile_info!=NULL))
1119         *pfile_info=file_info;
1120
1121     if ((err==UNZ_OK) && (pfile_info_internal!=NULL))
1122         *pfile_info_internal=file_info_internal;
1123
1124     return err;
1125 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



13.107.4.11 static int unz64local_getLong (const zlib_filefunc64_32_def * *pzlib_filefunc_def,* voidpf *filestream,* uLong * *pX*) [static]

Definition at line 269 of file unzip.c.

References unz64local_getByte(), and UNZ_OK.

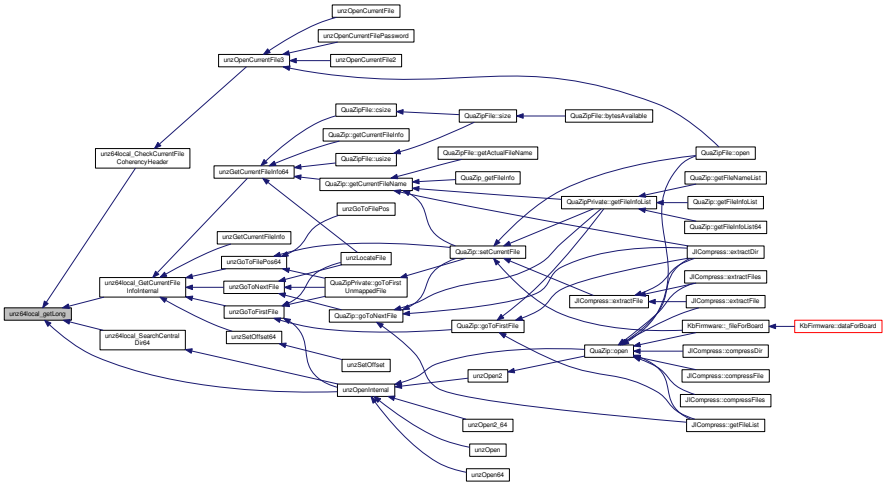
Referenced by unz64local_CheckCurrentFileCoherencyHeader(), unz64local_GetCurrentFileInfoInternal(), unz64local_SearchCentralDir64(), and unzOpenInternal().

```
272 {
273     uLong x ;
```

```
274     int i = 0;
275     int err;
276
277     err = unz64local_getByte(pzlib_filefunc_def,filestream,&i);
278     x = (uLong)i;
279
280     if (err==UNZ_OK)
281         err = unz64local_getByte(pzlib_filefunc_def,filestream,&i);
282     x |= ((uLong)i)<<8;
283
284     if (err==UNZ_OK)
285         err = unz64local_getByte(pzlib_filefunc_def,filestream,&i);
286     x |= ((uLong)i)<<16;
287
288     if (err==UNZ_OK)
289         err = unz64local_getByte(pzlib_filefunc_def,filestream,&i);
290     x += ((uLong)i)<<24;
291
292     if (err==UNZ_OK)
293         *pX = x;
294     else
295         *pX = 0;
296     return err;
297 }
```

Here is the call graph for this function:

Here is the caller graph for this function:



13.107.4.12 static int unz64local_getLong64 (const zlib_filefunc64_32_def * *pzlib_filefunc_def*, voidpf *filestream*, ZPOS64_T * *pX*) [static]

Definition at line 305 of file unzip.c.

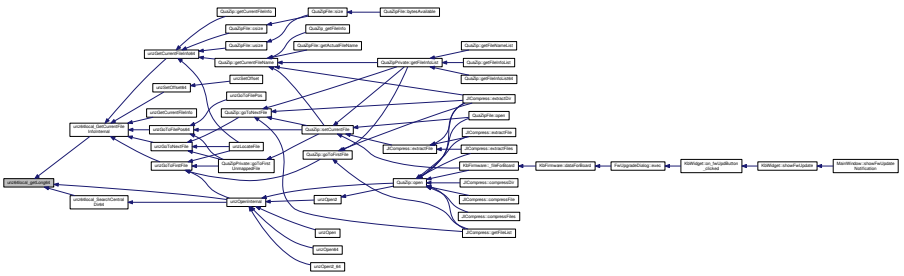
References unz64local_getByte(), and UNZ_OK.

Referenced by unz64local_GetCurrentFileInfoInternal(), unz64local_SearchCentralDir64(), and unzOpenInternal().

```
308 {
309     ZPOS64_T x ;
310     int i = 0;
311     int err;
312
313     err = unz64local_getByte(pzlib_filefunc_def,filestream,&i);
314     x = (ZPOS64_T)i;
315
316     if (err==UNZ_OK)
317         err = unz64local_getByte(pzlib_filefunc_def,filestream,&i);
318     x |= ((ZPOS64_T)i)<<8;
319
320     if (err==UNZ_OK)
321         err = unz64local_getByte(pzlib_filefunc_def,filestream,&i);
322     x |= ((ZPOS64_T)i)<<16;
323
324     if (err==UNZ_OK)
325         err = unz64local_getByte(pzlib_filefunc_def,filestream,&i);
326     x |= ((ZPOS64_T)i)<<24;
327
328     if (err==UNZ_OK)
329         err = unz64local_getByte(pzlib_filefunc_def,filestream,&i);
330     x |= ((ZPOS64_T)i)<<32;
331
332     if (err==UNZ_OK)
333         err = unz64local_getByte(pzlib_filefunc_def,filestream,&i);
334     x |= ((ZPOS64_T)i)<<40;
335
336     if (err==UNZ_OK)
337         err = unz64local_getByte(pzlib_filefunc_def,filestream,&i);
338     x |= ((ZPOS64_T)i)<<48;
339
340     if (err==UNZ_OK)
341         err = unz64local_getByte(pzlib_filefunc_def,filestream,&i);
342     x |= ((ZPOS64_T)i)<<56;
343
344     if (err==UNZ_OK)
345         *pX = x;
346     else
347         *pX = 0;
348     return err;
349 }
```

Here is the call graph for this function:

Here is the caller graph for this function:



13.107.4.13 static int unz64local_getShort (const zlib_filefunc64_32_def * *pzlib_filefunc_def,* voidpf *filestream,* uLong * *pX*) [static]

Definition at line 242 of file unzip.c.

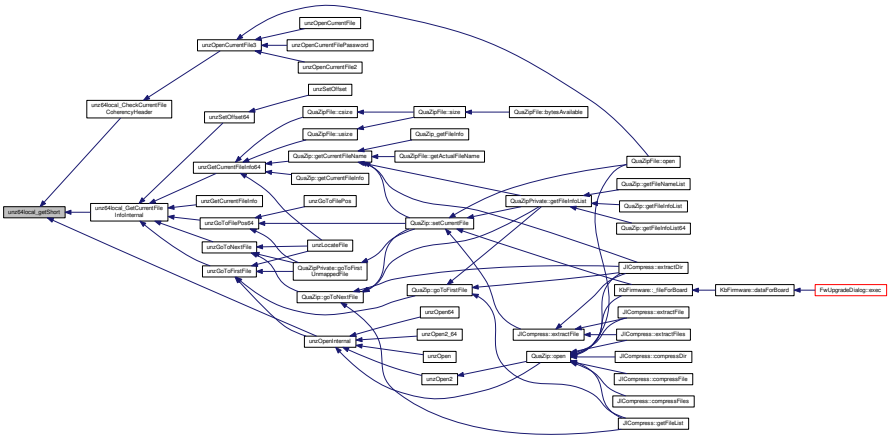
References unz64local_getByte(), and UNZ_OK.

Referenced by unz64local_CheckCurrentFileCoherencyHeader(), unz64local_GetCurrentFileInfoInternal(), and unzOpenInternal().

```
245 {
246     uLong x ;
247     int i = 0;
248     int err;
249
250     err = unz64local_getByte(pzlib_filefunc_def,filestream,&i);
251     x = (uLong)i;
252
253     if (err==UNZ_OK)
254         err = unz64local_getByte(pzlib_filefunc_def,filestream,&i);
255     x |= ((uLong)i)<<8;
256
257     if (err==UNZ_OK)
258         *pX = x;
259     else
260         *pX = 0;
261     return err;
262 }
```

Here is the call graph for this function:

Here is the caller graph for this function:



13.107.4.14 static ZPOS64_T unz64local_SearchCentralDir (const zlib_filefunc64_32_def * *pzlib_filefunc_def,* voidpf *filestream*) [static]

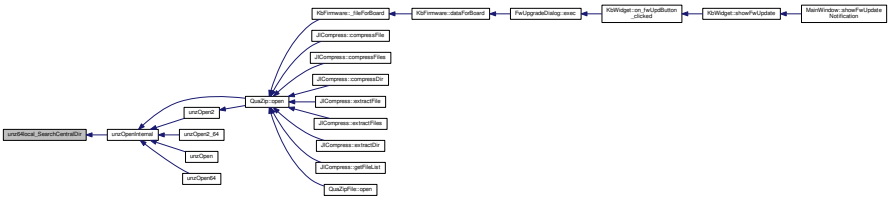
Definition at line 416 of file unzip.c.

References ALLOC, BUFREADCOMMENT, TRYFREE, ZLIB_FILEFUNC_SEEK_END, ZLIB_FILEFUNC_SEEK_-SET, ZREAD64, ZSEEK64, and ZTELL64.

Referenced by unzOpenInternal().

```
417 {
418     unsigned char* buf;
419     ZPOS64_T uSizeFile;
420     ZPOS64_T uBackRead;
421     ZPOS64_T uMaxBack=0xffff; /* maximum size of global comment */
422     ZPOS64_T uPosFound=0;
423
424     if (ZSEEK64(*pzlib_filefunc_def,filestream,0,ZLIB_FILEFUNC_SEEK_END) != 0)
425         return 0;
426
427
428     uSizeFile = ZTELL64(*pzlib_filefunc_def,filestream);
429
430     if (uMaxBack>uSizeFile)
431         uMaxBack = uSizeFile;
432
433     buf = (unsigned char*)ALLOC(BUFREADCOMMENT+4);
434     if (buf==NULL)
435         return 0;
436
437     uBackRead = 4;
438     while (uBackRead<uMaxBack)
439     {
440         uLong uReadSize;
441         ZPOS64_T uReadPos ;
442         int i;
443         if (uBackRead+BUFREADCOMMENT>uMaxBack)
444             uBackRead = uMaxBack;
445         else
446             uBackRead+=BUFREADCOMMENT;
447         uReadPos = uSizeFile-uBackRead ;
448
449         uReadSize = ((BUFREADCOMMENT+4) < (uSizeFile-uReadPos)) ?
450                      (BUFREADCOMMENT+4) : (uLong)(uSizeFile-uReadPos);
451         if (ZSEEK64(*pzlib_filefunc_def,filestream,uReadPos,
    ZLIB_FILEFUNC_SEEK_SET)!=0)
452             break;
453
454         if (ZREAD64(*pzlib_filefunc_def,filestream,buf,uReadSize)!=uReadSize)
455             break;
456
457         for (i=(int)uReadSize-3; (i--)>0;)
458             if (((*(buf+i))==0x50) && ((*(buf+i+1))==0x4b) &&
459                 ((*(buf+i+2))==0x05) && ((*(buf+i+3))==0x06))
460             {
461                 uPosFound = uReadPos+i;
462                 break;
463             }
464
465         if (uPosFound!=0)
466             break;
467     }
468     TRYFREE(buf);
469     return uPosFound;
470 }
```

Here is the caller graph for this function:



13.107.4.15 static ZPOS64_T unz64local_SearchCentralDir64 (const zlib_filefunc64_32_def * *pzlib_filefunc_def,* voidpf *filestream*) [static]

Definition at line 481 of file unzip.c.

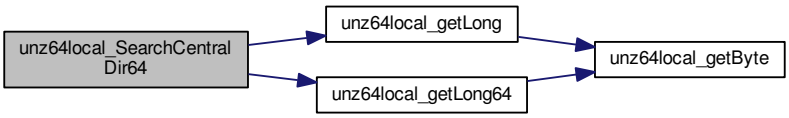
References ALLOC, BUFREADCOMMENT, TRYFREE, unz64local_getLong(), unz64local_getLong64(), UNZ_OK, ZLIB_FILEFUNC_SEEK_END, ZLIB_FILEFUNC_SEEK_SET, ZREAD64, ZSEEK64, and ZTELL64.

Referenced by unzOpenInternal().

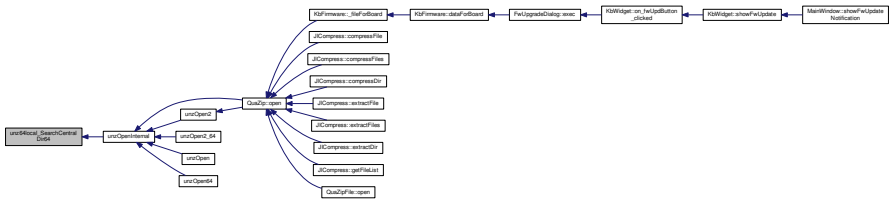
```
483 {
484     unsigned char* buf;
485     ZPOS64_T uSizeFile;
486     ZPOS64_T uBackRead;
487     ZPOS64_T uMaxBack=0xffff; /* maximum size of global comment */
488     ZPOS64_T uPosFound=0;
489     uLong uL;
490                 ZPOS64_T relativeOffset;
491
492     if (ZSEEK64(*pzlib_filefunc_def,filestream,0,ZLIB_FILEFUNC_SEEK_END) != 0)
493         return 0;
494
495
496     uSizeFile = ZTELL64(*pzlib_filefunc_def,filestream);
497
498     if (uMaxBack>uSizeFile)
499         uMaxBack = uSizeFile;
500
501     buf = (unsigned char*)ALLOC(BUFREADCOMMENT+4);
502     if (buf==NULL)
503         return 0;
504
505     uBackRead = 4;
506     while (uBackRead<uMaxBack)
507     {
508         uLong uReadSize;
509         ZPOS64_T uReadPos;
510         int i;
511         if (uBackRead+BUFREADCOMMENT>uMaxBack)
512             uBackRead = uMaxBack;
513         else
514             uBackRead+=BUFREADCOMMENT;
515         uReadPos = uSizeFile-uBackRead ;
516
517         uReadSize = ((BUFREADCOMMENT+4) < (uSizeFile-uReadPos)) ?
518                      (BUFREADCOMMENT+4) : (uLong)(uSizeFile-uReadPos);
519         if (ZSEEK64(*pzlib_filefunc_def,filestream,uReadPos,
    ZLIB_FILEFUNC_SEEK_SET)!=0)
520             break;
521
522         if (ZREAD64(*pzlib_filefunc_def,filestream,buf,uReadSize)!=uReadSize)
523             break;
524
525         for (i=(int)uReadSize-3; (i--)>0;)
526             if (((*(buf+i))==0x50) && ((*(buf+i+1))==0x4b) &&
527                 ((*(buf+i+2))==0x06) && ((*(buf+i+3))==0x07))
528             {
529                 uPosFound = uReadPos+i;
530                 break;
531             }
532
533         if (uPosFound!=0)
534             break;
535     }
536     TRYFREE(buf);
537     if (uPosFound == 0)
538         return 0;
539
540     /* Zip64 end of central directory locator */
541     if (ZSEEK64(*pzlib_filefunc_def,filestream, uPosFound,
    ZLIB_FILEFUNC_SEEK_SET)!=0)
542         return 0;
543
544     /* the signature, already checked */
545     if (unz64local_getLong(pzlib_filefunc_def,filestream,&uL)!=
    UNZ_OK)
546         return 0;
547
548     /* number of the disk with the start of the zip64 end of  central directory */
549     if (unz64local_getLong(pzlib_filefunc_def,filestream,&uL)!=
    UNZ_OK)
550         return 0;
551     if (uL != 0)
552         return 0;
```

```
553
554     /* relative offset of the zip64 end of central directory record */
555     if (unz64local_getLong64(pzlib_filefunc_def,filestream,&relativeOffset)!=
    UNZ_OK)
556         return 0;
557
558     /* total number of disks */
559     if (unz64local_getLong(pzlib_filefunc_def,filestream,&uL)!=
    UNZ_OK)
560         return 0;
561     if (uL != 1)
562         return 0;
563
564     /* Goto end of central directory record */
565     if (ZSEEK64(*pzlib_filefunc_def,filestream, relativeOffset,
    ZLIB_FILEFUNC_SEEK_SET)!=0)
566         return 0;
567
568      /* the signature */
569     if (unz64local_getLong(pzlib_filefunc_def,filestream,&uL)!=
    UNZ_OK)
570         return 0;
571
572     if (uL != 0x06064b50)
573         return 0;
574
575     return relativeOffset;
576 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



13.107.4.16 int ZEXPORT unzClearFlags (unzFile *file,* unsigned *flags*)

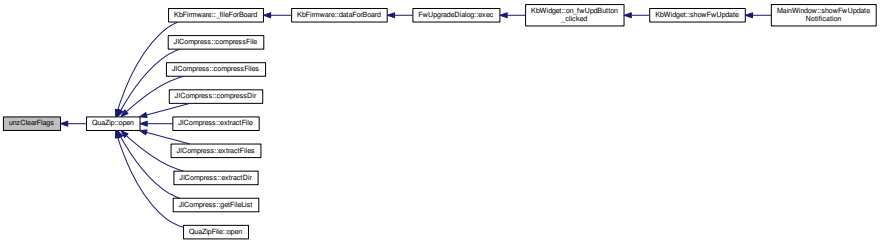
Definition at line 2147 of file unzip.c.

References unz64_s::flags, UNZ_OK, and UNZ_PARAMERROR.

Referenced by QuaZip::open().

```
2148 {
2149     unz64_s* s;
2150     if (file == NULL)
2151         return UNZ_PARAMERROR;
2152     s = (unz64_s*)file;
2153     s->flags &= ~flags;
2154     return UNZ_OK;
2155 }
```

Here is the caller graph for this function:



13.107.4.17 int ZEXPORT unzClose (unzFile *file*)

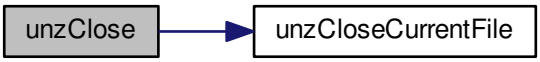
Definition at line 815 of file unzip.c.

References unz64_s::filestream, unz64_s::flags, unz64_s::pfile_in_zip_read, TRYFREE, UNZ_AUTO_CLOSE, UNZ_OK, UNZ_PARAMERROR, unzCloseCurrentFile(), unz64_s::z_filefunc, ZCLOSE64, and ZFAKECLOSE64.

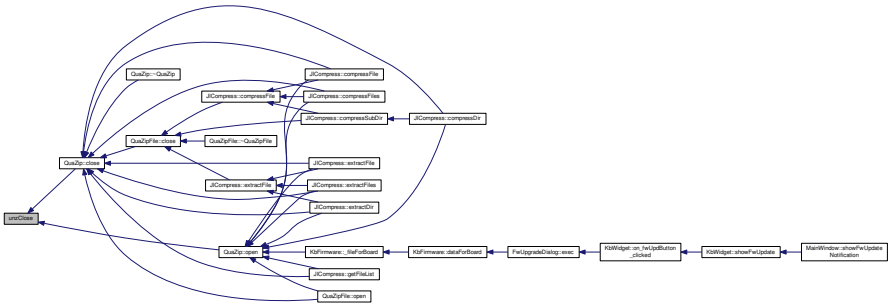
Referenced by QuaZip::close(), and QuaZip::open().

```
816 {
817     unz64_s* s;
818     if (file==NULL)
819         return UNZ_PARAMERROR;
820     s=(unz64_s*)file;
821 
822     if (s->pfile_in_zip_read!=NULL)
823         unzCloseCurrentFile(file);
824 
825     if ((s->flags & UNZ_AUTO_CLOSE) != 0)
826         ZCLOSE64(s->z_filefunc, s->filestream);
827     else
828         ZFAKECLOSE64(s->z_filefunc, s->filestream);
829     TRYFREE(s);
830     return UNZ_OK;
831 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



13.107.4.18 int ZEXPORT unzCloseCurrentFile (unzFile *file*)

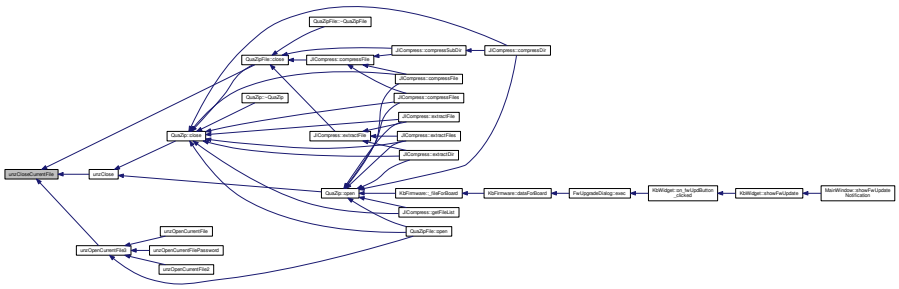
Definition at line 2012 of file unzip.c.

References file_in_zip64_read_info_s::crc32, file_in_zip64_read_info_s::crc32_wait, unz64_s::pfile_in_zip_read, file_in_zip64_read_info_s::raw, file_in_zip64_read_info_s::read_buffer, file_in_zip64_read_info_s::rest_read_uncompressed, file_in_zip64_read_info_s::stream, file_in_zip64_read_info_s::stream_initialised, TRYFREE, UNZ_CRCERROR, UNZ_OK, UNZ_PARAMERROR, and Z_BZIP2ED.

Referenced by QuaZipFile::close(), unzClose(), and unzOpenCurrentFile3().

```
2013 {
2014     int err=UNZ_OK;
2015
2016     unz64_s* s;
2017     file_in_zip64_read_info_s* pfile_in_zip_read_info;
2018     if (file==NULL)
2019         return UNZ_PARAMERROR;
2020     s=(unz64_s*)file;
2021     pfile_in_zip_read_info=s->pfile_in_zip_read;
2022
2023     if (pfile_in_zip_read_info==NULL)
2024         return UNZ_PARAMERROR;
2025
2026
2027     if ((pfile_in_zip_read_info->rest_read_uncompressed == 0) &&
2028         (!pfile_in_zip_read_info->raw))
2029     {
2030         if (pfile_in_zip_read_info->crc32 != pfile_in_zip_read_info->
    crc32_wait)
2031             err=UNZ_CRCERROR;
2032     }
2033
2034
2035     TRYFREE(pfile_in_zip_read_info->read_buffer);
2036     pfile_in_zip_read_info->read_buffer = NULL;
2037     if (pfile_in_zip_read_info->stream_initialised == Z_DEFLATED)
2038         inflateEnd(&pfile_in_zip_read_info->stream);
2039 #ifdef HAVE_BZIP2
2040     else if (pfile_in_zip_read_info->stream_initialised ==
    Z_BZIP2ED)
2041         BZ2_bzDecompressEnd(&pfile_in_zip_read_info->bstream);
2042 #endif
2043
2044
2045     pfile_in_zip_read_info->stream_initialised = 0;
2046     TRYFREE(pfile_in_zip_read_info);
2047
2048     s->pfile_in_zip_read=NULL;
2049
2050     return err;
2051 }
```

Here is the caller graph for this function:



13.107.4.19 int ZEXPORT unzeof (unzFile *file*)

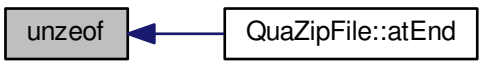
Definition at line 1932 of file unzip.c.

References unz64_s::pfile_in_zip_read, file_in_zip64_read_info_s::rest_read_uncompressed, and UNZ_PARAMERROR.

Referenced by QuaZipFile::atEnd().

```
1933 {
1934     unz64_s* s;
1935     file_in_zip64_read_info_s* pfile_in_zip_read_info;
1936     if (file==NULL)
1937         return UNZ_PARAMERROR;
1938     s=(unz64_s*)file;
1939     pfile_in_zip_read_info=s->pfile_in_zip_read;
1940
1941     if (pfile_in_zip_read_info==NULL)
1942         return UNZ_PARAMERROR;
1943
1944     if (pfile_in_zip_read_info->rest_read_uncompressed == 0)
1945         return 1;
1946     else
1947         return 0;
1948 }
```

Here is the caller graph for this function:



13.107.4.20 int ZEXPORT unzGetCurrentFileInfo (unzFile *file*, unz_file_info * *pfile_info*, char * *szFileName*, uLong *fileNameBufferSize*, void * *extraField*, uLong *extraFieldBufferSize*, char * *szComment*, uLong *commentBufferSize*)

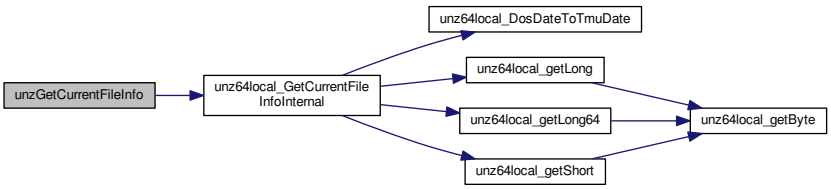
Definition at line 1146 of file unzip.c.

References unz_file_info64_s::compressed_size, unz_file_info_s::compressed_size, unz_file_info64_s::compression_method, unz_file_info_s::compression_method, unz_file_info64_s::crc, unz_file_info_s::crc, unz_file_info64_s::disk_num_start, unz_file_info_s::disk_num_start, unz_file_info64_s::dosDate, unz_file_info_s::dosDate,

unz_file_info64_s::external_fa, unz_file_info_s::external_fa, unz_file_info64_s::flag, unz_file_info_s::flag, unz_file_info64_s::internal_fa, unz_file_info_s::internal_fa, unz_file_info64_s::size_file_comment, unz_file_info_s::size_file_comment, unz_file_info64_s::size_file_extra, unz_file_info_s::size_file_extra, unz_file_info64_s::size_filename, unz_file_info_s::size_filename, unz_file_info64_s::tmu_date, unz_file_info_s::tmu_date, unz_file_info64_s::uncompressed_size, unz_file_info_s::uncompressed_size, unz64local_GetCurrentFileInfoInternal(), UNZ_OK, unz_file_info64_s::version, unz_file_info_s::version, unz_file_info64_s::version_needed, and unz_file_info_s::version_needed.

```
1151 {
1152     int err;
1153     unz_file_info64 file_info64;
1154     err = unz64local_GetCurrentFileInfoInternal(file,&file_info64,NULL
     ,
1155                                                 szFileName,fileNameBufferSize,
1156                                                 extraField,extraFieldBufferSize,
1157                                                 szComment,commentBufferSize);
1158     if (err==UNZ_OK && pfile_info != NULL)
1159     {
1160         pfile_info->version = file_info64.version;
1161         pfile_info->version_needed = file_info64.version_needed;
1162         pfile_info->flag = file_info64.flag;
1163         pfile_info->compression_method = file_info64.
     compression_method;
1164         pfile_info->dosDate = file_info64.dosDate;
1165         pfile_info->crc = file_info64.crc;
1166 
1167         pfile_info->size_filename = file_info64.size_filename;
1168         pfile_info->size_file_extra = file_info64.size_file_extra;
1169         pfile_info->size_file_comment = file_info64.
     size_file_comment;
1170 
1171         pfile_info->disk_num_start = file_info64.disk_num_start;
1172         pfile_info->internal_fa = file_info64.internal_fa;
1173         pfile_info->external_fa = file_info64.external_fa;
1174 
1175         pfile_info->tmu_date = file_info64.tmu_date,
1176 
1177 
1178         pfile_info->compressed_size = (uLong)file_info64.
     compressed_size;
1179         pfile_info->uncompressed_size = (uLong)file_info64.
     uncompressed_size;
1180 
1181     }
1182     return err;
1183 }
```

Here is the call graph for this function:



13.107.4.21 int ZEXPORT unzGetCurrentFileInfo64 (unzFile *file*, unz_file_info64 * *pfile_info*, char * *szFileName*, uLong *fileNameBufferSize*, void * *extraField*, uLong *extraFieldBufferSize*, char * *szComment*, uLong *commentBufferSize*)

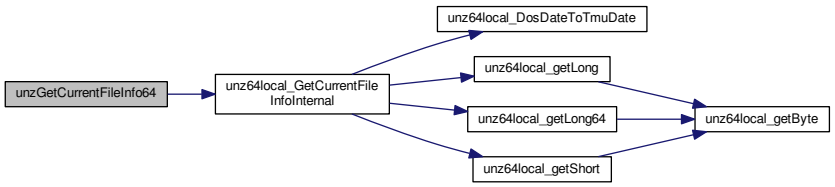
Definition at line 1134 of file unzip.c.

References unz64local_GetCurrentFileInfoInternal().

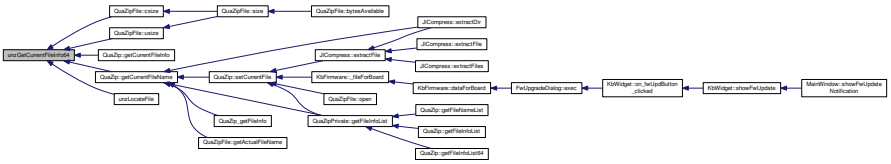
Referenced by QuaZipFile::csize(), QuaZip::getCurrentFileInfo(), QuaZip::getCurrentFileName(), unzLocateFile(), and QuaZipFile::usize().

```
1139 {
1140     return unz64local_GetCurrentFileInfoInternal(file,pfile_info,NULL,
1141                                                  szFileName,fileNameBufferSize,
1142                                                  extraField,extraFieldBufferSize,
1143                                                  szComment,commentBufferSize);
1144 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



13.107.4.22 ZPOS64_T ZEXPORT unzGetCurrentFileZStreamPos64 (unzFile *file*)

Definition at line 1666 of file unzip.c.

References file_in_zip64_read_info_s::byte_before_the_zipfile, unz64_s::pfile_in_zip_read, and file_in_zip64_-read_info_s::pos_in_zipfile.

```
1667 {
1668     unz64_s* s;
1669     file_in_zip64_read_info_s* pfile_in_zip_read_info;
1670     s=(unz64_s*)file;
1671     if (file==NULL)
1672         return 0; /*UNZ_PARAMERROR; */
1673     pfile_in_zip_read_info=s->pfile_in_zip_read;
1674     if (pfile_in_zip_read_info==NULL)
1675         return 0; /*UNZ_PARAMERROR; */
1676     return pfile_in_zip_read_info->pos_in_zipfile +
1677                          pfile_in_zip_read_info->byte_before_the_zipfile;
1678 }
```

13.107.4.23 int ZEXPORT unzGetFilePos (unzFile *file,* unz_file_pos * *file_pos*)

Definition at line 1334 of file unzip.c.

References unz_file_pos_s::num_of_file, unz64_file_pos_s::num_of_file, unz_file_pos_s::pos_in_zip_directory, unz64_file_pos_s::pos_in_zip_directory, UNZ_OK, and unzGetFilePos64().

```
1337 {
1338     unz64_file_pos file_pos64;
1339     int err = unzGetFilePos64(file,&file_pos64);
1340     if (err==UNZ_OK)
1341     {
```

```
1342         file_pos->pos_in_zip_directory = (uLong)file_pos64.
     pos_in_zip_directory;
1343         file_pos->num_of_file = (uLong)file_pos64.num_of_file;
1344     }
1345     return err;
1346 }
```

Here is the call graph for this function:

13.107.4.24 int ZEXPORT unzGetFilePos64 (unzFile *file*, unz64_file_pos * *file_pos*)

Definition at line 1318 of file unzip.c.

References unz64_s::current_file_ok, unz64_s::num_file, unz64_file_pos_s::num_of_file, unz64_s::pos_in_central_dir, unz64_file_pos_s::pos_in_zip_directory, UNZ_END_OF_LIST_OF_FILE, UNZ_OK, and UNZ_PARAMERROR.

Referenced by QuaZipPrivate::addCurrentFileToDirectoryMap(), and unzGetFilePos().

```
1319 {
1320     unz64_s* s;
1321
1322     if (file==NULL || file_pos==NULL)
1323         return UNZ_PARAMERROR;
1324     s=(unz64_s*)file;
1325     if (!s->current_file_ok)
1326         return UNZ_END_OF_LIST_OF_FILE;
1327
1328     file_pos->pos_in_zip_directory  = s->pos_in_central_dir;
1329     file_pos->num_of_file           = s->num_file;
1330
1331     return UNZ_OK;
1332 }
```

Here is the caller graph for this function:

13.107.4.25 int ZEXPORT unzGetGlobalComment (unzFile *file*, char * *szComment*, uLong *uSizeBuf*)

Definition at line 2059 of file unzip.c.

References unz64_s::central_pos, unz64_s::filestream, unz64_s::gi, unz_global_info64_s::size_comment, UNZ_ERRNO, UNZ_PARAMERROR, unz64_s::z_filefunc, ZLIB_FILEFUNC_SEEK_SET, ZREAD64, and ZSEEK64.

Referenced by QuaZip::getComment().

```
2060 {
2061     unz64_s* s;
2062     uLong uReadThis ;
2063     if (file==NULL)
2064         return (int)UNZ_PARAMERROR;
2065     s=(unz64_s*)file;
2066
2067     uReadThis = uSizeBuf;
2068     if (uReadThis>s->gi.size_comment)
2069         uReadThis = s->gi.size_comment;
2070
2071     if (ZSEEK64(s->z_filefunc,s->filestream,s->
     central_pos+22,ZLIB_FILEFUNC_SEEK_SET)!=0)
2072         return UNZ_ERRNO;
2073
2074     if (uReadThis>0)
2075     {
2076       *szComment='\0';
2077       if (ZREAD64(s->z_filefunc,s->filestream,szComment,uReadThis)!=uReadThis)
2078         return UNZ_ERRNO;
2079     }
2080
2081     if ((szComment != NULL) && (uSizeBuf > s->gi.size_comment))
2082         *(szComment+s->gi.size_comment)='\0';
2083     return (int)uReadThis;
2084 }
```

Here is the caller graph for this function:

13.107.4.26 int ZEXPORT unzGetGlobalInfo (unzFile *file*, unz_global_info * *pglobal_info32*)

Definition at line 848 of file unzip.c.

References unz64_s::gi, unz_global_info64_s::number_entry, unz_global_info_s::number_entry, unz_global_info64_s::size_comment, unz_global_info_s::size_comment, UNZ_OK, and UNZ_PARAMERROR.

```
849 {
850     unz64_s* s;
851     if (file==NULL)
852         return UNZ_PARAMERROR;
853     s=(unz64_s*)file;
854     /* to do : check if number_entry is not truncated */
855     pglobal_info32->number_entry = (uLong)s->gi.number_entry;
856     pglobal_info32->size_comment = s->gi.size_comment;
857     return UNZ_OK;
858 }
```

13.107.4.27 int ZEXPORT unzGetGlobalInfo64 (unzFile *file*, unz_global_info64 * *pglobal_info*)

Definition at line 838 of file unzip.c.

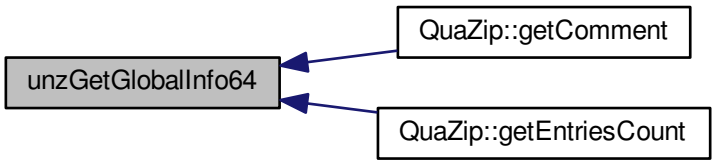
References unz64_s::gi, UNZ_OK, and UNZ_PARAMERROR.

Referenced by QuaZip::getComment(), and QuaZip::getEntriesCount().

```
839 {
840     unz64_s* s;
841     if (file==NULL)
```

```
842         return UNZ_PARAMERROR;
843     s=(unz64_s*)file;
844     *pglobal_info=s->gi;
845     return UNZ_OK;
846 }
```

Here is the caller graph for this function:



13.107.4.28 int ZEXPORT unzGetLocalExtrafield (unzFile *file*, voidp *buf*, unsigned *len*)

Definition at line 1964 of file unzip.c.

References file_in_zip64_read_info_s::filestream, file_in_zip64_read_info_s::offset_local_extrafield, unz64_s::pfile_in_zip_read, file_in_zip64_read_info_s::pos_local_extrafield, file_in_zip64_read_info_s::size_local_extrafield, UNZ_ERRNO, UNZ_PARAMERROR, file_in_zip64_read_info_s::z_filefunc, ZLIB_FILEFUNC_SEEK_SET, ZREAD64, and ZSEEK64.

```
1965 {
1966     unz64_s* s;
1967     file_in_zip64_read_info_s* pfile_in_zip_read_info;
1968     uInt read_now;
1969     ZPOS64_T size_to_read;
1970
1971     if (file==NULL)
1972         return UNZ_PARAMERROR;
1973     s=(unz64_s*)file;
1974     pfile_in_zip_read_info=s->pfile_in_zip_read;
1975
1976     if (pfile_in_zip_read_info==NULL)
1977         return UNZ_PARAMERROR;
1978
1979     size_to_read = (pfile_in_zip_read_info->size_local_extrafield -
1980                 pfile_in_zip_read_info->pos_local_extrafield);
1981
1982     if (buf==NULL)
1983         return (int)size_to_read;
1984
1985     if (len>size_to_read)
1986         read_now = (uInt)size_to_read;
1987     else
1988         read_now = (uInt)len ;
1989
1990     if (read_now==0)
1991         return 0;
1992
1993     if (ZSEEK64(pfile_in_zip_read_info->z_filefunc,
1994               pfile_in_zip_read_info->filestream,
1995               pfile_in_zip_read_info->offset_local_extrafield +
1996               pfile_in_zip_read_info->pos_local_extrafield,
1997               ZLIB_FILEFUNC_SEEK_SET)!=0)
1998         return UNZ_ERRNO;
1999
2000     if (ZREAD64(pfile_in_zip_read_info->z_filefunc,
2001               pfile_in_zip_read_info->filestream,
2002               buf,read_now)!=read_now)
2003         return UNZ_ERRNO;
2004
2005     return (int)read_now;
2006 }
```

13.107.4.29 uLong ZEXPORT unzGetOffset (unzFile *file*)

Definition at line 2102 of file unzip.c.

References unzGetOffset64().

```
2103 {
2104     ZPOS64_T offset64;
2105
2106     if (file==NULL)
2107           return 0; /*UNZ_PARAMERROR; */
2108     offset64 = unzGetOffset64(file);
2109     return (uLong)offset64;
2110 }
```

Here is the call graph for this function:

13.107.4.30 ZPOS64_T ZEXPORT unzGetOffset64 (unzFile *file*)

Definition at line 2087 of file unzip.c.

References unz64_s::current_file_ok, unz64_s::gi, unz64_s::num_file, unz_global_info64_s::number_entry, and unz64_s::pos_in_central_dir.

Referenced by unzGetOffset().

```
2088 {
2089     unz64_s* s;
2090
2091     if (file==NULL)
2092           return 0; /*UNZ_PARAMERROR; */
2093     s=(unz64_s*)file;
2094     if (!s->current_file_ok)
2095       return 0;
2096     if (s->gi.number_entry != 0 && s->gi.number_entry != 0xffff)
2097       if (s->num_file==s->gi.number_entry)
2098          return 0;
2099     return s->pos_in_central_dir;
2100 }
```

Here is the caller graph for this function:

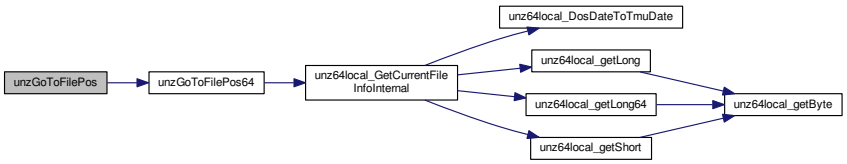
13.107.4.31 int ZEXPORT unzGoToFilePos (unzFile *file*, unz_file_pos * *file_pos*)

Definition at line 1370 of file unzip.c.

References unz_file_pos_s::num_of_file, unz64_file_pos_s::num_of_file, unz_file_pos_s::pos_in_zip_directory, unz64_file_pos_s::pos_in_zip_directory, UNZ_PARAMERROR, and unzGoToFilePos64().

```
1373 {
1374     unz64_file_pos file_pos64;
1375     if (file_pos == NULL)
1376         return UNZ_PARAMERROR;
1377
1378     file_pos64.pos_in_zip_directory = file_pos->
     pos_in_zip_directory;
1379     file_pos64.num_of_file = file_pos->num_of_file;
1380     return unzGoToFilePos64(file,&file_pos64);
1381 }
```

Here is the call graph for this function:



13.107.4.32 int ZEXPORT unzGoToFilePos64 (unzFile *file*, const unz64_file_pos * *file_pos*)

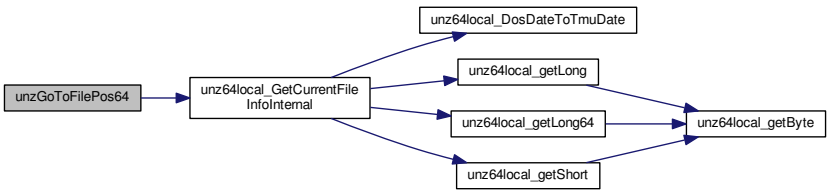
Definition at line 1348 of file unzip.c.

References unz64_s::cur_file_info, unz64_s::cur_file_info_internal, unz64_s::current_file_ok, unz64_s::num_file, unz64_file_pos_s::num_of_file, unz64_s::pos_in_central_dir, unz64_file_pos_s::pos_in_zip_directory, unz64local_GetCurrentFileInfoInternal(), UNZ_OK, and UNZ_PARAMERROR.

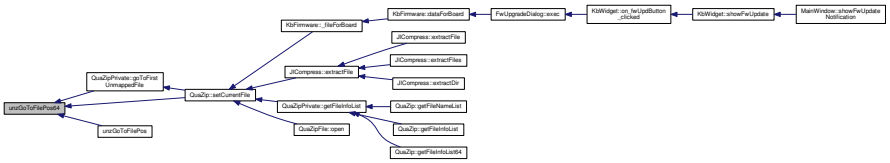
Referenced by QuaZipPrivate::goToFirstUnmappedFile(), QuaZip::setCurrentFile(), and unzGoToFilePos().

```
1349 {
1350     unz64_s* s;
1351     int err;
1352
1353     if (file==NULL || file_pos==NULL)
1354         return UNZ_PARAMERROR;
1355     s=(unz64_s*)file;
1356
1357     /* jump to the right spot */
1358     s->pos_in_central_dir = file_pos->pos_in_zip_directory;
1359     s->num_file           = file_pos->num_of_file;
1360
1361     /* set the current file */
1362     err = unz64local_GetCurrentFileInfoInternal(file,&s->
     cur_file_info,
1363                                                 &s->cur_file_info_internal,
1364                                                 NULL,0,NULL,0,NULL,0);
1365     /* return results */
1366     s->current_file_ok = (err == UNZ_OK);
1367     return err;
1368 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



13.107.4.33 int ZEXPORT unzGoToFirstFile (unzFile *file*)

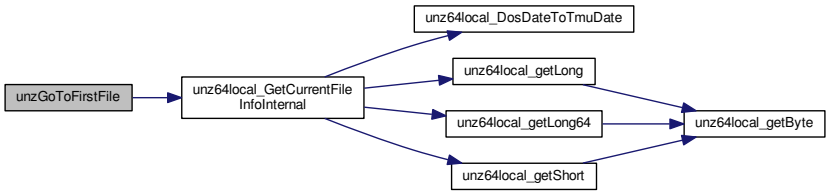
Definition at line 1188 of file unzip.c.

References unz64_s::cur_file_info, unz64_s::cur_file_info_internal, unz64_s::current_file_ok, unz64_s::num_file, unz64_s::offset_central_dir, unz64_s::pos_in_central_dir, unz64local_GetCurrentFileInfoInternal(), UNZ_OK, and UNZ_PARAMERROR.

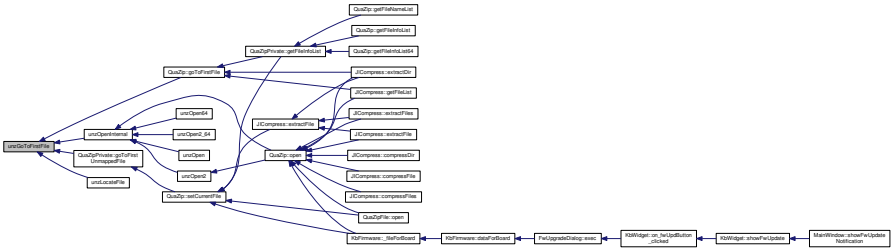
Referenced by QuaZip::goToFirstFile(), QuaZipPrivate::goToFirstUnmappedFile(), unzLocateFile(), and unzOpenInternal().

```
1189 {
1190     int err=UNZ_OK;
1191     unz64_s* s;
1192     if (file==NULL)
1193         return UNZ_PARAMERROR;
1194     s=(unz64_s*)file;
1195     s->pos_in_central_dir=s->offset_central_dir;
1196     s->num_file=0;
1197     err=unz64local_GetCurrentFileInfoInternal(file,&s->
      cur_file_info,
1198                                               &s->cur_file_info_internal,
1199                                               NULL,0,NULL,0,NULL,0);
1200     s->current_file_ok = (err == UNZ_OK);
1201     return err;
1202 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



13.107.4.34 int ZEXPORT unzGoToNextFile (unzFile *file*)

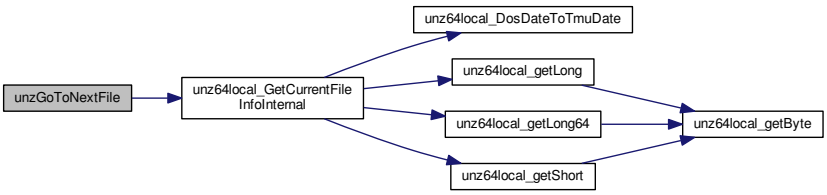
Definition at line 1209 of file unzip.c.

References unz64_s::cur_file_info, unz64_s::cur_file_info_internal, unz64_s::current_file_ok, unz64_s::gi, unz64_s::num_file, unz_global_info64_s::number_entry, unz64_s::pos_in_central_dir, unz_file_info64_s::size_file_comment, unz_file_info64_s::size_file_extra, unz_file_info64_s::size_filename, SIZECENTRALDIRITEM, unz64local_GetCurrentFileInfoInternal(), UNZ_END_OF_LIST_OF_FILE, UNZ_OK, and UNZ_PARAMERROR.

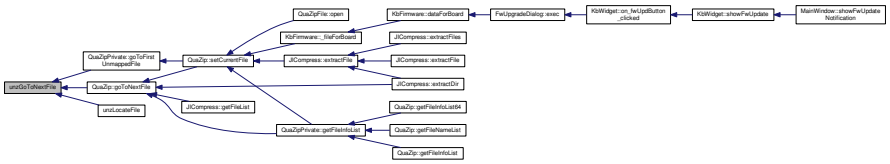
Referenced by QuaZipPrivate::goToFirstUnmappedFile(), QuaZip::goToNextFile(), and unzLocateFile().

```
1210 {
1211     unz64_s* s;
1212     int err;
1213
1214     if (file==NULL)
1215         return UNZ_PARAMERROR;
1216     s=(unz64_s*)file;
1217     if (!s->current_file_ok)
1218         return UNZ_END_OF_LIST_OF_FILE;
1219     if (s->gi.number_entry != 0xffff)    /* 2^16 files overflow hack */
1220       if (s->num_file+1==s->gi.number_entry)
1221         return UNZ_END_OF_LIST_OF_FILE;
1222
1223     s->pos_in_central_dir += SIZECENTRALDIRITEM + s->
     cur_file_info.size_filename +
1224             s->cur_file_info.size_file_extra + s->
     cur_file_info.size_file_comment ;
1225     s->num_file++;
1226     err = unz64local_GetCurrentFileInfoInternal(file,&s->
     cur_file_info,
1227                                                 &s->cur_file_info_internal,
1228                                                 NULL,0,NULL,0,NULL,0);
1229     s->current_file_ok = (err == UNZ_OK);
1230     return err;
1231 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



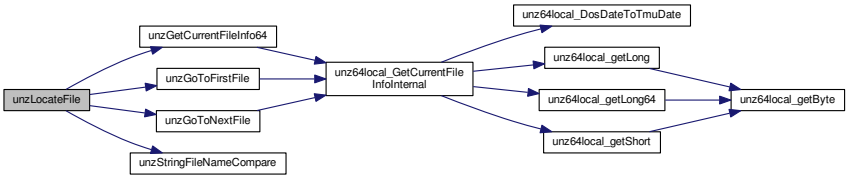
13.107.4.35 int ZEXPORT unzLocateFile (unzFile *file*, const char * *szFileName*, int *iCaseSensitivity*)

Definition at line 1242 of file unzip.c.

References unz64_s::cur_file_info, unz64_s::cur_file_info_internal, unz64_s::current_file_ok, unz64_s::num_file, unz64_s::pos_in_central_dir, UNZ_END_OF_LIST_OF_FILE, UNZ_MAXFILENAMEINZIP, UNZ_OK, UNZ_PARAMERROR, unzGetCurrentFileInfo64(), unzGoToFirstFile(), unzGoToNextFile(), and unzStringFileNameCompare().

```
1243 {
1244     unz64_s* s;
1245     int err;
1246
1247     /* We remember the 'current' position in the file so that we can jump
1248      * back there if we fail.
1249      */
1250     unz_file_info64 cur_file_infoSaved;
1251     unz_file_info64_internal cur_file_info_internalSaved;
1252     ZPOS64_T num_fileSaved;
1253     ZPOS64_T pos_in_central_dirSaved;
1254
1255
1256     if (file==NULL)
1257         return UNZ_PARAMERROR;
1258
1259     if (strlen(szFileName)>=UNZ_MAXFILENAMEINZIP)
1260         return UNZ_PARAMERROR;
1261
1262     s=(unz64_s*)file;
1263     if (!s->current_file_ok)
1264         return UNZ_END_OF_LIST_OF_FILE;
1265
1266     /* Save the current state */
1267     num_fileSaved = s->num_file;
1268     pos_in_central_dirSaved = s->pos_in_central_dir;
1269     cur_file_infoSaved = s->cur_file_info;
1270     cur_file_info_internalSaved = s->cur_file_info_internal;
1271
1272     err = unzGoToFirstFile(file);
1273
1274     while (err == UNZ_OK)
1275     {
1276         char szCurrentFileName[UNZ_MAXFILENAMEINZIP+1];
1277         err = unzGetCurrentFileInfo64(file,NULL,
1278                                       szCurrentFileName,sizeof(szCurrentFileName)-1,
1279                                       NULL,0,NULL,0);
1280         if (err == UNZ_OK)
1281         {
1282             if (unzStringFileNameCompare(szCurrentFileName,
1283                                          szFileName,iCaseSensitivity)==0)
1284                 return UNZ_OK;
1285             err = unzGoToNextFile(file);
1286         }
1287     }
1288
1289     /* We failed, so restore the state of the 'current file' to where we
1290      * were.
1291      */
1292     s->num_file = num_fileSaved ;
1293     s->pos_in_central_dir = pos_in_central_dirSaved ;
1294     s->cur_file_info = cur_file_infoSaved;
1295     s->cur_file_info_internal = cur_file_info_internalSaved;
1296     return err;
1297 }
```

Here is the call graph for this function:



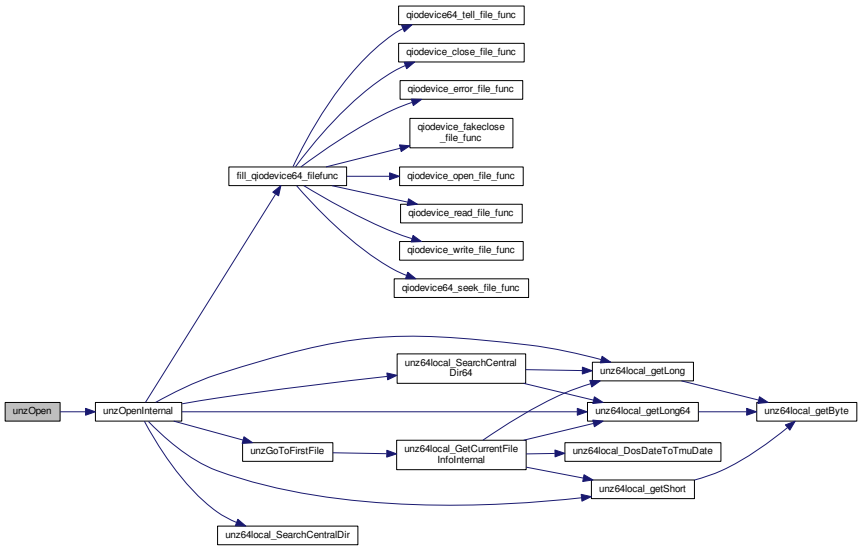
13.107.4.36 unzFile ZEXPORT unzOpen (voidpf *file*)

Definition at line 800 of file unzip.c.

References UNZ_DEFAULT_FLAGS, and unzOpenInternal().

```
801 {
802     return unzOpenInternal(file, NULL, 0, UNZ_DEFAULT_FLAGS);
803 }
```

Here is the call graph for this function:



13.107.4.37 unzFile ZEXPORT unzOpen2 (voidpf *file*, zlib_filefunc_def * *pzlib_filefunc32_def*)

Definition at line 772 of file unzip.c.

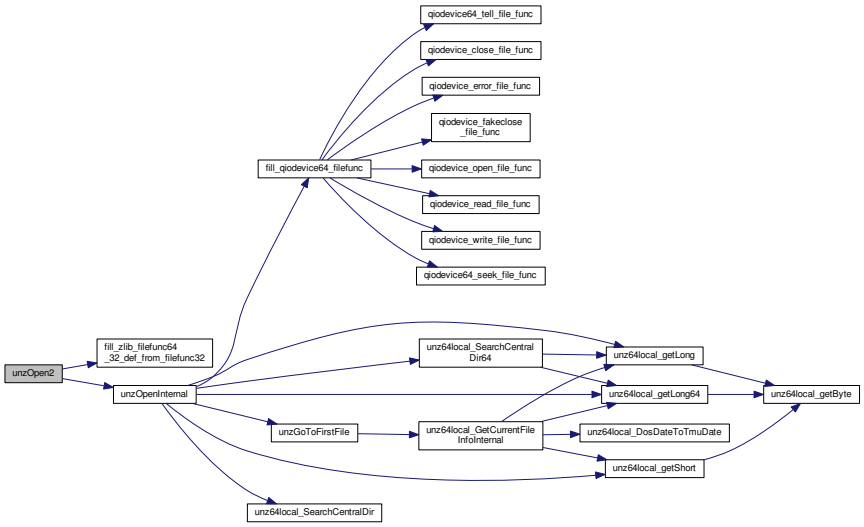
References fill_zlib_filefunc64_32_def_from_filefunc32(), UNZ_DEFAULT_FLAGS, and unzOpenInternal().

Referenced by QuaZip::open().

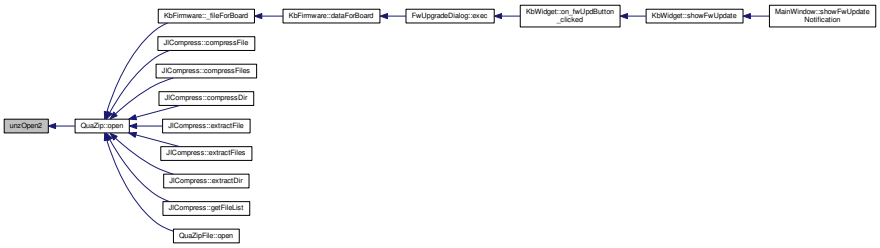
```
774 {
775     if (pzlib_filefunc32_def != NULL)
776     {
777         zlib_filefunc64_32_def zlib_filefunc64_32_def_fill;
778         fill_zlib_filefunc64_32_def_from_filefunc32(&
```

```
    zlib_filefunc64_32_def_fill,pzlib_filefunc32_def);
779         return unzOpenInternal(file, &zlib_filefunc64_32_def_fill, 0,
    UNZ_DEFAULT_FLAGS);
780     }
781     else
782         return unzOpenInternal(file, NULL, 0, UNZ_DEFAULT_FLAGS);
783 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



13.107.4.38 unzFile ZEXPORT unzOpen2_64 (voidpf *file*, zlib_filefunc64_def * *pzlib_filefunc_def*)

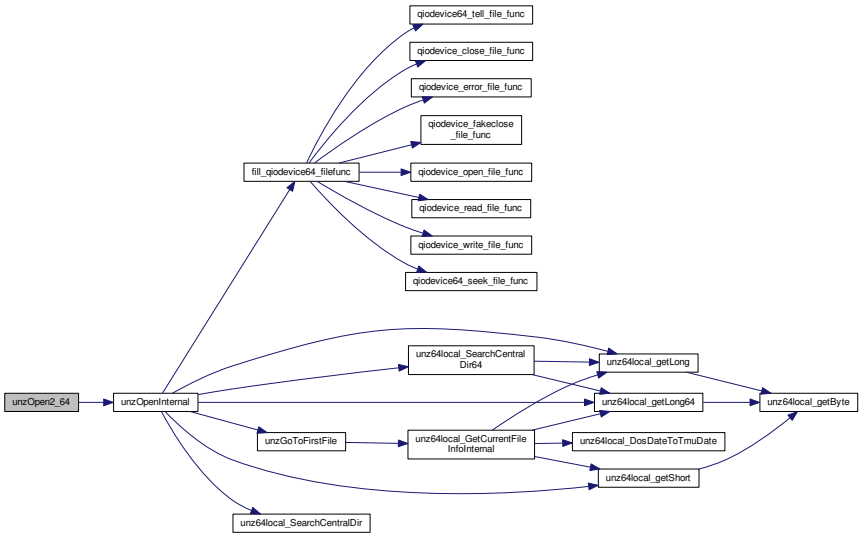
Definition at line 785 of file unzip.c.

References UNZ_DEFAULT_FLAGS, unzOpenInternal(), zlib_filefunc64_32_def_s::zfile_func64, zlib_filefunc64_-32_def_s::zseek32_file, and zlib_filefunc64_32_def_s::ztell32_file.

```
787 {
788     if (pzlib_filefunc_def != NULL)
789     {
790         zlib_filefunc64_32_def zlib_filefunc64_32_def_fill;
791         zlib_filefunc64_32_def_fill.zfile_func64 = *pzlib_filefunc_def;
792         zlib_filefunc64_32_def_fill.ztell32_file = NULL;
793         zlib_filefunc64_32_def_fill.zseek32_file = NULL;
794         return unzOpenInternal(file, &zlib_filefunc64_32_def_fill, 1,
    UNZ_DEFAULT_FLAGS);
795     }
796     else
```

```
797         return unzOpenInternal(file, NULL, 1, UNZ_DEFAULT_FLAGS);
798 }
```

Here is the call graph for this function:



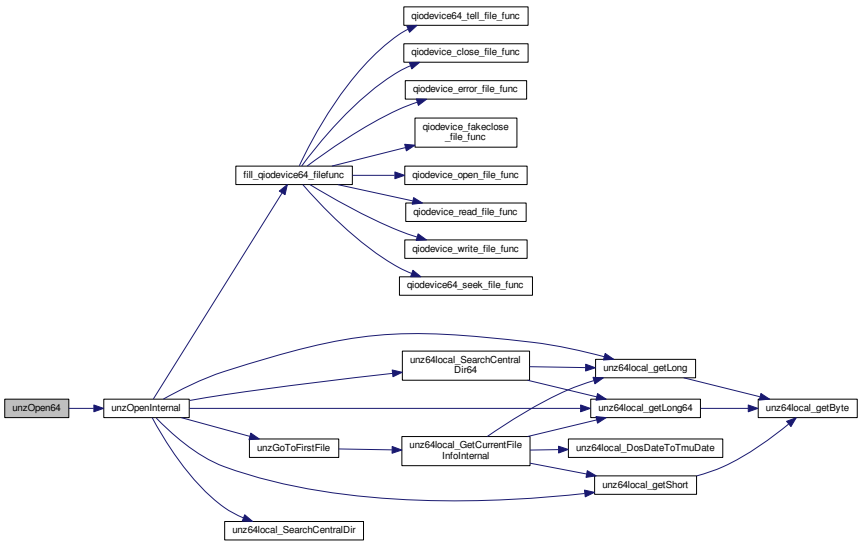
13.107.4.39 unzFile ZEXPORT unzOpen64 (voidpf *file*)

Definition at line 805 of file unzip.c.

References UNZ_DEFAULT_FLAGS, and unzOpenInternal().

```
806 {
807     return unzOpenInternal(file, NULL, 1, UNZ_DEFAULT_FLAGS);
808 }
```

Here is the call graph for this function:



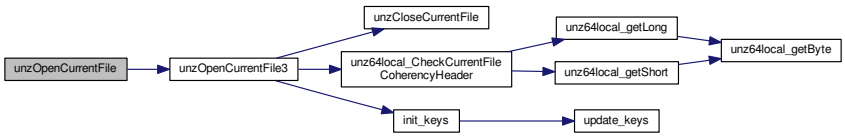
13.107.4.40 int ZEXPORT unzOpenCurrentFile (unzFile *file*)

Definition at line 1649 of file unzip.c.

References unzOpenCurrentFile3().

```
1650 {
1651     return unzOpenCurrentFile3(file, NULL, NULL, 0, NULL);
1652 }
```

Here is the call graph for this function:



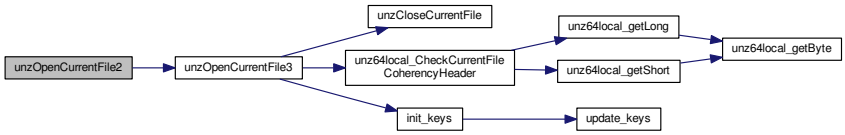
13.107.4.41 int ZEXPORT unzOpenCurrentFile2 (unzFile *file*, int * *method*, int * *level*, int *raw*)

Definition at line 1659 of file unzip.c.

References unzOpenCurrentFile3().

```
1660 {
1661     return unzOpenCurrentFile3(file, method, level, raw, NULL);
1662 }
```

Here is the call graph for this function:



13.107.4.42 int ZEXPORT unzOpenCurrentFile3 (unzFile *file*, int * *method*, int * *level*, int *raw*, const char * *password*)

Definition at line 1480 of file unzip.c.

References ALLOC, file_in_zip64_read_info_s::byte_before_the_zipfile, unz64_s::byte_before_the_zipfile, unz_file_info64_s::compressed_size, unz_file_info64_s::compression_method, file_in_zip64_read_info_s::compression_method, unz_file_info64_s::crc, file_in_zip64_read_info_s::crc32, file_in_zip64_read_info_s::crc32_wait, unz64_s::cur_file_info, unz64_s::cur_file_info_internal, unz64_s::current_file_ok, unz64_s::encrypted, file_in_zip64_read_info_s::filestream, unz64_s::filestream, unz_file_info64_s::flag, init_keys(), unz64_s::keys, unz_file_info64_internal_s::offset_curfile, file_in_zip64_read_info_s::offset_local_extrafield, unz64_s::pcrc_32_tab, unz64_s::pfile_in_zip_read, file_in_zip64_read_info_s::pos_in_zipfile, file_in_zip64_read_info_s::pos_local_extrafield, file_in_zip64_read_info_s::raw, file_in_zip64_read_info_s::read_buffer, file_in_zip64_read_info_s::rest_read_compressed, file_in_zip64_read_info_s::rest_read_uncompressed, SEEK_SET, file_in_zip64_read_info_s::size_local_extrafield,

SIZEZIPLOCALHEADER, file_in_zip64_read_info_s::stream, file_in_zip64_read_info_s::stream_initialised, file_in_zip64_read_info_s::total_out_64, TRYFREE, unz_file_info64_s::uncompressed_size, unz64local_CheckCurrentFileCoherencyHeader(), UNZ_BADZIPFILE, UNZ_BUFSIZE, UNZ_INTERNALERROR, UNZ_OK, UNZ_PARAMERROR, unzCloseCurrentFile(), Z_BZIP2ED, file_in_zip64_read_info_s::z_filefunc, unz64_s::z_filefunc, zdecode, ZREAD64, and ZSEEK64.

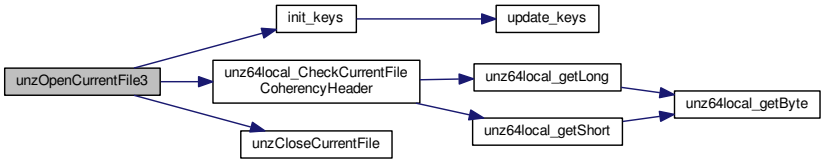
Referenced by QuaZipFile::open(), unzOpenCurrentFile(), unzOpenCurrentFile2(), and unzOpenCurrentFilePassword().

```
1482 {
1483     int err=UNZ_OK;
1484     uInt iSizeVar;
1485     unz64_s* s;
1486     file_in_zip64_read_info_s* pfile_in_zip_read_info;
1487     ZPOS64_T offset_local_extrafield;  /* offset of the local extra field */
1488     uInt  size_local_extrafield;    /* size of the local extra field */
1489 #    ifndef NOUNCRYPT
1490     char source[12];
1491 #    else
1492     if (password != NULL)
1493         return UNZ_PARAMERROR;
1494 #    endif
1495
1496     if (file==NULL)
1497         return UNZ_PARAMERROR;
1498     s=(unz64_s*)file;
1499     if (!s->current_file_ok)
1500         return UNZ_PARAMERROR;
1501
1502     if (s->pfile_in_zip_read != NULL)
1503         unzCloseCurrentFile(file);
1504
1505     if (unz64local_CheckCurrentFileCoherencyHeader(s,&iSizeVar, &
     offset_local_extrafield,&size_local_extrafield)!=UNZ_OK)
1506         return UNZ_BADZIPFILE;
1507
1508     pfile_in_zip_read_info = (file_in_zip64_read_info_s*)
     ALLOC(sizeof(file_in_zip64_read_info_s));
1509     if (pfile_in_zip_read_info==NULL)
1510         return UNZ_INTERNALERROR;
1511
1512     pfile_in_zip_read_info->read_buffer=(char*)ALLOC(UNZ_BUFSIZE);
1513     pfile_in_zip_read_info->offset_local_extrafield = offset_local_extrafield;
1514     pfile_in_zip_read_info->size_local_extrafield = size_local_extrafield;
1515     pfile_in_zip_read_info->pos_local_extrafield=0;
1516     pfile_in_zip_read_info->raw=raw;
1517
1518     if (pfile_in_zip_read_info->read_buffer==NULL)
1519     {
1520         TRYFREE(pfile_in_zip_read_info);
1521         return UNZ_INTERNALERROR;
1522     }
1523
1524     pfile_in_zip_read_info->stream_initialised=0;
1525
1526     if (method!=NULL)
1527         *method = (int)s->cur_file_info.compression_method;
1528
1529     if (level!=NULL)
1530     {
1531         *level = 6;
1532         switch (s->cur_file_info.flag & 0x06)
1533         {
1534           case 6 : *level = 1; break;
1535           case 4 : *level = 2; break;
1536           case 2 : *level = 9; break;
1537         }
1538     }
1539
1540     if ((s->cur_file_info.compression_method!=0) &&
1541 /* #ifdef HAVE_BZIP2 */
1542         (s->cur_file_info.compression_method!=
     Z_BZIP2ED) &&
1543 /* #endif */
1544         (s->cur_file_info.compression_method!=Z_DEFLATED))
1545
1546         err=UNZ_BADZIPFILE;
1547
1548     pfile_in_zip_read_info->crc32_wait=s->cur_file_info.
     crc;
1549     pfile_in_zip_read_info->crc32=0;
1550     pfile_in_zip_read_info->total_out_64=0;
1551     pfile_in_zip_read_info->compression_method = s->
     cur_file_info.compression_method;
```

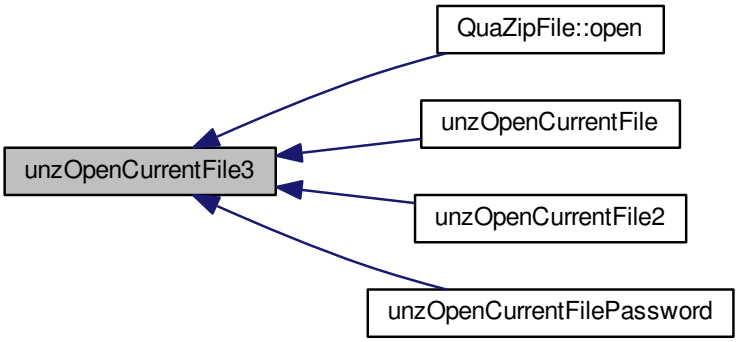
```
1552     pfile_in_zip_read_info->filestream=s->filestream;
1553     pfile_in_zip_read_info->z_filefunc=s->z_filefunc;
1554     pfile_in_zip_read_info->byte_before_the_zipfile=s->
     byte_before_the_zipfile;
1555
1556     pfile_in_zip_read_info->stream.total_out = 0;
1557
1558     if ((s->cur_file_info.compression_method==
     Z_BZIP2ED) && (!raw))
1559     {
1560 #ifdef HAVE_BZIP2
1561       pfile_in_zip_read_info->bstream.bzalloc = (void *(*) (void *, int, int))0;
1562       pfile_in_zip_read_info->bstream.bzfree = (free_func)0;
1563       pfile_in_zip_read_info->bstream.opaque = (voidpf)0;
1564       pfile_in_zip_read_info->bstream.state = (voidpf)0;
1565
1566       pfile_in_zip_read_info->stream.zalloc = (alloc_func)0;
1567       pfile_in_zip_read_info->stream.zfree = (free_func)0;
1568       pfile_in_zip_read_info->stream.opaque = (voidpf)0;
1569       pfile_in_zip_read_info->stream.next_in = (voidpf)0;
1570       pfile_in_zip_read_info->stream.avail_in = 0;
1571
1572       err=BZ2_bzDecompressInit(&pfile_in_zip_read_info->bstream, 0, 0);
1573       if (err == Z_OK)
1574         pfile_in_zip_read_info->stream_initialised=Z_BZIP2ED;
1575       else
1576       {
1577         TRYFREE(pfile_in_zip_read_info);
1578         return err;
1579       }
1580 #else
1581       pfile_in_zip_read_info->raw=1;
1582 #endif
1583     }
1584     else if ((s->cur_file_info.compression_method==Z_DEFLATED) && (!raw))
1585     {
1586       pfile_in_zip_read_info->stream.zalloc = (alloc_func)0;
1587       pfile_in_zip_read_info->stream.zfree = (free_func)0;
1588       pfile_in_zip_read_info->stream.opaque = (voidpf)0;
1589       pfile_in_zip_read_info->stream.next_in = 0;
1590       pfile_in_zip_read_info->stream.avail_in = 0;
1591
1592       err=inflateInit2(&pfile_in_zip_read_info->stream, -MAX_WBITS);
1593       if (err == Z_OK)
1594         pfile_in_zip_read_info->stream_initialised=Z_DEFLATED;
1595       else
1596       {
1597         TRYFREE(pfile_in_zip_read_info);
1598         return err;
1599       }
1600         /* windowBits is passed < 0 to tell that there is no zlib header.
1601          * Note that in this case inflate *requires* an extra "dummy" byte
1602          * after the compressed stream in order to complete decompression and
1603          * return Z_STREAM_END.
1604          * In unzip, i don't wait absolutely Z_STREAM_END because I known the
1605          * size of both compressed and uncompressed data
1606          */
1607     }
1608     pfile_in_zip_read_info->rest_read_compressed =
1609             s->cur_file_info.compressed_size ;
1610     pfile_in_zip_read_info->rest_read_uncompressed =
1611             s->cur_file_info.uncompressed_size ;
1612
1613
1614     pfile_in_zip_read_info->pos_in_zipfile =
1615             s->cur_file_info_internal.offset_curfile +
     SIZEZIPLOCALHEADER +
1616               iSizeVar;
1617
1618     pfile_in_zip_read_info->stream.avail_in = (uInt)0;
1619
1620     s->pfile_in_zip_read = pfile_in_zip_read_info;
1621                 s->encrypted = 0;
1622
1623 #    ifndef NOUNCRYPT
1624     if (password != NULL)
1625     {
1626         int i;
1627         s->pcrc_32_tab = get_crc_table();
1628         init_keys(password,s->keys,s->pcrc_32_tab);
1629         if (ZSEEK64(s->z_filefunc, s->filestream,
1630                   s->pfile_in_zip_read->pos_in_zipfile +
1631                      s->pfile_in_zip_read->
     byte_before_the_zipfile,
1632                   SEEK_SET)!=0)
1633             return UNZ_INTERNALERROR;
1634         if(ZREAD64(s->z_filefunc, s->filestream,source, 12)<12)
```

```
1635             return UNZ_INTERNALERROR;
1636
1637         for (i = 0; i<12; i++)
1638             zdecode(s->keys,s->pcrc_32_tab,source[i]);
1639
1640         s->pfile_in_zip_read->pos_in_zipfile+=12;
1641         s->encrypted=1;
1642     }
1643 #    endif
1644
1645
1646     return UNZ_OK;
1647 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



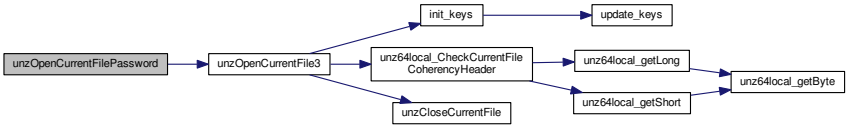
13.107.4.43 int ZEXPORT unzOpenCurrentFilePassword (unzFile *file*, const char * *password*)

Definition at line 1654 of file unzip.c.

References unzOpenCurrentFile3().

```
1655 {
1656     return unzOpenCurrentFile3(file, NULL, NULL, 0, password);
1657 }
```

Here is the call graph for this function:



13.107.4.44 unzFile unzOpenInternal (voidpf *file*, zlib_filefunc64_32_def * *pzlib_filefunc64_32_def*, int *is64bitOpenFunction*, unsigned *flags*)

Definition at line 587 of file unzip.c.

References ALLOC, unz64_s::byte_before_the_zipfile, unz64_s::central_pos, unz64_s::encrypted, unz64_s::filestream, fill_qiodevice64_filefunc(), unz64_s::flags, unz64_s::gi, unz64_s::is64bitOpenFunction, unz64_s::isZip64, unz_global_info64_s::number_entry, unz64_s::offset_central_dir, unz64_s::pfile_in_zip_read, unz64_s::size_central_dir, unz_global_info64_s::size_comment, unz64local_getLong(), unz64local_getLong64(), unz64local_getShort(), unz64local_SearchCentralDir(), unz64local_SearchCentralDir64(), UNZ_AUTO_CLOSE, UNZ_BADZIPFILE, unz_copyright, UNZ_ERRNO, UNZ_OK, unzGoToFirstFile(), unz64_s::z_filefunc, ZCLOSE64, ZFAKECLOSE64, zlib_filefunc64_32_def_s::zfile_func64, ZLIB_FILEFUNC_MODE_EXISTING, ZLIB_FILEFUNC_MODE_READ, ZLIB_FILEFUNC_SEEK_SET, ZOPEN64, zlib_filefunc64_32_def_s::zseek32_file, ZSEEK64, and zlib_filefunc64_32_def_s::ztell32_file.

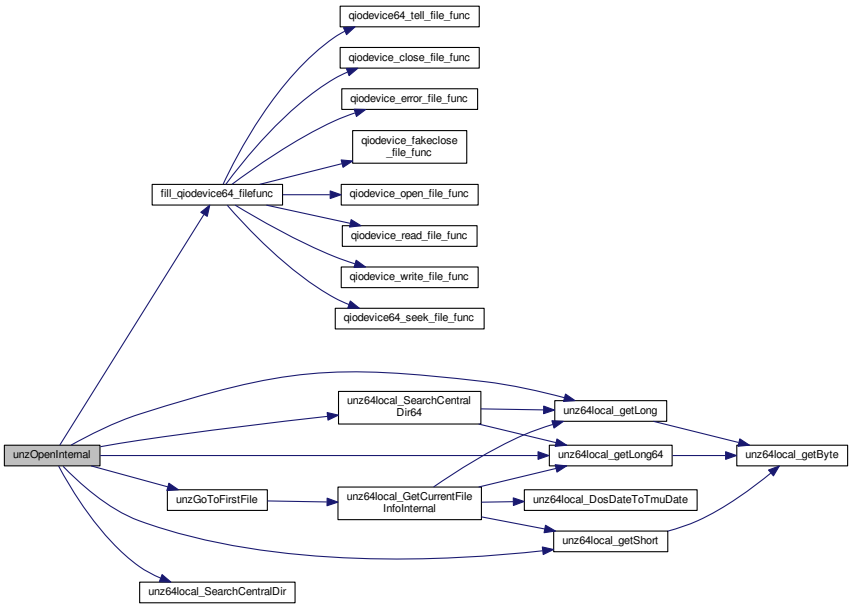
Referenced by QuaZip::open(), unzOpen(), unzOpen2(), unzOpen2_64(), and unzOpen64().

```
590 {
591     unz64_s us;
592     unz64_s *s;
593     ZPOS64_T central_pos;
594     uLong   uL;
595 
596     uLong number_disk;          /* number of the current dist, used for
597                                    spaning ZIP, unsupported, always 0*/
598     uLong number_disk_with_CD;  /* number the the disk with central dir, used
599                                    for spaning ZIP, unsupported, always 0*/
600     ZPOS64_T number_entry_CD;       /* total number of entries in
601                                    the central dir
602                                    (same than number_entry on nospan) */
603 
604     int err=UNZ_OK;
605 
606     if (unz_copyright[0]!=' ')
607         return NULL;
608 
609     us.flags = flags;
610     us.z_filefunc.zseek32_file = NULL;
611     us.z_filefunc.ztell32_file = NULL;
612     if (pzlib_filefunc64_32_def==NULL)
613         fill_qiodevice64_filefunc(&us.z_filefunc.
    zfile_func64);
614     else
615         us.z_filefunc = *pzlib_filefunc64_32_def;
616     us.is64bitOpenFunction = is64bitOpenFunction;
617 
618 
619 
620     us.filestream = ZOPEN64(us.z_filefunc,
621                                                 file,
622                                                 ZLIB_FILEFUNC_MODE_READ |
623                                                 ZLIB_FILEFUNC_MODE_EXISTING);
624     if (us.filestream==NULL)
625         return NULL;
626 
627     central_pos = unz64local_SearchCentralDir64(&us.
    z_filefunc,us.filestream);
628     if (central_pos)
629     {
630         uLong uS;
631         ZPOS64_T uL64;
632 
```

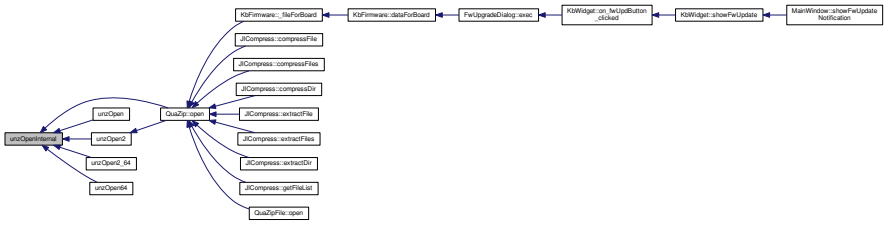
```
633         us.isZip64 = 1;
634
635         if (ZSEEK64(us.z_filefunc, us.filestream,
636                                         central_pos,ZLIB_FILEFUNC_SEEK_SET)!=0)
637         err=UNZ_ERRNO;
638
639         /* the signature, already checked */
640         if (unz64local_getLong(&us.z_filefunc, us.
    filestream,&uL)!=UNZ_OK)
641             err=UNZ_ERRNO;
642
643         /* size of zip64 end of central directory record */
644         if (unz64local_getLong64(&us.z_filefunc, us.
    filestream,&uL64)!=UNZ_OK)
645             err=UNZ_ERRNO;
646
647         /* version made by */
648         if (unz64local_getShort(&us.z_filefunc, us.
    filestream,&uS)!=UNZ_OK)
649             err=UNZ_ERRNO;
650
651         /* version needed to extract */
652         if (unz64local_getShort(&us.z_filefunc, us.
    filestream,&uS)!=UNZ_OK)
653             err=UNZ_ERRNO;
654
655         /* number of this disk */
656         if (unz64local_getLong(&us.z_filefunc, us.
    filestream,&number_disk)!=UNZ_OK)
657             err=UNZ_ERRNO;
658
659         /* number of the disk with the start of the central directory */
660         if (unz64local_getLong(&us.z_filefunc, us.
    filestream,&number_disk_with_CD)!=UNZ_OK)
661             err=UNZ_ERRNO;
662
663         /* total number of entries in the central directory on this disk */
664         if (unz64local_getLong64(&us.z_filefunc, us.
    filestream,&us.gi.number_entry)!=UNZ_OK)
665             err=UNZ_ERRNO;
666
667         /* total number of entries in the central directory */
668         if (unz64local_getLong64(&us.z_filefunc, us.
    filestream,&number_entry_CD)!=UNZ_OK)
669             err=UNZ_ERRNO;
670
671         if ((number_entry_CD!=us.gi.number_entry) ||
672             (number_disk_with_CD!=0) ||
673             (number_disk!=0))
674             err=UNZ_BADZIPFILE;
675
676         /* size of the central directory */
677         if (unz64local_getLong64(&us.z_filefunc, us.
    filestream,&us.size_central_dir)!=UNZ_OK)
678             err=UNZ_ERRNO;
679
680         /* offset of start of central directory with respect to the
681           starting disk number */
682         if (unz64local_getLong64(&us.z_filefunc, us.
    filestream,&us.offset_central_dir)!=UNZ_OK)
683             err=UNZ_ERRNO;
684
685         us.gi.size_comment = 0;
686       }
687       else
688       {
689         central_pos = unz64local_SearchCentralDir(&us.
    z_filefunc,us.filestream);
690         if (central_pos==0)
691             err=UNZ_ERRNO;
692
693         us.isZip64 = 0;
694
695         if (ZSEEK64(us.z_filefunc, us.filestream,
696                                         central_pos,ZLIB_FILEFUNC_SEEK_SET)!=0)
697             err=UNZ_ERRNO;
698
699         /* the signature, already checked */
700         if (unz64local_getLong(&us.z_filefunc, us.
    filestream,&uL)!=UNZ_OK)
701             err=UNZ_ERRNO;
702
703         /* number of this disk */
704         if (unz64local_getShort(&us.z_filefunc, us.
    filestream,&number_disk)!=UNZ_OK)
705             err=UNZ_ERRNO;
706
```

```
707         /* number of the disk with the start of the central directory */
708         if (unz64local_getShort(&us.z_filefunc, us.
    filestream,&number_disk_with_CD)!=UNZ_OK)
709             err=UNZ_ERRNO;
710
711         /* total number of entries in the central dir on this disk */
712         if (unz64local_getShort(&us.z_filefunc, us.
    filestream,&uL)!=UNZ_OK)
713             err=UNZ_ERRNO;
714         us.gi.number_entry = uL;
715
716         /* total number of entries in the central dir */
717         if (unz64local_getShort(&us.z_filefunc, us.
    filestream,&uL)!=UNZ_OK)
718             err=UNZ_ERRNO;
719         number_entry_CD = uL;
720
721         if ((number_entry_CD!=us.gi.number_entry) ||
722             (number_disk_with_CD!=0) ||
723             (number_disk!=0))
724             err=UNZ_BADZIPFILE;
725
726         /* size of the central directory */
727         if (unz64local_getLong(&us.z_filefunc, us.
    filestream,&uL)!=UNZ_OK)
728             err=UNZ_ERRNO;
729         us.size_central_dir = uL;
730
731         /* offset of start of central directory with respect to the
732             starting disk number */
733         if (unz64local_getLong(&us.z_filefunc, us.
    filestream,&uL)!=UNZ_OK)
734             err=UNZ_ERRNO;
735         us.offset_central_dir = uL;
736
737         /* zipfile comment length */
738         if (unz64local_getShort(&us.z_filefunc, us.
    filestream,&us.gi.size_comment)!=UNZ_OK)
739             err=UNZ_ERRNO;
740     }
741
742     if ((central_pos<us.offset_central_dir+us.size_central_dir) &&
743         (err==UNZ_OK))
744         err=UNZ_BADZIPFILE;
745
746     if (err!=UNZ_OK)
747     {
748         if ((us.flags & UNZ_AUTO_CLOSE) != 0)
749             ZCLOSE64(us.z_filefunc, us.filestream);
750         else
751             ZFAKECLOSE64(us.z_filefunc, us.filestream);
752         return NULL;
753     }
754
755     us.byte_before_the_zipfile = central_pos -
756                             (us.offset_central_dir+us.
    size_central_dir);
757     us.central_pos = central_pos;
758     us.pfile_in_zip_read = NULL;
759     us.encrypted = 0;
760
761
762     s=(unz64_s*)ALLOC(sizeof(unz64_s));
763     if( s != NULL)
764     {
765         *s=us;
766         unzGoToFirstFile((unzFile)s);
767     }
768     return (unzFile)s;
769 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



13.107.4.45 int ZEXPORT unzReadCurrentFile (unzFile *file,* voidp *buf,* unsigned *len*)

Definition at line 1692 of file unzip.c.

References file_in_zip64_read_info_s::byte_before_the_zipfile, file_in_zip64_read_info_s::compression_method, file_in_zip64_read_info_s::crc32, unz64_s::encrypted, file_in_zip64_read_info_s::filestream, unz64_s::keys, unz64_s::pcrc_32_tab, unz64_s::pfile_in_zip_read, file_in_zip64_read_info_s::pos_in_zipfile, file_in_zip64_read_info_s::raw, file_in_zip64_read_info_s::read_buffer, file_in_zip64_read_info_s::rest_read_compressed, file_in_zip64_read_info_s::rest_read_uncompressed, file_in_zip64_read_info_s::stream, file_in_zip64_read_info_s::total_out_64, UNZ_BUFSIZE, UNZ_END_OF_LIST_OF_FILE, UNZ_EOF, UNZ_ERRNO, UNZ_OK, UNZ_PARAMERROR, Z_BZIP2ED, file_in_zip64_read_info_s::z_filefunc, zdecode, ZLIB_FILEFUNC_SEEK_SET, ZREAD64, and ZSEEK64.

Referenced by QuaZipFile::readData().

```
1693 {
1694     int err=UNZ_OK;
1695     uInt iRead = 0;
1696     unz64_s* s;
1697     file_in_zip64_read_info_s* pfile_in_zip_read_info;
1698     if (file==NULL)
```

```
1699         return UNZ_PARAMERROR;
1700     s=(unz64_s*)file;
1701     pfile_in_zip_read_info=s->pfile_in_zip_read;
1702 
1703     if (pfile_in_zip_read_info==NULL)
1704         return UNZ_PARAMERROR;
1705 
1706 
1707     if (pfile_in_zip_read_info->read_buffer == NULL)
1708         return UNZ_END_OF_LIST_OF_FILE;
1709     if (len==0)
1710         return 0;
1711 
1712     pfile_in_zip_read_info->stream.next_out = (Bytef*)buf;
1713 
1714     pfile_in_zip_read_info->stream.avail_out = (uInt)len;
1715 
1716     if ((len>pfile_in_zip_read_info->rest_read_uncompressed) &&
1717         (!(pfile_in_zip_read_info->raw)))
1718         pfile_in_zip_read_info->stream.avail_out =
1719             (uInt)pfile_in_zip_read_info->rest_read_uncompressed;
1720 
1721     if ((len>pfile_in_zip_read_info->rest_read_compressed+
1722            pfile_in_zip_read_info->stream.avail_in) &&
1723          (pfile_in_zip_read_info->raw))
1724         pfile_in_zip_read_info->stream.avail_out =
1725             (uInt)pfile_in_zip_read_info->rest_read_compressed+
1726             pfile_in_zip_read_info->stream.avail_in;
1727 
1728     while (pfile_in_zip_read_info->stream.avail_out>0)
1729     {
1730         if ((pfile_in_zip_read_info->stream.avail_in==0) &&
1731             (pfile_in_zip_read_info->rest_read_compressed>0))
1732         {
1733             uInt uReadThis = UNZ_BUFSIZE;
1734             if (pfile_in_zip_read_info->rest_read_compressed<uReadThis)
1735                 uReadThis = (uInt)pfile_in_zip_read_info->rest_read_compressed;
1736             if (uReadThis == 0)
1737                 return UNZ_EOF;
1738             if (ZSEEK64(pfile_in_zip_read_info->z_filefunc,
1739                       pfile_in_zip_read_info->filestream,
1740                       pfile_in_zip_read_info->pos_in_zipfile +
1741                          pfile_in_zip_read_info->byte_before_the_zipfile,
1742                          ZLIB_FILEFUNC_SEEK_SET)!=0)
1743                 return UNZ_ERRNO;
1744             if (ZREAD64(pfile_in_zip_read_info->z_filefunc,
1745                       pfile_in_zip_read_info->filestream,
1746                       pfile_in_zip_read_info->read_buffer,
1747                       uReadThis)!=uReadThis)
1748                 return UNZ_ERRNO;
1749 
1750 
1751 #            ifndef NOUNCRYPT
1752             if(s->encrypted)
1753             {
1754                 uInt i;
1755                 for(i=0;i<uReadThis;i++)
1756                   pfile_in_zip_read_info->read_buffer[i] =
1757                       zdecode(s->keys,s->pcrc_32_tab,
1758                               pfile_in_zip_read_info->read_buffer[i]);
1759             }
1760 #            endif
1761 
1762 
1763             pfile_in_zip_read_info->pos_in_zipfile += uReadThis;
1764 
1765             pfile_in_zip_read_info->rest_read_compressed-=uReadThis;
1766 
1767             pfile_in_zip_read_info->stream.next_in =
1768                 (Bytef*)pfile_in_zip_read_info->read_buffer;
1769             pfile_in_zip_read_info->stream.avail_in = (uInt)uReadThis;
1770         }
1771 
1772         if ((pfile_in_zip_read_info->compression_method==0) || (pfile_in_zip_read_info->
    raw))
1773         {
1774             uInt uDoCopy,i ;
1775 
1776             if ((pfile_in_zip_read_info->stream.avail_in == 0) &&
1777                 (pfile_in_zip_read_info->rest_read_compressed == 0))
1778                 return (iRead==0) ? UNZ_EOF : iRead;
1779 
1780             if (pfile_in_zip_read_info->stream.avail_out <
1781                             pfile_in_zip_read_info->stream.avail_in)
1782                 uDoCopy = pfile_in_zip_read_info->stream.avail_out ;
1783             else
1784                 uDoCopy = pfile_in_zip_read_info->stream.avail_in ;
```

```
1785
1786            for (i=0;i<uDoCopy;i++)
1787                *(pfile_in_zip_read_info->stream.next_out+i) =
1788                        *(pfile_in_zip_read_info->stream.next_in+i);
1789
1790            pfile_in_zip_read_info->total_out_64 = pfile_in_zip_read_info->
    total_out_64 + uDoCopy;
1791
1792            pfile_in_zip_read_info->crc32 = crc32(pfile_in_zip_read_info->
    crc32,
1793                                pfile_in_zip_read_info->stream.next_out,
1794                                uDoCopy);
1795            pfile_in_zip_read_info->rest_read_uncompressed-=uDoCopy;
1796            pfile_in_zip_read_info->stream.avail_in -= uDoCopy;
1797            pfile_in_zip_read_info->stream.avail_out -= uDoCopy;
1798            pfile_in_zip_read_info->stream.next_out += uDoCopy;
1799            pfile_in_zip_read_info->stream.next_in += uDoCopy;
1800            pfile_in_zip_read_info->stream.total_out += uDoCopy;
1801            iRead += uDoCopy;
1802        }
1803        else if (pfile_in_zip_read_info->compression_method==
    Z_BZIP2ED)
1804        {
1805 #ifdef HAVE_BZIP2
1806            uLong uTotalOutBefore,uTotalOutAfter;
1807            const Bytef *bufBefore;
1808            uLong uOutThis;
1809
1810            pfile_in_zip_read_info->bstream.next_in        = (char*)pfile_in_zip_read_info->
    stream.next_in;
1811            pfile_in_zip_read_info->bstream.avail_in       = pfile_in_zip_read_info->
    stream.avail_in;
1812            pfile_in_zip_read_info->bstream.total_in_lo32  = pfile_in_zip_read_info->
    stream.total_in;
1813            pfile_in_zip_read_info->bstream.total_in_hi32  = 0;
1814            pfile_in_zip_read_info->bstream.next_out       = (char*)pfile_in_zip_read_info->
    stream.next_out;
1815            pfile_in_zip_read_info->bstream.avail_out      = pfile_in_zip_read_info->
    stream.avail_out;
1816            pfile_in_zip_read_info->bstream.total_out_lo32 = pfile_in_zip_read_info->
    stream.total_out;
1817            pfile_in_zip_read_info->bstream.total_out_hi32 = 0;
1818
1819            uTotalOutBefore = pfile_in_zip_read_info->bstream.total_out_lo32;
1820            bufBefore = (const Bytef *)pfile_in_zip_read_info->bstream.next_out;
1821
1822            err=BZ2_bzDecompress(&pfile_in_zip_read_info->bstream);
1823
1824            uTotalOutAfter = pfile_in_zip_read_info->bstream.total_out_lo32;
1825            uOutThis = uTotalOutAfter-uTotalOutBefore;
1826
1827            pfile_in_zip_read_info->total_out_64 = pfile_in_zip_read_info->
    total_out_64 + uOutThis;
1828
1829            pfile_in_zip_read_info->crc32 = crc32(pfile_in_zip_read_info->
    crc32,bufBefore, (uInt)(uOutThis));
1830            pfile_in_zip_read_info->rest_read_uncompressed -= uOutThis;
1831            iRead += (uInt)(uTotalOutAfter - uTotalOutBefore);
1832
1833            pfile_in_zip_read_info->stream.next_in   = (Bytef*)pfile_in_zip_read_info->bstream.
    next_in;
1834            pfile_in_zip_read_info->stream.avail_in  = pfile_in_zip_read_info->bstream.avail_in;
1835            pfile_in_zip_read_info->stream.total_in  = pfile_in_zip_read_info->bstream.total_in_lo32;
1836            pfile_in_zip_read_info->stream.next_out  = (Bytef*)pfile_in_zip_read_info->bstream.
    next_out;
1837            pfile_in_zip_read_info->stream.avail_out = pfile_in_zip_read_info->bstream.avail_out;
1838            pfile_in_zip_read_info->stream.total_out = pfile_in_zip_read_info->bstream.total_out_lo32
    ;
1839
1840            if (err==BZ_STREAM_END)
1841              return (iRead==0) ? UNZ_EOF : iRead;
1842            if (err!=BZ_OK)
1843              break;
1844 #endif
1845        } /* end Z_BZIP2ED */
1846        else
1847        {
1848            ZPOS64_T uTotalOutBefore,uTotalOutAfter;
1849            const Bytef *bufBefore;
1850            ZPOS64_T uOutThis;
1851            int flush=Z_SYNC_FLUSH;
1852
1853            uTotalOutBefore = pfile_in_zip_read_info->stream.total_out;
1854            bufBefore = pfile_in_zip_read_info->stream.next_out;
1855
1856            /*
1857            if ((pfile_in_zip_read_info->rest_read_uncompressed ==
```

```
1858                     pfile_in_zip_read_info->stream.avail_out) &&
1859                 (pfile_in_zip_read_info->rest_read_compressed == 0))
1860                 flush = Z_FINISH;
1861             */
1862             err=inflate(&pfile_in_zip_read_info->stream,flush);
1863 
1864             if ((err>=0) && (pfile_in_zip_read_info->stream.msg!=NULL))
1865               err = Z_DATA_ERROR;
1866 
1867             uTotalOutAfter = pfile_in_zip_read_info->stream.total_out;
1868             uOutThis = uTotalOutAfter-uTotalOutBefore;
1869 
1870             pfile_in_zip_read_info->total_out_64 = pfile_in_zip_read_info->
      total_out_64 + uOutThis;
1871 
1872             pfile_in_zip_read_info->crc32 =
1873                 crc32(pfile_in_zip_read_info->crc32,bufBefore,
1874                         (uInt)(uOutThis));
1875 
1876             pfile_in_zip_read_info->rest_read_uncompressed -=
1877                 uOutThis;
1878 
1879             iRead += (uInt)(uTotalOutAfter - uTotalOutBefore);
1880 
1881             if (err==Z_STREAM_END)
1882                 return (iRead==0) ? UNZ_EOF : iRead;
1883             if (err!=Z_OK)
1884                 break;
1885         }
1886     }
1887 
1888     if (err==Z_OK)
1889         return iRead;
1890     return err;
1891 }
```

Here is the caller graph for this function:

13.107.4.46 int ZEXPORT unzSetFlags (unzFile *file*, unsigned *flags*)

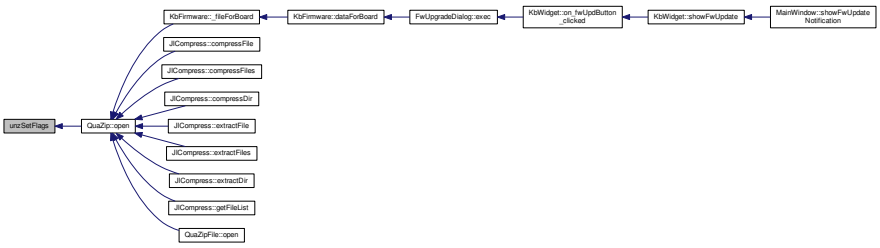
Definition at line 2136 of file unzip.c.

References unz64_s::flags, UNZ_OK, and UNZ_PARAMERROR.

Referenced by QuaZip::open().

```
2137 {
2138     unz64_s* s;
2139     if (file == NULL)
2140         return UNZ_PARAMERROR;
2141     s = (unz64_s*)file;
2142     s->flags |= flags;
2143     return UNZ_OK;
2144 }
```

Here is the caller graph for this function:



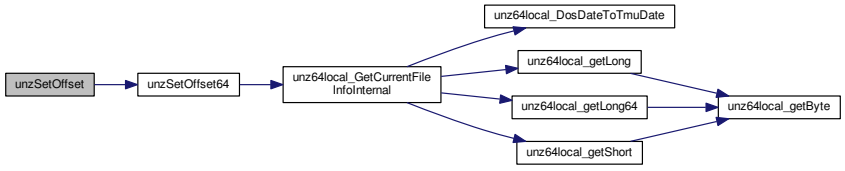
13.107.4.47 int ZEXPORT unzSetOffset (unzFile *file*, uLong *pos*)

Definition at line 2130 of file unzip.c.

References unzSetOffset64().

```
2131 {
2132     return unzSetOffset64(file,pos);
2133 }
```

Here is the call graph for this function:



13.107.4.48 int ZEXPORT unzSetOffset64 (unzFile *file*, ZPOS64_T *pos*)

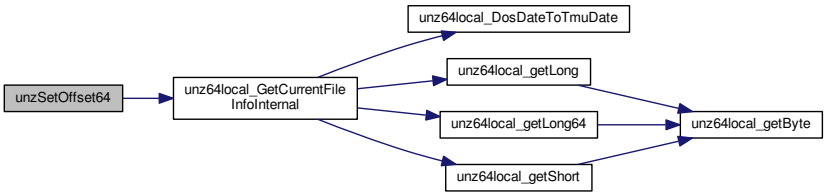
Definition at line 2112 of file unzip.c.

References unz64_s::cur_file_info, unz64_s::cur_file_info_internal, unz64_s::current_file_ok, unz64_s::gi, unz64_s::num_file, unz_global_info64_s::number_entry, unz64_s::pos_in_central_dir, unz64local_GetCurrentFileInfoInternal(), UNZ_OK, and UNZ_PARAMERROR.

Referenced by unzSetOffset().

```
2113 {
2114     unz64_s* s;
2115     int err;
2116
2117     if (file==NULL)
2118         return UNZ_PARAMERROR;
2119     s=(unz64_s*)file;
2120
2121     s->pos_in_central_dir = pos;
2122     s->num_file = s->gi.number_entry;       /* hack */
2123     err = unz64local_GetCurrentFileInfoInternal(file,&s->
      cur_file_info,
2124                                                 &s->cur_file_info_internal,
2125                                                 NULL,0,NULL,0,NULL,0);
2126     s->current_file_ok = (err == UNZ_OK);
2127     return err;
2128 }
```

Here is the call graph for this function:



Here is the caller graph for this function:

13.107.4.49 int ZEXPORT unzStringFileNameCompare (const char * *fileName1*, const char * *fileName2*, int *iCaseSensitivity*)

Definition at line 393 of file unzip.c.

References CASESENSITIVITYDEFAULTVALUE, and STRCMPCASENOSENTIVEFUNCTION.

Referenced by unzLocateFile().

```
397 {
398     if (iCaseSensitivity==0)
399         iCaseSensitivity=CASESENSITIVITYDEFAULTVALUE;
400
401     if (iCaseSensitivity==1)
402         return strcmp(fileName1,fileName2);
403
404     return STRCMPCASENOSENTIVEFUNCTION(fileName1,fileName2);
405 }
```

Here is the caller graph for this function:

13.107.4.50 z_off_t ZEXPORT unztell (unzFile *file*)

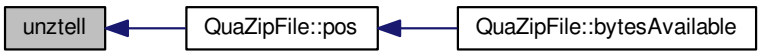
Definition at line 1897 of file unzip.c.

References unz64_s::pfile_in_zip_read, file_in_zip64_read_info_s::stream, and UNZ_PARAMERROR.

Referenced by QuaZipFile::pos().

```
1898 {
1899     unz64_s* s;
1900     file_in_zip64_read_info_s* pfile_in_zip_read_info;
1901     if (file==NULL)
1902         return UNZ_PARAMERROR;
1903     s=(unz64_s*)file;
1904     pfile_in_zip_read_info=s->pfile_in_zip_read;
1905
1906     if (pfile_in_zip_read_info==NULL)
1907         return UNZ_PARAMERROR;
1908
1909     return (z_off_t)pfile_in_zip_read_info->stream.total_out;
1910 }
```

Here is the caller graph for this function:



13.107.4.51 ZPOS64_T ZEXPORT unztell64 (unzFile *file*)

Definition at line 1912 of file unzip.c.

References unz64_s::pfile_in_zip_read, and file_in_zip64_read_info_s::total_out_64.

```
1913 {
1914
1915     unz64_s* s;
1916     file_in_zip64_read_info_s* pfile_in_zip_read_info;
1917     if (file==NULL)
1918         return (ZPOS64_T)-1;
1919     s=(unz64_s*)file;
1920     pfile_in_zip_read_info=s->pfile_in_zip_read;
1921
1922     if (pfile_in_zip_read_info==NULL)
1923         return (ZPOS64_T)-1;
1924
1925     return pfile_in_zip_read_info->total_out_64;
1926 }
```

13.107.5 Variable Documentation

13.107.5.1 const char unz_copyright[ ]

Initial value:

```
=
   " unzip 1.01 Copyright 1998-2004 Gilles Vollant - http://www.winimage.com/zLibDll"
```

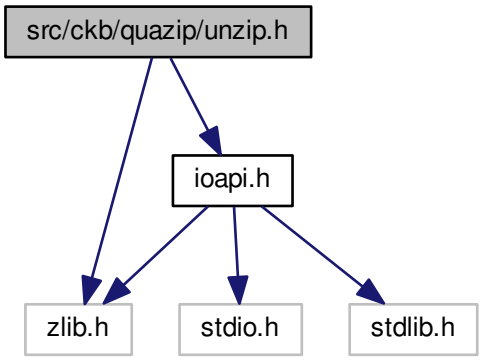
Definition at line 124 of file unzip.c.

Referenced by unzOpenInternal().

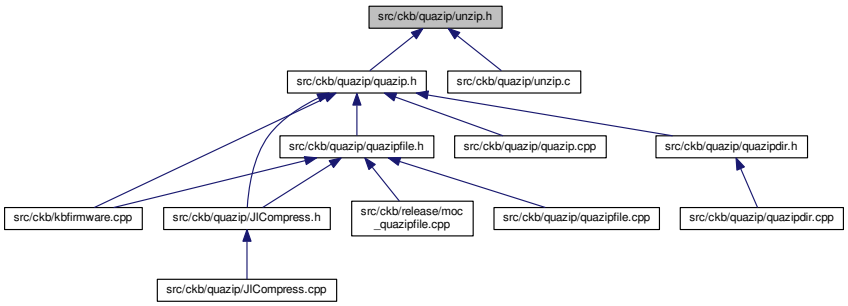
13.108 src/ckb/quazip/unzip.h File Reference

```
#include "zlib.h"
#include "ioapi.h"
```

Include dependency graph for unzip.h:



This graph shows which files directly or indirectly include this file:



Data Structures

- struct tm_unz_s
- struct unz_global_info64_s
- struct unz_global_info_s
- struct unz_file_info64_s
- struct unz_file_info_s
- struct unz_file_pos_s
- struct unz64_file_pos_s

Macros

- #define Z_BZIP2ED 12

- #define UNZ_OK (0)
- #define UNZ_END_OF_LIST_OF_FILE (-100)
- #define UNZ_ERRNO (Z_ERRNO)
- #define UNZ_EOF (0)
- #define UNZ_PARAMERROR (-102)
- #define UNZ_BADZIPFILE (-103)
- #define UNZ_INTERNALERROR (-104)
- #define UNZ_CRCERROR (-105)
- #define UNZ_AUTO_CLOSE 0x01u
- #define UNZ_DEFAULT_FLAGS UNZ_AUTO_CLOSE

Typedefs

- typedef voidp unzFile
- typedef struct tm_unz_s tm_unz
- typedef struct unz_global_info64_s unz_global_info64
- typedef struct unz_global_info_s unz_global_info
- typedef struct unz_file_info64_s unz_file_info64
- typedef struct unz_file_info_s unz_file_info
- typedef struct unz_file_pos_s unz_file_pos
- typedef struct unz64_file_pos_s unz64_file_pos

Functions

- int ZEXPORT unzStringFileNameCompare _Z_OF ((const char ∗fileName1, const char ∗fileName2, int iCaseSensitivity))
- unzFile ZEXPORT unzOpen _Z_OF ((voidpf file))
- unzFile ZEXPORT unzOpen2 _Z_OF ((voidpf file, zlib_filefunc_def ∗pzlib_filefunc_def))
- unzFile ZEXPORT unzOpen2_64 _Z_OF ((voidpf file, zlib_filefunc64_def ∗pzlib_filefunc_def))
- unzFile unzOpenInternal (voidpf file, zlib_filefunc64_32_def ∗pzlib_filefunc64_32_def, int is64bitOpenFunction, unsigned flags)
- int ZEXPORT unzClose _Z_OF ((unzFile file))

  Addition for GDAL : START.
- int ZEXPORT unzGetGlobalInfo _Z_OF ((unzFile file, unz_global_info ∗pglobal_info))
- int ZEXPORT unzGetGlobalInfo64 _Z_OF ((unzFile file, unz_global_info64 ∗pglobal_info))
- int ZEXPORT unzGetGlobalComment _Z_OF ((unzFile file, char ∗szComment, uLong uSizeBuf))
- int ZEXPORT unzLocateFile _Z_OF ((unzFile file, const char ∗szFileName, int iCaseSensitivity))
- int ZEXPORT unzGetFilePos (unzFile file, unz_file_pos ∗file_pos)
- int ZEXPORT unzGoToFilePos (unzFile file, unz_file_pos ∗file_pos)
- int ZEXPORT unzGetFilePos64 (unzFile file, unz64_file_pos ∗file_pos)
- int ZEXPORT unzGoToFilePos64 (unzFile file, const unz64_file_pos ∗file_pos)
- int ZEXPORT unzGetCurrentFileInfo64 _Z_OF ((unzFile file, unz_file_info64 ∗pfile_info, char ∗szFileName, uLong fileNameBufferSize, void ∗extraField, uLong extraFieldBufferSize, char ∗szComment, uLong commentBufferSize))
- int ZEXPORT unzGetCurrentFileInfo _Z_OF ((unzFile file, unz_file_info ∗pfile_info, char ∗szFileName, uLong fileNameBufferSize, void ∗extraField, uLong extraFieldBufferSize, char ∗szComment, uLong commentBufferSize))
- int ZEXPORT unzOpenCurrentFilePassword _Z_OF ((unzFile file, const char ∗password))
- int ZEXPORT unzOpenCurrentFile2 _Z_OF ((unzFile file, int ∗method, int ∗level, int raw))
- int ZEXPORT unzOpenCurrentFile3 _Z_OF ((unzFile file, int ∗method, int ∗level, int raw, const char ∗password))
- int ZEXPORT unzReadCurrentFile _Z_OF ((unzFile file, voidp buf, unsigned len))
- ZPOS64_T ZEXPORT unzGetOffset64 (unzFile file)

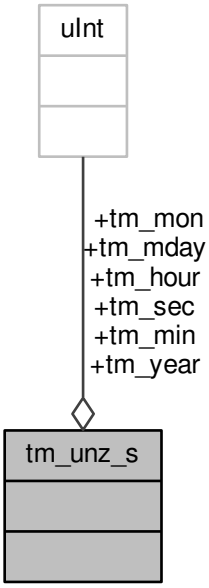
- uLong ZEXPORT unzGetOffset (unzFile file)
- int ZEXPORT unzSetOffset64 (unzFile file, ZPOS64_T pos)
- int ZEXPORT unzSetOffset (unzFile file, uLong pos)
- int ZEXPORT unzSetFlags (unzFile file, unsigned flags)
- int ZEXPORT unzClearFlags (unzFile file, unsigned flags)

13.108.1 Data Structure Documentation

13.108.1.1 struct tm_unz_s

Definition at line 92 of file unzip.h.

Collaboration diagram for tm_unz_s:



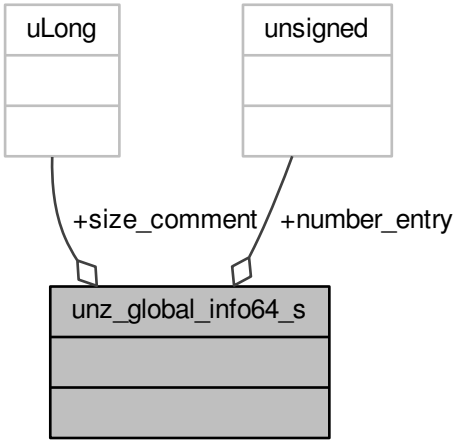
Data Fields

uInt	tm_hour	
uInt	tm_mday	
uInt	tm_min	
uInt	tm_mon	
uInt	tm_sec	
uInt	tm_year	

13.108.1.2 struct unz_global_info64_s

Definition at line 104 of file unzip.h.

Collaboration diagram for unz_global_info64_s:



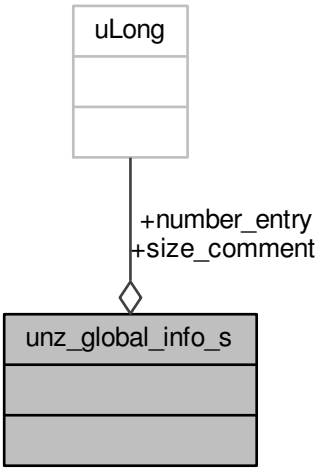
Data Fields

ZPOS64_T	number_entry	
uLong	size_comment	

13.108.1.3 struct unz_global_info_s

Definition at line 111 of file unzip.h.

Collaboration diagram for unz_global_info_s:



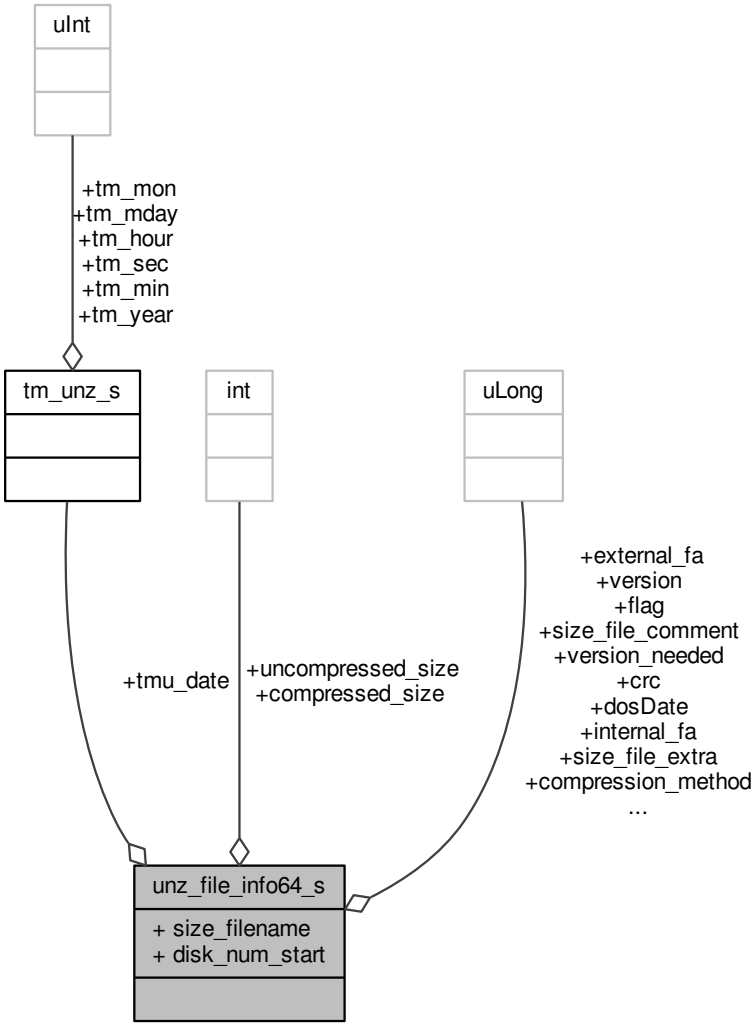
Data Fields

uLong	number_entry	
uLong	size_comment	

13.108.1.4 struct unz_file_info64_s

Definition at line 119 of file unzip.h.

Collaboration diagram for unz_file_info64_s:



Data Fields

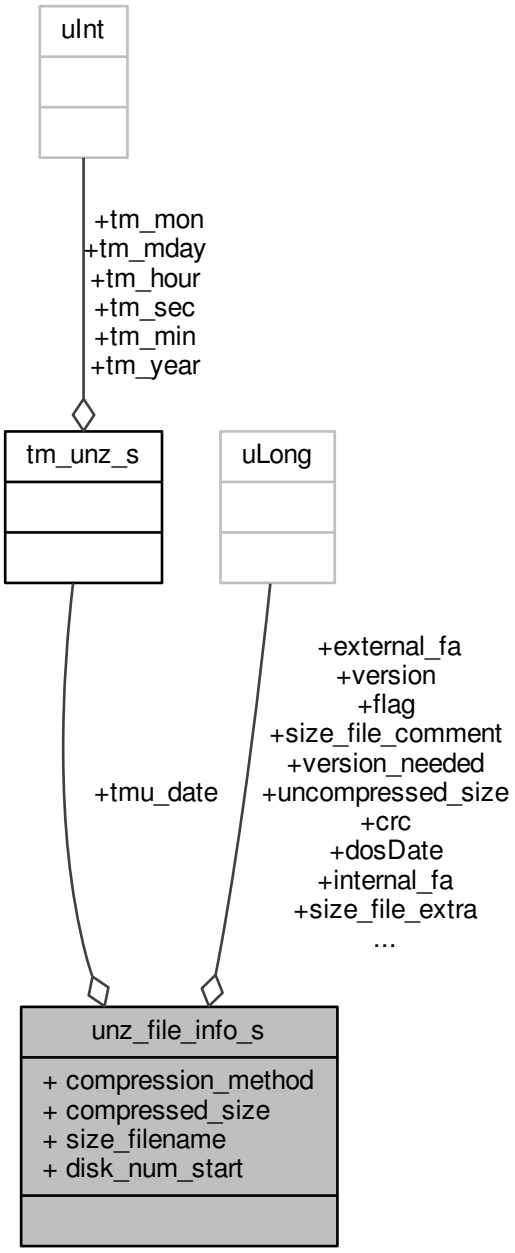
ZPOS64_T	compressed_-size	

uLong	compression_- method	
uLong	crc	
uLong	disk_num_start	
uLong	dosDate	
uLong	external_fa	
uLong	flag	
uLong	internal_fa	
uLong	size_file_- comment	
uLong	size_file_extra	
uLong	size_filename	
tm_unz	tmu_date	
ZPOS64_T	uncompressed_- size	
uLong	version	
uLong	version_needed	

13.108.1.5 struct unz_file_info_s

Definition at line 140 of file unzip.h.

Collaboration diagram for unz_file_info_s:



Data Fields

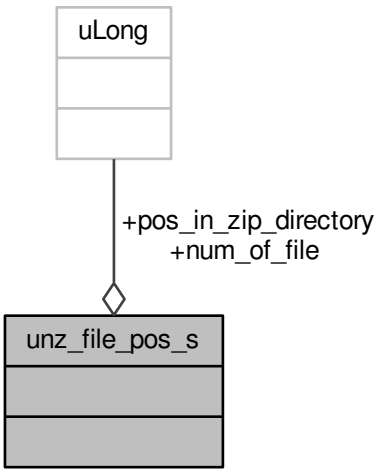
uLong	compressed_-size	

uLong	compression_-method	
uLong	crc	
uLong	disk_num_start	
uLong	dosDate	
uLong	external_fa	
uLong	flag	
uLong	internal_fa	
uLong	size_file_-comment	
uLong	size_file_extra	
uLong	size_filename	
tm_unz	tmu_date	
uLong	uncompressed_-size	
uLong	version	
uLong	version_needed	

13.108.1.6 struct unz_file_pos_s

Definition at line 278 of file unzip.h.

Collaboration diagram for unz_file_pos_s:



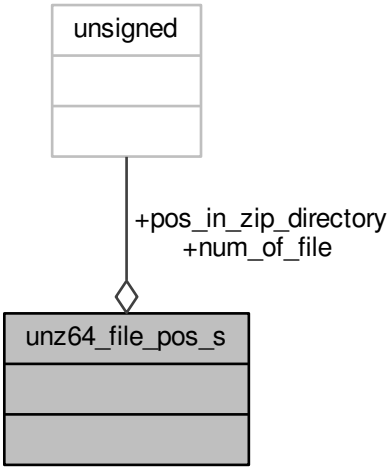
Data Fields

uLong	num_of_file	
uLong	pos_in_zip_-directory	

13.108.1.7 struct unz64_file_pos_s

Definition at line 292 of file unzip.h.

Collaboration diagram for unz64_file_pos_s:



Data Fields

ZPOS64_T	num_of_file	
ZPOS64_T	pos_in_zip_-directory	

13.108.2 Macro Definition Documentation

13.108.2.1 #define UNZ_AUTO_CLOSE 0x01u

Definition at line 88 of file unzip.h.

Referenced by QuaZip::open(), unzClose(), and unzOpenInternal().

13.108.2.2 #define UNZ_BADZIPFILE (-103)

Definition at line 84 of file unzip.h.

Referenced by unz64local_CheckCurrentFileCoherencyHeader(), unz64local_GetCurrentFileInfoInternal(), unzOpenCurrentFile3(), and unzOpenInternal().

13.108.2.3 #define UNZ_CRCERROR (-105)

Definition at line 86 of file unzip.h.

Referenced by unzCloseCurrentFile().

13.108.2.4 #define UNZ_DEFAULT_FLAGS UNZ_AUTO_CLOSE

Definition at line 89 of file unzip.h.

Referenced by unzOpen(), unzOpen2(), unzOpen2_64(), and unzOpen64().

13.108.2.5 #define UNZ_END_OF_LIST_OF_FILE (-100)

Definition at line 80 of file unzip.h.

Referenced by QuaZipPrivate::goToFirstUnmappedFile(), QuaZip::goToNextFile(), unzGetFilePos64(), unzGoToNextFile(), unzLocateFile(), and unzReadCurrentFile().

13.108.2.6 #define UNZ_EOF (0)

Definition at line 82 of file unzip.h.

Referenced by unz64local_getByte(), and unzReadCurrentFile().

13.108.2.7 #define UNZ_ERRNO (Z_ERRNO)

Definition at line 81 of file unzip.h.

Referenced by unz64local_CheckCurrentFileCoherencyHeader(), unz64local_getByte(), unz64local_GetCurrentFileInfoInternal(), unzGetGlobalComment(), unzGetLocalExtrafield(), unzOpenInternal(), and unzReadCurrentFile().

13.108.2.8 #define UNZ_INTERNALERROR (-104)

Definition at line 85 of file unzip.h.

Referenced by unzOpenCurrentFile3().

13.108.2.9 #define UNZ_OK (0)

Definition at line 79 of file unzip.h.

Referenced by QuaZip::close(), QuaZipFile::close(), JlCompress::compressFile(), QuaZipFile::csize(), JlCompress::extractFile(), QuaZipFile::getActualFileName(), QuaZip::getComment(), QuaZip::getCurrentFileInfo(), QuaZip::getCurrentFileName(), QuaZip::getEntriesCount(), QuaZipFile::getFileInfo(), QuaZipPrivate::getFileInfoList(), QuaZip::goToFirstFile(), QuaZipPrivate::goToFirstUnmappedFile(), QuaZip::goToNextFile(), QuaZip::open(), QuaZipFile::open(), QuaZipFile::readData(), QuaZipFilePrivate::resetZipError(), QuaZip::setCurrentFile(), QuaZipFilePrivate::setZipError(), unz64local_CheckCurrentFileCoherencyHeader(), unz64local_getByte(), unz64local_GetCurrentFileInfoInternal(), unz64local_getLong(), unz64local_getLong64(), unz64local_getShort(), unz64local_SearchCentralDir64(), unzClearFlags(), unzClose(), unzCloseCurrentFile(), unzGetCurrentFileInfo(), unzGetFilePos(), unzGetFilePos64(), unzGetGlobalInfo(), unzGetGlobalInfo64(), unzGoToFilePos64(), unzGoToFirstFile(), unzGoToNextFile(), unzLocateFile(), unzOpenCurrentFile3(), unzOpenInternal(), unzReadCurrentFile(), unzSetFlags(), unzSetOffset64(), and QuaZipFile::usize().

13.108.2.10 #define UNZ_PARAMERROR (-102)

Definition at line 83 of file unzip.h.

Referenced by QuaZip::setCurrentFile(), unz64local_GetCurrentFileInfoInternal(), unzClearFlags(), unzClose(), unzCloseCurrentFile(), unzeof(), unzGetFilePos64(), unzGetGlobalComment(), unzGetGlobalInfo(), unzGetGlobalInfo64(), unzGetLocalExtrafield(), unzGoToFilePos(), unzGoToFilePos64(), unzGoToFirstFile(), unzGoToNextFile(), unzLocateFile(), unzOpenCurrentFile3(), unzReadCurrentFile(), unzSetFlags(), unzSetOffset64(), and unztell().

13.108.2.11 #define Z_BZIP2ED 12

Definition at line 67 of file unzip.h.

Referenced by unz64local_CheckCurrentFileCoherencyHeader(), unzCloseCurrentFile(), unzOpenCurrentFile3(), unzReadCurrentFile(), zip64FlushWriteBuffer(), zipCloseFileInZipRaw64(), zipOpenNewFileInZip4_64(), and zipWriteInFileInZip().

13.108.3 Typedef Documentation

13.108.3.1 typedef struct tm_unz_s tm_unz

13.108.3.2 typedef struct unz64_file_pos_s unz64_file_pos

13.108.3.3 typedef struct unz_file_info_s unz_file_info

13.108.3.4 typedef struct unz_file_info64_s unz_file_info64

13.108.3.5 typedef struct unz_file_pos_s unz_file_pos

13.108.3.6 typedef struct unz_global_info_s unz_global_info

13.108.3.7 typedef struct unz_global_info64_s unz_global_info64

13.108.3.8 typedef voidp unzFile

Definition at line 75 of file unzip.h.

13.108.4 Function Documentation

13.108.4.1 int ZEXPORT unzStringFileNameCompare _Z_OF ((const char *fileName1, const char *fileName2, int iCaseSensitivity))

13.108.4.2 unzFile ZEXPORT unzOpen _Z_OF ((voidpf file))

13.108.4.3 unzFile ZEXPORT unzOpen2 _Z_OF ((voidpf file, zlib_filefunc_def *pzlib_filefunc_def))

13.108.4.4 unzFile ZEXPORT unzOpen2_64 _Z_OF ((voidpf file, zlib_filefunc64_def *pzlib_filefunc_def))

13.108.4.5 int ZEXPORT unzClose _Z_OF ((unzFile file))

Addition for GDAL : END.

13.108.4.6 int ZEXPORT unzGetGlobalInfo _Z_OF ((unzFile file, unz_global_info *pglobal_info))

13.108.4.7 int ZEXPORT unzGetGlobalInfo64 _Z_OF ((unzFile file, unz_global_info64 *pglobal_info))

13.108.4.8 int ZEXPORT unzGetGlobalComment _Z_OF ((unzFile file, char *szComment, uLong uSizeBuf))

13.108.4.9 int ZEXPORT unzLocateFile _Z_OF ((unzFile file, const char *szFileName, int iCaseSensitivity))

13.108.4.10 int ZEXPORT unzGetCurrentFileInfo64 _Z_OF ((unzFile file, unz_file_info64 *pfile_info, char *szFileName, uLong fileNameBufferSize, void *extraField, uLong extraFieldBufferSize, char *szComment, uLong commentBufferSize))

13.108.4.11 int ZEXPORT unzGetCurrentFileInfo _Z_OF ((unzFile file, unz_file_info ∗pfile_info, char ∗szFileName, uLong fileNameBufferSize, void ∗extraField, uLong extraFieldBufferSize, char ∗szComment, uLong commentBufferSize))

13.108.4.12 int ZEXPORT unzOpenCurrentFilePassword _Z_OF ((unzFile file, const char ∗password))

13.108.4.13 int ZEXPORT unzOpenCurrentFile2 _Z_OF ((unzFile file, int ∗method, int ∗level, int raw))

13.108.4.14 int ZEXPORT unzOpenCurrentFile3 _Z_OF ((unzFile file, int ∗method, int ∗level, int raw, const char ∗password))

13.108.4.15 int ZEXPORT unzReadCurrentFile _Z_OF ((unzFile file, voidp buf, unsigned len))

13.108.4.16 int ZEXPORT unzClearFlags (unzFile *file,* unsigned *flags*)

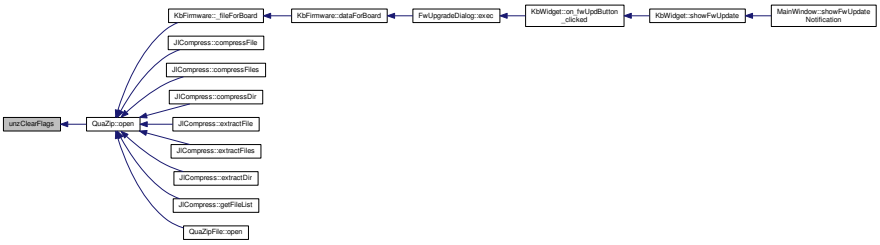
Definition at line 2147 of file unzip.c.

References unz64_s::flags, UNZ_OK, and UNZ_PARAMERROR.

Referenced by QuaZip::open().

```
2148 {
2149     unz64_s* s;
2150     if (file == NULL)
2151         return UNZ_PARAMERROR;
2152     s = (unz64_s*)file;
2153     s->flags &= ~flags;
2154     return UNZ_OK;
2155 }
```

Here is the caller graph for this function:



13.108.4.17 int ZEXPORT unzGetFilePos (unzFile *file,* unz_file_pos ∗ *file_pos*)

Definition at line 1334 of file unzip.c.

References unz_file_pos_s::num_of_file, unz64_file_pos_s::num_of_file, unz_file_pos_s::pos_in_zip_directory, unz64_file_pos_s::pos_in_zip_directory, UNZ_OK, and unzGetFilePos64().

```
1337 {
1338     unz64_file_pos file_pos64;
1339     int err = unzGetFilePos64(file,&file_pos64);
1340     if (err==UNZ_OK)
1341     {
1342         file_pos->pos_in_zip_directory = (uLong)file_pos64.
      pos_in_zip_directory;
1343         file_pos->num_of_file = (uLong)file_pos64.num_of_file;
1344     }
1345     return err;
1346 }
```

Here is the call graph for this function:

13.108.4.18 int ZEXPORT unzGetFilePos64 (unzFile *file*, unz64_file_pos * *file_pos*)

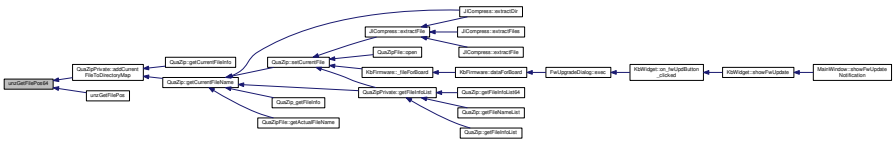
Definition at line 1318 of file unzip.c.

References unz64_s::current_file_ok, unz64_s::num_file, unz64_file_pos_s::num_of_file, unz64_s::pos_in_central_dir, unz64_file_pos_s::pos_in_zip_directory, UNZ_END_OF_LIST_OF_FILE, UNZ_OK, and UNZ_PARAMERROR.

Referenced by QuaZipPrivate::addCurrentFileToDirectoryMap(), and unzGetFilePos().

```
1319 {
1320     unz64_s* s;
1321
1322     if (file==NULL || file_pos==NULL)
1323         return UNZ_PARAMERROR;
1324     s=(unz64_s*)file;
1325     if (!s->current_file_ok)
1326         return UNZ_END_OF_LIST_OF_FILE;
1327
1328     file_pos->pos_in_zip_directory  = s->pos_in_central_dir;
1329     file_pos->num_of_file           = s->num_file;
1330
1331     return UNZ_OK;
1332 }
```

Here is the caller graph for this function:



13.108.4.19 uLong ZEXPORT unzGetOffset (unzFile *file*)

Definition at line 2102 of file unzip.c.

References unzGetOffset64().

```
2103 {
2104     ZPOS64_T offset64;
2105
2106     if (file==NULL)
2107           return 0; /*UNZ_PARAMERROR; */
2108     offset64 = unzGetOffset64(file);
2109     return (uLong)offset64;
2110 }
```

Here is the call graph for this function:

13.108.4.20 ZPOS64_T ZEXPORT unzGetOffset64 (unzFile *file*)

Definition at line 2087 of file unzip.c.

References unz64_s::current_file_ok, unz64_s::gi, unz64_s::num_file, unz_global_info64_s::number_entry, and unz64_s::pos_in_central_dir.

Referenced by unzGetOffset().

```
2088 {
2089     unz64_s* s;
2090
2091     if (file==NULL)
2092           return 0; /*UNZ_PARAMERROR; */
2093     s=(unz64_s*)file;
2094     if (!s->current_file_ok)
2095       return 0;
2096     if (s->gi.number_entry != 0 && s->gi.number_entry != 0xffff)
2097       if (s->num_file==s->gi.number_entry)
2098          return 0;
2099     return s->pos_in_central_dir;
2100 }
```

Here is the caller graph for this function:

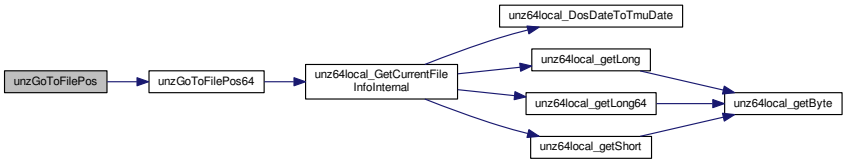
13.108.4.21 int ZEXPORT unzGoToFilePos (unzFile *file,* unz_file_pos * *file_pos*)

Definition at line 1370 of file unzip.c.

References unz_file_pos_s::num_of_file, unz64_file_pos_s::num_of_file, unz_file_pos_s::pos_in_zip_directory, unz64_file_pos_s::pos_in_zip_directory, UNZ_PARAMERROR, and unzGoToFilePos64().

```
1373 {
1374     unz64_file_pos file_pos64;
1375     if (file_pos == NULL)
1376         return UNZ_PARAMERROR;
1377
1378     file_pos64.pos_in_zip_directory = file_pos->
    pos_in_zip_directory;
1379     file_pos64.num_of_file = file_pos->num_of_file;
1380     return unzGoToFilePos64(file,&file_pos64);
1381 }
```

Here is the call graph for this function:



13.108.4.22 int ZEXPORT unzGoToFilePos64 (unzFile *file*, const unz64_file_pos * *file_pos*)

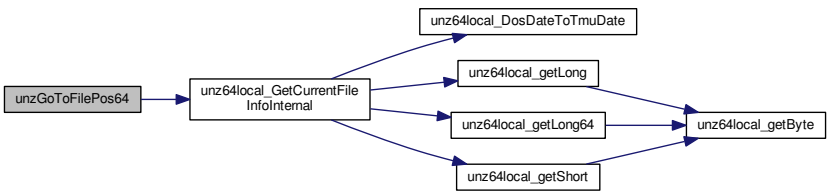
Definition at line 1348 of file unzip.c.

References unz64_s::cur_file_info, unz64_s::cur_file_info_internal, unz64_s::current_file_ok, unz64_s::num_file, unz64_file_pos_s::num_of_file, unz64_s::pos_in_central_dir, unz64_file_pos_s::pos_in_zip_directory, unz64local_GetCurrentFileInfoInternal(), UNZ_OK, and UNZ_PARAMERROR.

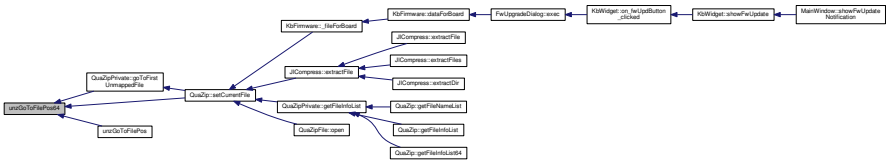
Referenced by QuaZipPrivate::goToFirstUnmappedFile(), QuaZip::setCurrentFile(), and unzGoToFilePos().

```
1349 {
1350     unz64_s* s;
1351     int err;
1352
1353     if (file==NULL || file_pos==NULL)
1354         return UNZ_PARAMERROR;
1355     s=(unz64_s*)file;
1356
1357     /* jump to the right spot */
1358     s->pos_in_central_dir = file_pos->pos_in_zip_directory;
1359     s->num_file           = file_pos->num_of_file;
1360
1361     /* set the current file */
1362     err = unz64local_GetCurrentFileInfoInternal(file,&s->
      cur_file_info,
1363                                                 &s->cur_file_info_internal,
1364                                                 NULL,0,NULL,0,NULL,0);
1365     /* return results */
1366     s->current_file_ok = (err == UNZ_OK);
1367     return err;
1368 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



13.108.4.23 unzFile unzOpenInternal (voidpf *file*, zlib_filefunc64_32_def * *pzlib_filefunc64_32_def*, int *is64bitOpenFunction*, unsigned *flags*)

Definition at line 587 of file unzip.c.

References ALLOC, unz64_s::byte_before_the_zipfile, unz64_s::central_pos, unz64_s::encrypted, unz64_s::filestream, fill_qiodevice64_filefunc(), unz64_s::flags, unz64_s::gi, unz64_s::is64bitOpenFunction, unz64_s::isZip64, unz_global_info64_s::number_entry, unz64_s::offset_central_dir, unz64_s::pfile_in_zip_read, unz64_s::size_central_dir, unz_global_info64_s::size_comment, unz64local_getLong(), unz64local_getLong64(), unz64local_getShort(), unz64local_SearchCentralDir(), unz64local_SearchCentralDir64(), UNZ_AUTO_CLOSE, UNZ_BADZIPFILE, unz_copyright, UNZ_ERRNO, UNZ_OK, unzGoToFirstFile(), unz64_s::z_filefunc, ZCLOSE64, ZFAKECLOSE64, zlib_filefunc64_32_def_s::zfile_func64, ZLIB_FILEFUNC_MODE_EXISTING, ZLIB_FILEFUNC_MODE_READ, ZLIB_FILEFUNC_SEEK_SET, ZOPEN64, zlib_filefunc64_32_def_s::zseek32_file, ZSEEK64, and zlib_filefunc64_32_def_s::ztell32_file.

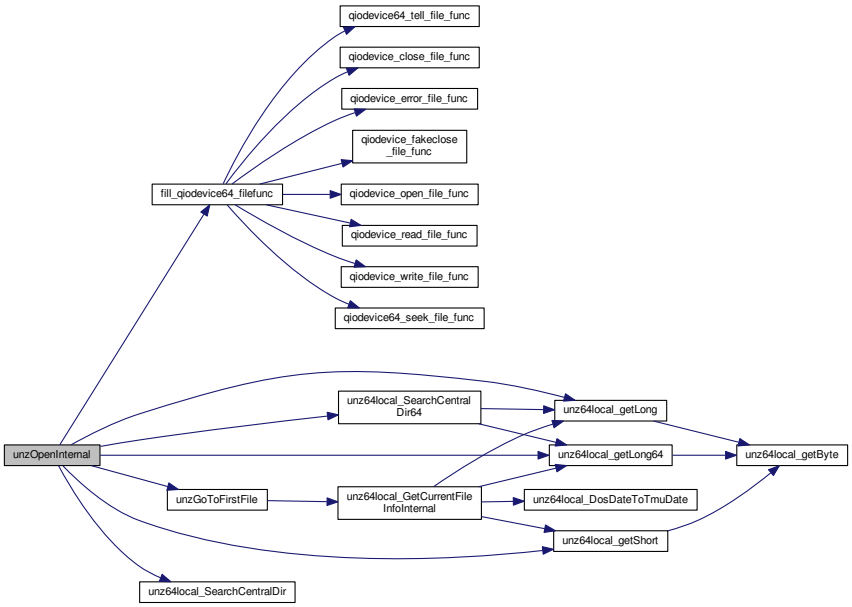
Referenced by QuaZip::open(), unzOpen(), unzOpen2(), unzOpen2_64(), and unzOpen64().

```
590 {
591     unz64_s us;
592     unz64_s *s;
593     ZPOS64_T central_pos;
594     uLong   uL;
595 
596     uLong number_disk;          /* number of the current dist, used for
597                                    spaning ZIP, unsupported, always 0*/
598     uLong number_disk_with_CD;  /* number the the disk with central dir, used
599                                    for spaning ZIP, unsupported, always 0*/
600     ZPOS64_T number_entry_CD;       /* total number of entries in
601                                    the central dir
602                                    (same than number_entry on nospan) */
603 
604     int err=UNZ_OK;
605 
606     if (unz_copyright[0]!=' ')
607         return NULL;
608 
609     us.flags = flags;
610     us.z_filefunc.zseek32_file = NULL;
611     us.z_filefunc.ztell32_file = NULL;
612     if (pzlib_filefunc64_32_def==NULL)
613         fill_qiodevice64_filefunc(&us.z_filefunc.
    zfile_func64);
614     else
615         us.z_filefunc = *pzlib_filefunc64_32_def;
616     us.is64bitOpenFunction = is64bitOpenFunction;
617 
618 
619 
620     us.filestream = ZOPEN64(us.z_filefunc,
621                                                 file,
622                                                 ZLIB_FILEFUNC_MODE_READ |
623                                                 ZLIB_FILEFUNC_MODE_EXISTING);
624     if (us.filestream==NULL)
625         return NULL;
626 
627     central_pos = unz64local_SearchCentralDir64(&us.
    z_filefunc,us.filestream);
628     if (central_pos)
629     {
630         uLong uS;
631         ZPOS64_T uL64;
632 
```

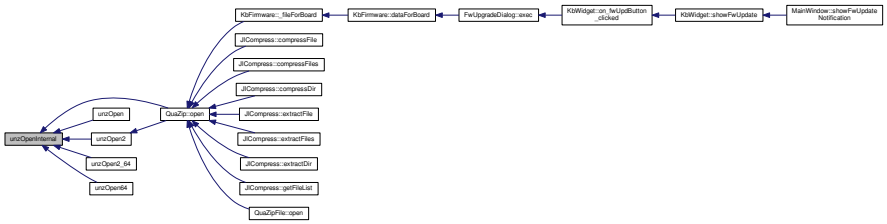
```
633         us.isZip64 = 1;
634
635         if (ZSEEK64(us.z_filefunc, us.filestream,
636                                         central_pos,ZLIB_FILEFUNC_SEEK_SET)!=0)
637         err=UNZ_ERRNO;
638
639         /* the signature, already checked */
640         if (unz64local_getLong(&us.z_filefunc, us.
    filestream,&uL)!=UNZ_OK)
641             err=UNZ_ERRNO;
642
643         /* size of zip64 end of central directory record */
644         if (unz64local_getLong64(&us.z_filefunc, us.
    filestream,&uL64)!=UNZ_OK)
645             err=UNZ_ERRNO;
646
647         /* version made by */
648         if (unz64local_getShort(&us.z_filefunc, us.
    filestream,&uS)!=UNZ_OK)
649             err=UNZ_ERRNO;
650
651         /* version needed to extract */
652         if (unz64local_getShort(&us.z_filefunc, us.
    filestream,&uS)!=UNZ_OK)
653             err=UNZ_ERRNO;
654
655         /* number of this disk */
656         if (unz64local_getLong(&us.z_filefunc, us.
    filestream,&number_disk)!=UNZ_OK)
657             err=UNZ_ERRNO;
658
659         /* number of the disk with the start of the central directory */
660         if (unz64local_getLong(&us.z_filefunc, us.
    filestream,&number_disk_with_CD)!=UNZ_OK)
661             err=UNZ_ERRNO;
662
663         /* total number of entries in the central directory on this disk */
664         if (unz64local_getLong64(&us.z_filefunc, us.
    filestream,&us.gi.number_entry)!=UNZ_OK)
665             err=UNZ_ERRNO;
666
667         /* total number of entries in the central directory */
668         if (unz64local_getLong64(&us.z_filefunc, us.
    filestream,&number_entry_CD)!=UNZ_OK)
669             err=UNZ_ERRNO;
670
671         if ((number_entry_CD!=us.gi.number_entry) ||
672             (number_disk_with_CD!=0) ||
673             (number_disk!=0))
674             err=UNZ_BADZIPFILE;
675
676         /* size of the central directory */
677         if (unz64local_getLong64(&us.z_filefunc, us.
    filestream,&us.size_central_dir)!=UNZ_OK)
678             err=UNZ_ERRNO;
679
680         /* offset of start of central directory with respect to the
681           starting disk number */
682         if (unz64local_getLong64(&us.z_filefunc, us.
    filestream,&us.offset_central_dir)!=UNZ_OK)
683             err=UNZ_ERRNO;
684
685         us.gi.size_comment = 0;
686     }
687     else
688     {
689         central_pos = unz64local_SearchCentralDir(&us.
    z_filefunc,us.filestream);
690         if (central_pos==0)
691             err=UNZ_ERRNO;
692
693         us.isZip64 = 0;
694
695         if (ZSEEK64(us.z_filefunc, us.filestream,
696                                         central_pos,ZLIB_FILEFUNC_SEEK_SET)!=0)
697             err=UNZ_ERRNO;
698
699         /* the signature, already checked */
700         if (unz64local_getLong(&us.z_filefunc, us.
    filestream,&uL)!=UNZ_OK)
701             err=UNZ_ERRNO;
702
703         /* number of this disk */
704         if (unz64local_getShort(&us.z_filefunc, us.
    filestream,&number_disk)!=UNZ_OK)
705             err=UNZ_ERRNO;
706
```

```
707         /* number of the disk with the start of the central directory */
708         if (unz64local_getShort(&us.z_filefunc, us.
    filestream,&number_disk_with_CD)!=UNZ_OK)
709             err=UNZ_ERRNO;
710
711         /* total number of entries in the central dir on this disk */
712         if (unz64local_getShort(&us.z_filefunc, us.
    filestream,&uL)!=UNZ_OK)
713             err=UNZ_ERRNO;
714         us.gi.number_entry = uL;
715
716         /* total number of entries in the central dir */
717         if (unz64local_getShort(&us.z_filefunc, us.
    filestream,&uL)!=UNZ_OK)
718             err=UNZ_ERRNO;
719         number_entry_CD = uL;
720
721         if ((number_entry_CD!=us.gi.number_entry) ||
722             (number_disk_with_CD!=0) ||
723             (number_disk!=0))
724             err=UNZ_BADZIPFILE;
725
726         /* size of the central directory */
727         if (unz64local_getLong(&us.z_filefunc, us.
    filestream,&uL)!=UNZ_OK)
728             err=UNZ_ERRNO;
729         us.size_central_dir = uL;
730
731         /* offset of start of central directory with respect to the
732             starting disk number */
733         if (unz64local_getLong(&us.z_filefunc, us.
    filestream,&uL)!=UNZ_OK)
734             err=UNZ_ERRNO;
735         us.offset_central_dir = uL;
736
737         /* zipfile comment length */
738         if (unz64local_getShort(&us.z_filefunc, us.
    filestream,&us.gi.size_comment)!=UNZ_OK)
739             err=UNZ_ERRNO;
740     }
741
742     if ((central_pos<us.offset_central_dir+us.size_central_dir) &&
743         (err==UNZ_OK))
744         err=UNZ_BADZIPFILE;
745
746     if (err!=UNZ_OK)
747     {
748         if ((us.flags & UNZ_AUTO_CLOSE) != 0)
749             ZCLOSE64(us.z_filefunc, us.filestream);
750         else
751             ZFAKECLOSE64(us.z_filefunc, us.filestream);
752         return NULL;
753     }
754
755     us.byte_before_the_zipfile = central_pos -
756                             (us.offset_central_dir+us.
    size_central_dir);
757     us.central_pos = central_pos;
758     us.pfile_in_zip_read = NULL;
759     us.encrypted = 0;
760
761
762     s=(unz64_s*)ALLOC(sizeof(unz64_s));
763     if( s != NULL)
764     {
765         *s=us;
766         unzGoToFirstFile((unzFile)s);
767     }
768     return (unzFile)s;
769 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



13.108.4.24 int ZEXPORT unzSetFlags (unzFile *file*, unsigned *flags*)

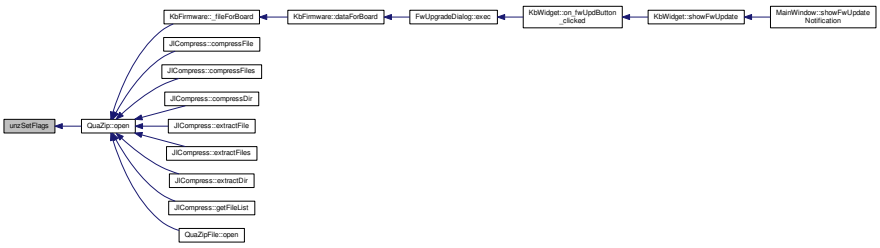
Definition at line 2136 of file unzip.c.

References unz64_s::flags, UNZ_OK, and UNZ_PARAMERROR.

Referenced by QuaZip::open().

```
2137 {
2138     unz64_s* s;
2139     if (file == NULL)
2140         return UNZ_PARAMERROR;
2141     s = (unz64_s*)file;
2142     s->flags |= flags;
2143     return UNZ_OK;
2144 }
```

Here is the caller graph for this function:



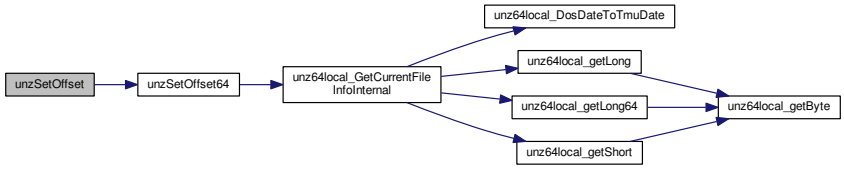
13.108.4.25 int ZEXPORT unzSetOffset (unzFile *file*, uLong *pos*)

Definition at line 2130 of file unzip.c.

References unzSetOffset64().

```
2131 {
2132     return unzSetOffset64(file,pos);
2133 }
```

Here is the call graph for this function:



13.108.4.26 int ZEXPORT unzSetOffset64 (unzFile *file*, ZPOS64_T *pos*)

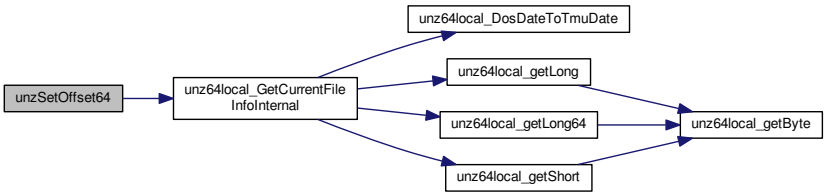
Definition at line 2112 of file unzip.c.

References unz64_s::cur_file_info, unz64_s::cur_file_info_internal, unz64_s::current_file_ok, unz64_s::gi, unz64_s::num_file, unz_global_info64_s::number_entry, unz64_s::pos_in_central_dir, unz64local_GetCurrentFileInfoInternal(), UNZ_OK, and UNZ_PARAMERROR.

Referenced by unzSetOffset().

```
2113 {
2114     unz64_s* s;
2115     int err;
2116
2117     if (file==NULL)
2118         return UNZ_PARAMERROR;
2119     s=(unz64_s*)file;
2120
2121     s->pos_in_central_dir = pos;
2122     s->num_file = s->gi.number_entry;       /* hack */
2123     err = unz64local_GetCurrentFileInfoInternal(file,&s->
      cur_file_info,
2124                                                 &s->cur_file_info_internal,
2125                                                 NULL,0,NULL,0,NULL,0);
2126     s->current_file_ok = (err == UNZ_OK);
2127     return err;
2128 }
```

Here is the call graph for this function:

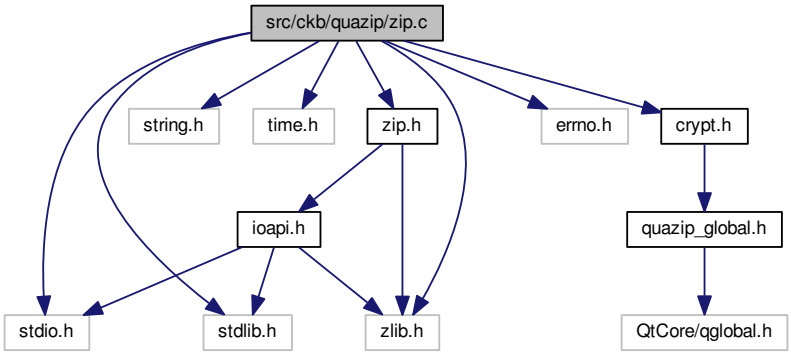


Here is the caller graph for this function:

13.109 src/ckb/quazip/zip.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <time.h>
#include "zlib.h"
#include "zip.h"
#include <errno.h>
#include "crypt.h"
```

Include dependency graph for zip.c:



Data Structures

- struct linkedlist_datablock_internal_s
- struct linkedlist_data_s
- struct curfile64_info
- struct zip64_internal

Macros

- #define local static
- #define VERSIONMADEBY (0x031e) /∗ best for standard pkware crypt ∗/
- #define Z_BUFSIZE (64∗1024) /∗ (16384) ∗/
- #define Z_MAXFILENAMEINZIP (256)
- #define ALLOC(size) (malloc(size))
- #define TRYFREE(p) {if (p) free(p);}
- #define MAKEULONG64(a, b) ((ZPOS64_T)(((unsigned long)(a)) | ((ZPOS64_T)((unsigned long)(b))) << 32))
- #define SEEK_CUR 1
- #define SEEK_END 2
- #define SEEK_SET 0
- #define SIZEDATA_INDATABLOCK (4096-(4∗4))
- #define LOCALHEADERMAGIC (0x04034b50)
- #define DESCRIPTORHEADERMAGIC (0x08074b50)
- #define CENTRALHEADERMAGIC (0x02014b50)
- #define ENDHEADERMAGIC (0x06054b50)
- #define ZIP64ENDHEADERMAGIC (0x6064b50)
- #define ZIP64ENDLOCHEADERMAGIC (0x7064b50)
- #define FLAG_LOCALHEADER_OFFSET (0x06)
- #define CRC_LOCALHEADER_OFFSET (0x0e)
- #define SIZECENTRALHEADER (0x2e) /∗ 46 ∗/
- #define INCLUDECRYPTINGCODE_IFCRYPTALLOWED
- #define BUFREADCOMMENT (0x400)

Typedefs

- typedef uLongf z_crc_t
- typedef struct linkedlist_datablock_internal_s linkedlist_datablock_internal
- typedef struct linkedlist_data_s linkedlist_data

Functions

- static linkedlist_datablock_internal ∗ allocate_new_datablock ()
- static void free_datablock (linkedlist_datablock_internal ∗ldi)
- static void init_linkedlist (linkedlist_data ∗ll)
- static void free_linkedlist (linkedlist_data ∗ll)
- static int add_data_in_datablock (linkedlist_data ∗ll, const void ∗buf, uLong len)
- static int zip64local_putValue _Z_OF ((const zlib_filefunc64_32_def ∗pzlib_filefunc_def, voidpf filestream, ZPOS64_T x, int nbByte))
- static int zip64local_putValue (const zlib_filefunc64_32_def ∗pzlib_filefunc_def, voidpf filestream, ZPOS64_T x, int nbByte)
- static void zip64local_putValue_inmemory _Z_OF ((void ∗dest, ZPOS64_T x, int nbByte))

- static void zip64local_putValue_inmemory (void *dest, ZPOS64_T x, int nbByte)
- static uLong zip64local_TmzDateToDosDate (const tm_zip *ptm)
- static int zip64local_getByte _Z_OF ((const zlib_filefunc64_32_def *pzlib_filefunc_def, voidpf filestream, int *pi))
- static int zip64local_getByte (const zlib_filefunc64_32_def *pzlib_filefunc_def, voidpf filestream, int *pi)
- static int zip64local_getShort _Z_OF ((const zlib_filefunc64_32_def *pzlib_filefunc_def, voidpf filestream, uLong *pX))
- static int zip64local_getShort (const zlib_filefunc64_32_def *pzlib_filefunc_def, voidpf filestream, uLong *pX)
- static int zip64local_getLong (const zlib_filefunc64_32_def *pzlib_filefunc_def, voidpf filestream, uLong *pX)
- static int zip64local_getLong64 _Z_OF ((const zlib_filefunc64_32_def *pzlib_filefunc_def, voidpf filestream, ZPOS64_T *pX))
- static int zip64local_getLong64 (const zlib_filefunc64_32_def *pzlib_filefunc_def, voidpf filestream, ZPOS64_T *pX)
- static ZPOS64_T zip64local_SearchCentralDir _Z_OF ((const zlib_filefunc64_32_def *pzlib_filefunc_def, voidpf filestream))
- static ZPOS64_T zip64local_SearchCentralDir (const zlib_filefunc64_32_def *pzlib_filefunc_def, voidpf filestream)
- static ZPOS64_T zip64local_SearchCentralDir64 (const zlib_filefunc64_32_def *pzlib_filefunc_def, voidpf filestream)
- int LoadCentralDirectoryRecord (zip64_internal *pziinit)
- zipFile ZEXPORT zipOpen3 (voidpf file, int append, zipcharpc *globalcomment, zlib_filefunc64_32_def *pzlib_filefunc64_32_def, unsigned flags)
- zipFile ZEXPORT zipOpen2 (voidpf file, int append, zipcharpc *globalcomment, zlib_filefunc_def *pzlib_filefunc32_def)
- zipFile ZEXPORT zipOpen2_64 (voidpf file, int append, zipcharpc *globalcomment, zlib_filefunc64_def *pzlib_filefunc_def)
- zipFile ZEXPORT zipOpen (voidpf file, int append)
- zipFile ZEXPORT zipOpen64 (voidpf file, int append)
- int Write_LocalFileHeader (zip64_internal *zi, const char *filename, uInt size_extrafield_local, const void *extrafield_local, uLong version_to_extract)
- int ZEXPORT zipOpenNewFileInZip4_64 (zipFile file, const char *filename, const zip_fileinfo *zipfi, const void *extrafield_local, uInt size_extrafield_local, const void *extrafield_global, uInt size_extrafield_global, const char *comment, int method, int level, int raw, int windowBits, int memLevel, int strategy, const char *password, uLong crcForCrypting, uLong versionMadeBy, uLong flagBase, int zip64)
- int ZEXPORT zipOpenNewFileInZip4 (zipFile file, const char *filename, const zip_fileinfo *zipfi, const void *extrafield_local, uInt size_extrafield_local, const void *extrafield_global, uInt size_extrafield_global, const char *comment, int method, int level, int raw, int windowBits, int memLevel, int strategy, const char *password, uLong crcForCrypting, uLong versionMadeBy, uLong flagBase)
- int ZEXPORT zipOpenNewFileInZip3 (zipFile file, const char *filename, const zip_fileinfo *zipfi, const void *extrafield_local, uInt size_extrafield_local, const void *extrafield_global, uInt size_extrafield_global, const char *comment, int method, int level, int raw, int windowBits, int memLevel, int strategy, const char *password, uLong crcForCrypting)
- int ZEXPORT zipOpenNewFileInZip3_64 (zipFile file, const char *filename, const zip_fileinfo *zipfi, const void *extrafield_local, uInt size_extrafield_local, const void *extrafield_global, uInt size_extrafield_global, const char *comment, int method, int level, int raw, int windowBits, int memLevel, int strategy, const char *password, uLong crcForCrypting, int zip64)
- int ZEXPORT zipOpenNewFileInZip2 (zipFile file, const char *filename, const zip_fileinfo *zipfi, const void *extrafield_local, uInt size_extrafield_local, const void *extrafield_global, uInt size_extrafield_global, const char *comment, int method, int level, int raw)
- int ZEXPORT zipOpenNewFileInZip2_64 (zipFile file, const char *filename, const zip_fileinfo *zipfi, const void *extrafield_local, uInt size_extrafield_local, const void *extrafield_global, uInt size_extrafield_global, const char *comment, int method, int level, int raw, int zip64)
- int ZEXPORT zipOpenNewFileInZip64 (zipFile file, const char *filename, const zip_fileinfo *zipfi, const void *extrafield_local, uInt size_extrafield_local, const void *extrafield_global, uInt size_extrafield_global, const char *comment, int method, int level, int zip64)

- int ZEXPORT zipOpenNewFileInZip (zipFile file, const char ∗filename, const zip_fileinfo ∗zipfi, const void ∗extrafield_local, uInt size_extrafield_local, const void ∗extrafield_global, uInt size_extrafield_global, const char ∗comment, int method, int level)
- static int zip64FlushWriteBuffer (zip64_internal ∗zi)
- int ZEXPORT zipWriteInFileInZip (zipFile file, const void ∗buf, unsigned int len)
- int ZEXPORT zipCloseFileInZipRaw (zipFile file, uLong uncompressed_size, uLong crc32)
- int ZEXPORT zipCloseFileInZipRaw64 (zipFile file, ZPOS64_T uncompressed_size, uLong crc32)
- int ZEXPORT zipCloseFileInZip (zipFile file)
- int Write_Zip64EndOfCentralDirectoryLocator (zip64_internal ∗zi, ZPOS64_T zip64eocd_pos_inzip)
- int Write_Zip64EndOfCentralDirectoryRecord (zip64_internal ∗zi, uLong size_centraldir, ZPOS64_T centraldir_pos_inzip)
- int Write_EndOfCentralDirectoryRecord (zip64_internal ∗zi, uLong size_centraldir, ZPOS64_T centraldir_pos_inzip)
- int Write_GlobalComment (zip64_internal ∗zi, const char ∗global_comment)
- int ZEXPORT zipClose (zipFile file, const char ∗global_comment)
- int ZEXPORT zipRemoveExtraInfoBlock (char ∗pData, int ∗dataLen, short sHeader)
- int ZEXPORT zipSetFlags (zipFile file, unsigned flags)
- int ZEXPORT zipClearFlags (zipFile file, unsigned flags)

Variables

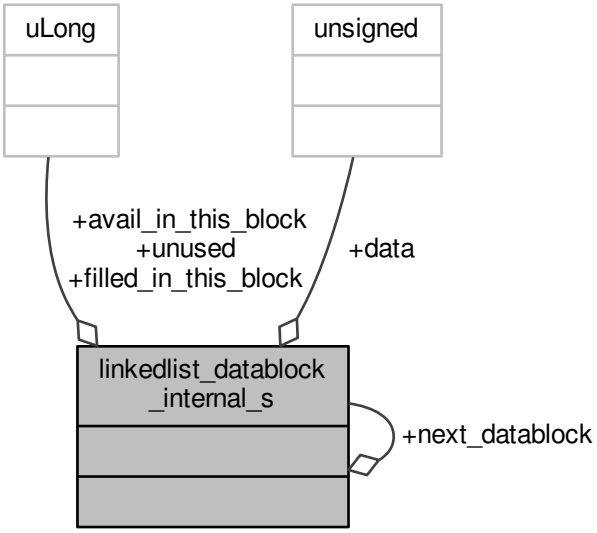
- const char zip_copyright [] =" zip 1.01 Copyright 1998-2004 Gilles Vollant - http://www.winimage.com/zLibDll"

13.109.1 Data Structure Documentation

13.109.1.1 struct linkedlist_datablock_internal_s

Definition at line 121 of file zip.c.

Collaboration diagram for linkedlist_datablock_internal_s:



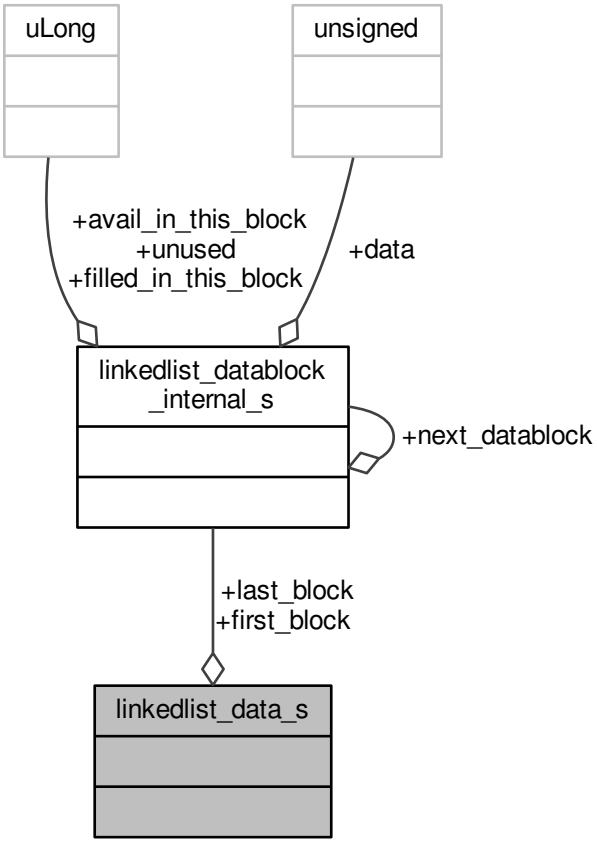
Data Fields

uLong	avail_in_this_-block	
unsigned char	data[(4096-(4 ∗4))]	
uLong	filled_in_this_-block	
struct linkedlist_-datablock_-internal_s ∗	next_datablock	
uLong	unused	

13.109.1.2 struct linkedlist_data_s

Definition at line 130 of file zip.c.

Collaboration diagram for linkedlist_data_s:



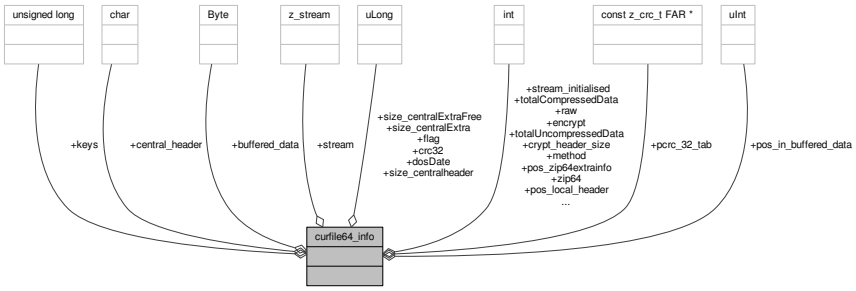
Data Fields

linkedlist_-datablock_-internal *	first_block	
linkedlist_-datablock_-internal *	last_block	

13.109.1.3 struct curfile64_info

Definition at line 137 of file zip.c.

Collaboration diagram for curfile64_info:



Data Fields

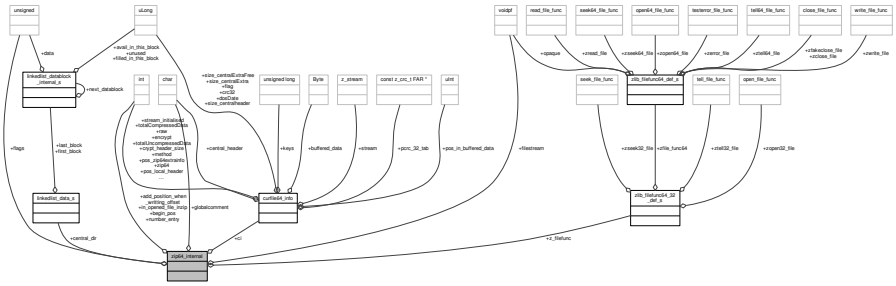
Byte	buffered_-data[(64 ∗1024)]	
char ∗	central_header	
uLong	crc32	
int	crypt_header_-size	
uLong	dosDate	
int	encrypt	
uLong	flag	
unsigned long	keys[3]	
int	method	
const z_crc_t FAR ∗	pcrc_32_tab	
uInt	pos_in_buffered-_data	
ZPOS64_T	pos_local_-header	
ZPOS64_T	pos_-zip64extrainfo	

int	raw	
uLong	size_central-Extra	
uLong	size_central-ExtraFree	
uLong	size_-centralheader	
z_stream	stream	
int	stream_-initialised	
ZPOS64_T	total-Compressed-Data	
ZPOS64_T	total-Uncompressed-Data	
int	zip64	

13.109.1.4 struct zip64_internal

Definition at line 172 of file zip.c.

Collaboration diagram for zip64_internal:



Data Fields

ZPOS64_T	add_position_-when_writting_-offset	
ZPOS64_T	begin_pos	
linkedlist_data	central_dir	
curfile64_info	ci	
voidpf	filestream	
unsigned	flags	
char *	globalcomment	
int	in_opened_file_-inzip	
ZPOS64_T	number_entry	
zlib_filefunc64_-32_def	z_filefunc	

13.109.2 Macro Definition Documentation

13.109.2.1 #define ALLOC(*size*) (malloc(size))

Definition at line 68 of file zip.c.

Referenced by allocate_new_datablock(), LoadCentralDirectoryRecord(), zip64local_SearchCentralDir(), zip64local_SearchCentralDir64(), zipOpen3(), zipOpenNewFileInZip4_64(), and zipRemoveExtraInfoBlock().

13.109.2.2 #define BUFREADCOMMENT (0x400)

Definition at line 481 of file zip.c.

Referenced by zip64local_SearchCentralDir(), and zip64local_SearchCentralDir64().

13.109.2.3 #define CENTRALHEADERMAGIC (0x02014b50)

Definition at line 111 of file zip.c.

Referenced by zipOpenNewFileInZip4_64().

13.109.2.4 #define CRC_LOCALHEADER_OFFSET (0x0e)

Definition at line 117 of file zip.c.

13.109.2.5 #define DESCRIPTORHEADERMAGIC (0x08074b50)

Definition at line 110 of file zip.c.

Referenced by zipCloseFileInZipRaw64().

13.109.2.6 #define ENDHEADERMAGIC (0x06054b50)

Definition at line 112 of file zip.c.

Referenced by Write_EndOfCentralDirectoryRecord().

13.109.2.7 #define FLAG_LOCALHEADER_OFFSET (0x06)

Definition at line 116 of file zip.c.

13.109.2.8 #define INCLUDECRYPTINGCODE_IFCRYPTALLOWED

Definition at line 194 of file zip.c.

13.109.2.9 #define local static

Definition at line 51 of file zip.c.

13.109.2.10 #define LOCALHEADERMAGIC (0x04034b50)

Definition at line 109 of file zip.c.

Referenced by Write_LocalFileHeader().

13.109.2.11 #define MAKEULONG64(a, b) ((ZPOS64_T)(((unsigned long)(a)) | ((ZPOS64_T)((unsigned long)(b))) << 32))

Definition at line 83 of file zip.c.

13.109.2.12 #define SEEK_CUR 1

Definition at line 86 of file zip.c.

13.109.2.13 #define SEEK_END 2

Definition at line 90 of file zip.c.

Referenced by zipOpen3().

13.109.2.14 #define SEEK_SET 0

Definition at line 94 of file zip.c.

13.109.2.15 #define SIZECENTRALHEADER (0x2e) /* 46 */

Definition at line 119 of file zip.c.

Referenced by zipOpenNewFileInZip4_64().

13.109.2.16 #define SIZEDATA_INDATABLOCK (4096-(4*4))

Definition at line 107 of file zip.c.

Referenced by allocate_new_datablock(), and LoadCentralDirectoryRecord().

13.109.2.17 #define TRYFREE(p) {if (p) free(p);}

Definition at line 71 of file zip.c.

Referenced by free_datablock(), LoadCentralDirectoryRecord(), zip64local_SearchCentralDir(), zip64local_SearchCentralDir64(), zipClose(), zipOpen3(), and zipRemoveExtraInfoBlock().

13.109.2.18 #define VERSIONMADEBY (0x031e) /* best for standard pkware crypt */

Definition at line 56 of file zip.c.

Referenced by zipOpenNewFileInZip(), zipOpenNewFileInZip2(), zipOpenNewFileInZip2_64(), zipOpenNewFileInZip3(), zipOpenNewFileInZip3_64(), and zipOpenNewFileInZip64().

13.109.2.19 #define Z_BUFSIZE (64*1024) /* (16384) */

Definition at line 60 of file zip.c.

Referenced by zipCloseFileInZipRaw64(), zipOpenNewFileInZip4_64(), and zipWriteInFileInZip().

13.109.2.20 #define Z_MAXFILENAMEINZIP (256)

Definition at line 64 of file zip.c.

13.109.2.21 #define ZIP64ENDHEADERMAGIC (0x6064b50)

Definition at line 113 of file zip.c.

Referenced by Write_Zip64EndOfCentralDirectoryRecord().

13.109.2.22 #define ZIP64ENDLOCHEADERMAGIC (0x7064b50)

Definition at line 114 of file zip.c.

Referenced by Write_Zip64EndOfCentralDirectoryLocator().

13.109.3 Typedef Documentation

13.109.3.1 typedef struct linkedlist_data_s linkedlist_data

13.109.3.2 typedef struct linkedlist_datablock_internal_s linkedlist_datablock_internal

13.109.3.3 typedef uLongf z_crc_t

Definition at line 34 of file zip.c.

13.109.4 Function Documentation

13.109.4.1 static int zip64local_putValue _Z_OF ((const zlib_filefunc64_32_def *pzlib_filefunc_def, voidpf filestream, ZPOS64_T x, int nbByte)) [static]

13.109.4.2 static void zip64local_putValue_inmemory _Z_OF ((void *dest, ZPOS64_T x, int nbByte)) [static]

13.109.4.3 static int zip64local_getByte _Z_OF ((const zlib_filefunc64_32_def *pzlib_filefunc_def, voidpf filestream, int *pi)) [static]

13.109.4.4 static int zip64local_getShort _Z_OF ((const zlib_filefunc64_32_def *pzlib_filefunc_def, voidpf filestream, uLong *pX)) [static]

13.109.4.5 static int zip64local_getLong64 _Z_OF ((const zlib_filefunc64_32_def *pzlib_filefunc_def, voidpf filestream, ZPOS64_T *pX)) [static]

13.109.4.6 static ZPOS64_T zip64local_SearchCentralDir _Z_OF ((const zlib_filefunc64_32_def *pzlib_filefunc_def, voidpf filestream)) [static]

13.109.4.7 static int add_data_in_datablock (linkedlist_data * *ll*, const void * *buf*, uLong *len*) [static]

Definition at line 234 of file zip.c.

References allocate_new_datablock(), linkedlist_datablock_internal_s::avail_in_this_block, linkedlist_datablock_internal_s::data, linkedlist_datablock_internal_s::filled_in_this_block, linkedlist_data_s::first_block, linkedlist_data_s::last_block, linkedlist_datablock_internal_s::next_datablock, ZIP_INTERNALERROR, and ZIP_OK.

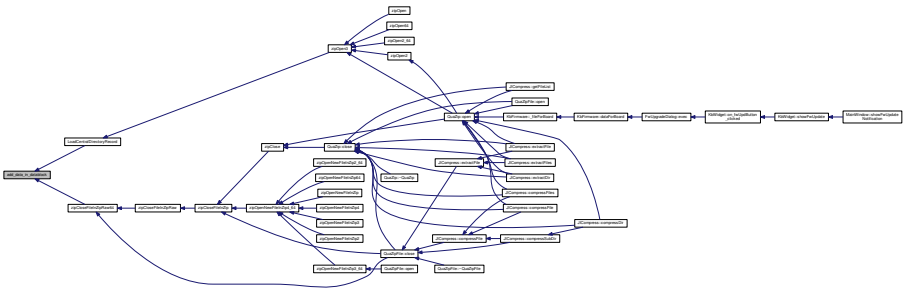
Referenced by LoadCentralDirectoryRecord(), and zipCloseFileInZipRaw64().

```
235 {
236     linkedlist_datablock_internal* ldi;
237     const unsigned char* from_copy;
238
239     if (ll==NULL)
240         return ZIP_INTERNALERROR;
241
242     if (ll->last_block == NULL)
243     {
```

```
244         ll->first_block = ll->last_block =
    allocate_new_datablock();
245         if (ll->first_block == NULL)
246             return ZIP_INTERNALERROR;
247     }
248
249     ldi = ll->last_block;
250     from_copy = (unsigned char*)buf;
251
252     while (len>0)
253     {
254         uInt copy_this;
255         uInt i;
256         unsigned char* to_copy;
257
258         if (ldi->avail_in_this_block==0)
259         {
260             ldi->next_datablock = allocate_new_datablock();
261             if (ldi->next_datablock == NULL)
262                 return ZIP_INTERNALERROR;
263             ldi = ldi->next_datablock ;
264             ll->last_block = ldi;
265         }
266
267         if (ldi->avail_in_this_block < len)
268             copy_this = (uInt)ldi->avail_in_this_block;
269         else
270             copy_this = (uInt)len;
271
272         to_copy = &(ldi->data[ldi->filled_in_this_block]);
273
274         for (i=0;i<copy_this;i++)
275             *(to_copy+i)=*(from_copy+i);
276
277         ldi->filled_in_this_block += copy_this;
278         ldi->avail_in_this_block -= copy_this;
279         from_copy += copy_this ;
280         len -= copy_this;
281     }
282     return ZIP_OK;
283 }
```

Here is the call graph for this function:

Here is the caller graph for this function:



13.109.4.8 static linkedlist_datablock_internal* allocate_new_datablock ( ) [static]

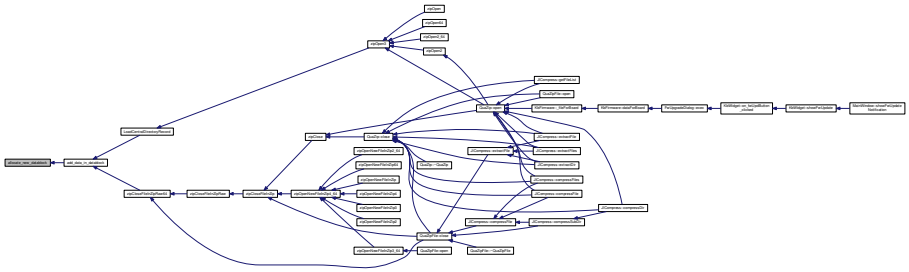
Definition at line 198 of file zip.c.

References ALLOC, linkedlist_datablock_internal_s::avail_in_this_block, linkedlist_datablock_internal_s::filled_in_this_block, linkedlist_datablock_internal_s::next_datablock, and SIZEDATA_INDATABLOCK.

Referenced by add_data_in_datablock().

```
199 {
200     linkedlist_datablock_internal* ldi;
201     ldi = (linkedlist_datablock_internal*)
202                 ALLOC(sizeof(linkedlist_datablock_internal));
203     if (ldi!=NULL)
204     {
205         ldi->next_datablock = NULL ;
206         ldi->filled_in_this_block = 0 ;
207         ldi->avail_in_this_block = SIZEDATA_INDATABLOCK ;
208     }
209     return ldi;
210 }
```

Here is the caller graph for this function:



13.109.4.9 static void free_datablock (linkedlist_datablock_internal * ldi) [static]

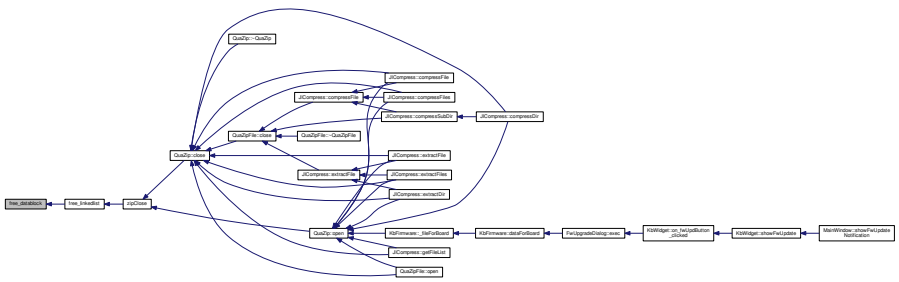
Definition at line 212 of file zip.c.

References linkedlist_datablock_internal_s::next_datablock, and TRYFREE.

Referenced by free_linkedlist().

```
213 {
214     while (ldi!=NULL)
215     {
216         linkedlist_datablock_internal* ldinext = ldi->
    next_datablock;
217         TRYFREE(ldi);
218         ldi = ldinext;
219     }
220 }
```

Here is the caller graph for this function:



13.109.4.10 static void free_linkedlist (linkedlist_data * ll) [static]

Definition at line 227 of file zip.c.

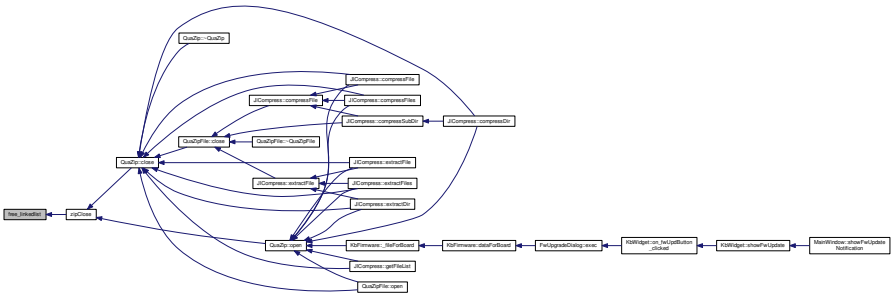
References linkedlist_data_s::first_block, free_datablock(), and linkedlist_data_s::last_block.

Referenced by zipClose().

```
228 {
229     free_datablock(ll->first_block);
230     ll->first_block = ll->last_block = NULL;
231 }
```

Here is the call graph for this function:

Here is the caller graph for this function:



13.109.4.11 static void init_linkedlist (linkedlist_data * ll) [static]

Definition at line 222 of file zip.c.

References linkedlist_data_s::first_block, and linkedlist_data_s::last_block.

Referenced by zipOpen3().

```
223 {
224     ll->first_block = ll->last_block = NULL;
225 }
```

Here is the caller graph for this function:

13.109.4.12 int LoadCentralDirectoryRecord (zip64_internal * *pziinit*)

Definition at line 649 of file zip.c.

References add_data_in_datablock(), zip64_internal::add_position_when_writting_offset, ALLOC, zip64_internal::begin_pos, zip64_internal::central_dir, zip64_internal::filestream, zip64_internal::flags, zip64_internal::globalcomment, zip64_internal::number_entry, SIZEDATA_INDATABLOCK, TRYFREE, zip64_internal::z_filefunc, ZCLOSE64, ZFAKECLOSE64, zip64local_getLong(), zip64local_getLong64(), zip64local_getShort(), zip64local_SearchCentralDir(), zip64local_SearchCentralDir64(), ZIP_AUTO_CLOSE, ZIP_BADZIPFILE, ZIP_ERRNO, ZIP_OK, ZLIB_FILEFUNC_SEEK_SET, ZREAD64, and ZSEEK64.

Referenced by zipOpen3().

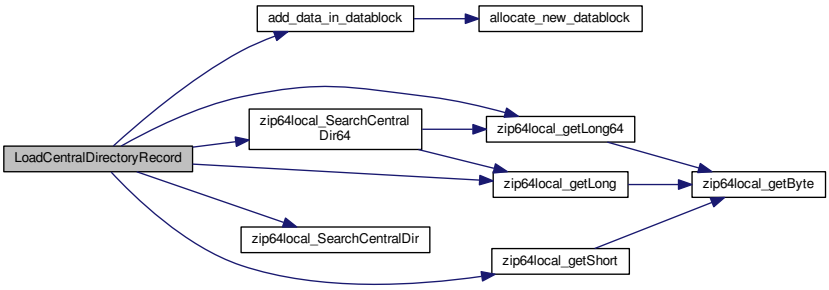
```
650 {
651   int err=ZIP_OK;
652   ZPOS64_T byte_before_the_zipfile;/* byte before the zipfile, (>0 for sfx)*/
653
654   ZPOS64_T size_central_dir;     /* size of the central directory  */
655   ZPOS64_T offset_central_dir;   /* offset of start of central directory */
656   ZPOS64_T central_pos;
657   uLong uL;
658
659   uLong number_disk;          /* number of the current dist, used for
660                                  spaning ZIP, unsupported, always 0*/
661   uLong number_disk_with_CD;  /* number the the disk with central dir, used
662                                  for spaning ZIP, unsupported, always 0*/
663   ZPOS64_T number_entry;
664   ZPOS64_T number_entry_CD;      /* total number of entries in
665                                    the central dir
666                                    (same than number_entry on nospan) */
667   uLong VersionMadeBy;
668   uLong VersionNeeded;
669   uLong size_comment;
670
671   int hasZIP64Record = 0;
672
673   /* check first if we find a ZIP64 record */
674   central_pos = zip64local_SearchCentralDir64(&pziinit->
      z_filefunc,pziinit->filestream);
675   if(central_pos > 0)
676   {
677     hasZIP64Record = 1;
678   }
679   else if(central_pos == 0)
680   {
681     central_pos = zip64local_SearchCentralDir(&pziinit->
      z_filefunc,pziinit->filestream);
682   }
683
684 /* disable to allow appending to empty ZIP archive
685         if (central_pos==0)
686             err=ZIP_ERRNO;
687 */
688
689   if(hasZIP64Record)
690   {
691     ZPOS64_T sizeEndOfCentralDirectory;
692     if (ZSEEK64(pziinit->z_filefunc, pziinit->filestream, central_pos,
      ZLIB_FILEFUNC_SEEK_SET) != 0)
693       err=ZIP_ERRNO;
694
695     /* the signature, already checked */
```

```
696     if (zip64local_getLong(&pziinit->z_filefunc, pziinit->
      filestream,&uL)!=ZIP_OK)
697       err=ZIP_ERRNO;
698
699     /* size of zip64 end of central directory record */
700     if (zip64local_getLong64(&pziinit->z_filefunc, pziinit->
      filestream, &sizeEndOfCentralDirectory)!=ZIP_OK)
701       err=ZIP_ERRNO;
702
703     /* version made by */
704     if (zip64local_getShort(&pziinit->z_filefunc, pziinit->
      filestream, &VersionMadeBy)!=ZIP_OK)
705       err=ZIP_ERRNO;
706
707     /* version needed to extract */
708     if (zip64local_getShort(&pziinit->z_filefunc, pziinit->
      filestream, &VersionNeeded)!=ZIP_OK)
709       err=ZIP_ERRNO;
710
711     /* number of this disk */
712     if (zip64local_getLong(&pziinit->z_filefunc, pziinit->
      filestream,&number_disk)!=ZIP_OK)
713       err=ZIP_ERRNO;
714
715     /* number of the disk with the start of the central directory */
716     if (zip64local_getLong(&pziinit->z_filefunc, pziinit->
      filestream,&number_disk_with_CD)!=ZIP_OK)
717       err=ZIP_ERRNO;
718
719     /* total number of entries in the central directory on this disk */
720     if (zip64local_getLong64(&pziinit->z_filefunc, pziinit->
      filestream, &number_entry)!=ZIP_OK)
721       err=ZIP_ERRNO;
722
723     /* total number of entries in the central directory */
724     if (zip64local_getLong64(&pziinit->z_filefunc, pziinit->
      filestream,&number_entry_CD)!=ZIP_OK)
725       err=ZIP_ERRNO;
726
727     if ((number_entry_CD!=number_entry) || (number_disk_with_CD!=0) || (number_disk!=0))
728       err=ZIP_BADZIPFILE;
729
730     /* size of the central directory */
731     if (zip64local_getLong64(&pziinit->z_filefunc, pziinit->
      filestream,&size_central_dir)!=ZIP_OK)
732       err=ZIP_ERRNO;
733
734     /* offset of start of central directory with respect to the
735     starting disk number */
736     if (zip64local_getLong64(&pziinit->z_filefunc, pziinit->
      filestream,&offset_central_dir)!=ZIP_OK)
737       err=ZIP_ERRNO;
738
739     /* TODO.. */
740     /* read the comment from the standard central header. */
741     size_comment = 0;
742   }
743   else
744   {
745     /* Read End of central Directory info */
746     if (ZSEEK64(pziinit->z_filefunc, pziinit->filestream, central_pos,
      ZLIB_FILEFUNC_SEEK_SET)!=0)
747       err=ZIP_ERRNO;
748
749     /* the signature, already checked */
750     if (zip64local_getLong(&pziinit->z_filefunc, pziinit->
      filestream,&uL)!=ZIP_OK)
751       err=ZIP_ERRNO;
752
753     /* number of this disk */
754     if (zip64local_getShort(&pziinit->z_filefunc, pziinit->
      filestream,&number_disk)!=ZIP_OK)
755       err=ZIP_ERRNO;
756
757     /* number of the disk with the start of the central directory */
758     if (zip64local_getShort(&pziinit->z_filefunc, pziinit->
      filestream,&number_disk_with_CD)!=ZIP_OK)
759       err=ZIP_ERRNO;
760
761     /* total number of entries in the central dir on this disk */
762     number_entry = 0;
763     if (zip64local_getShort(&pziinit->z_filefunc, pziinit->
      filestream, &uL)!=ZIP_OK)
764       err=ZIP_ERRNO;
765     else
766       number_entry = uL;
767
```

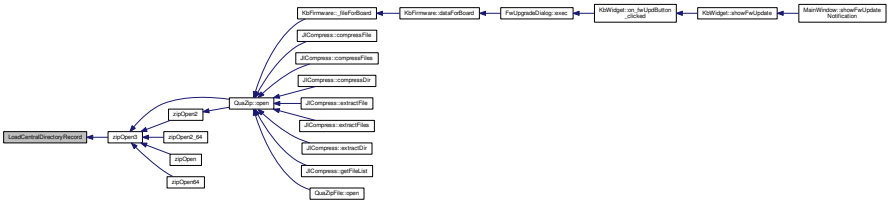
```
768     /* total number of entries in the central dir */
769     number_entry_CD = 0;
770     if (zip64local_getShort(&pziinit->z_filefunc, pziinit->
    filestream, &uL)!=ZIP_OK)
771       err=ZIP_ERRNO;
772     else
773       number_entry_CD = uL;
774
775     if ((number_entry_CD!=number_entry) || (number_disk_with_CD!=0) || (number_disk!=0))
776       err=ZIP_BADZIPFILE;
777
778     /* size of the central directory */
779     size_central_dir = 0;
780     if (zip64local_getLong(&pziinit->z_filefunc, pziinit->
    filestream, &uL)!=ZIP_OK)
781       err=ZIP_ERRNO;
782     else
783       size_central_dir = uL;
784
785     /* offset of start of central directory with respect to the starting disk number */
786     offset_central_dir = 0;
787     if (zip64local_getLong(&pziinit->z_filefunc, pziinit->
    filestream, &uL)!=ZIP_OK)
788       err=ZIP_ERRNO;
789     else
790       offset_central_dir = uL;
791
792
793     /* zipfile global comment length */
794     if (zip64local_getShort(&pziinit->z_filefunc, pziinit->
    filestream, &size_comment)!=ZIP_OK)
795       err=ZIP_ERRNO;
796   }
797
798   if ((central_pos<offset_central_dir+size_central_dir) &&
799     (err==ZIP_OK))
800     err=ZIP_BADZIPFILE;
801
802   if (err!=ZIP_OK)
803   {
804       if ((pziinit->flags & ZIP_AUTO_CLOSE) != 0) {
805         ZCLOSE64(pziinit->z_filefunc, pziinit->filestream);
806       } else {
807         ZFAKECLOSE64(pziinit->z_filefunc, pziinit->
    filestream);
808       }
809     return ZIP_ERRNO;
810   }
811
812   if (size_comment>0)
813   {
814     pziinit->globalcomment = (char*)ALLOC(size_comment+1);
815     if (pziinit->globalcomment)
816     {
817       size_comment = ZREAD64(pziinit->z_filefunc, pziinit->
    filestream, pziinit->globalcomment,size_comment);
818       pziinit->globalcomment[size_comment]=0;
819     }
820   }
821
822   byte_before_the_zipfile = central_pos - (offset_central_dir+size_central_dir);
823   pziinit->add_position_when_writting_offset = byte_before_the_zipfile;
824
825   {
826     ZPOS64_T size_central_dir_to_read = size_central_dir;
827     size_t buf_size = SIZEDATA_INDATABLOCK;
828     void* buf_read = (void*)ALLOC(buf_size);
829     if (ZSEEK64(pziinit->z_filefunc, pziinit->filestream, offset_central_dir +
    byte_before_the_zipfile, ZLIB_FILEFUNC_SEEK_SET) != 0)
830       err=ZIP_ERRNO;
831
832     while ((size_central_dir_to_read>0) && (err==ZIP_OK))
833     {
834       ZPOS64_T read_this = SIZEDATA_INDATABLOCK;
835       if (read_this > size_central_dir_to_read)
836         read_this = size_central_dir_to_read;
837
838       if (ZREAD64(pziinit->z_filefunc, pziinit->filestream,buf_read,(uLong)
    read_this) != read_this)
839         err=ZIP_ERRNO;
840
841       if (err==ZIP_OK)
842         err = add_data_in_datablock(&pziinit->central_dir,buf_read, (uLong)
    read_this);
843
844       size_central_dir_to_read-=read_this;
845     }
```

```
846     TRYFREE(buf_read);
847   }
848   pziinit->begin_pos = byte_before_the_zipfile;
849   pziinit->number_entry = number_entry_CD;
850
851   if (ZSEEK64(pziinit->z_filefunc, pziinit->filestream, offset_central_dir+
      byte_before_the_zipfile,ZLIB_FILEFUNC_SEEK_SET) != 0)
852     err=ZIP_ERRNO;
853
854   return err;
855 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



13.109.4.13 int Write_EndOfCentralDirectoryRecord (zip64_internal * *zi*, uLong *size_centraldir*, ZPOS64_T *centraldir_pos_inzip*)

Definition at line 1876 of file zip.c.

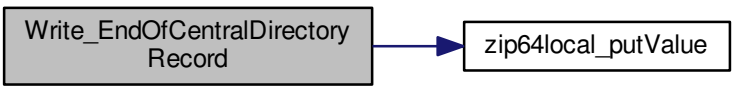
References zip64_internal::add_position_when_writting_offset, ENDHEADERMAGIC, zip64_internal::filestream, zip64_internal::number_entry, zip64_internal::z_filefunc, zip64local_putValue(), and ZIP_OK.

Referenced by zipClose().

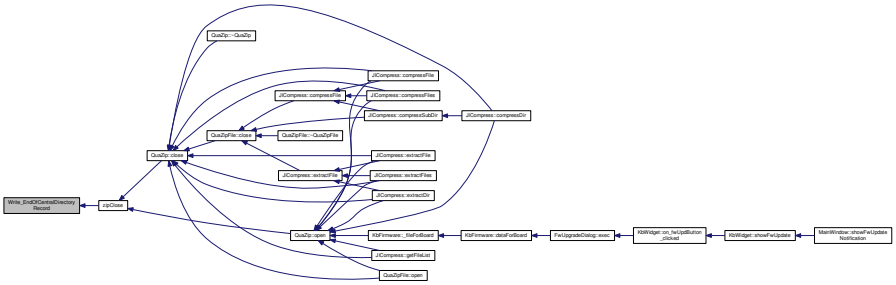
```
1877 {
1878   int err = ZIP_OK;
1879
1880   /*signature*/
1881   err = zip64local_putValue(&zi->z_filefunc,zi->
      filestream,(uLong)ENDHEADERMAGIC,4);
1882
1883   if (err==ZIP_OK) /* number of this disk */
1884     err = zip64local_putValue(&zi->z_filefunc,zi->
      filestream,(uLong)0,2);
1885
1886   if (err==ZIP_OK) /* number of the disk with the start of the central directory */
1887     err = zip64local_putValue(&zi->z_filefunc,zi->
      filestream,(uLong)0,2);
1888
1889   if (err==ZIP_OK) /* total number of entries in the central dir on this disk */
```

```
1890   {
1891     {
1892       if(zi->number_entry >= 0xFFFF)
1893         err = zip64local_putValue(&zi->z_filefunc,zi->
      filestream,(uLong)0xffff,2); /* use value in ZIP64 record */
1894       else
1895         err = zip64local_putValue(&zi->z_filefunc,zi->
      filestream,(uLong)zi->number_entry,2);
1896     }
1897   }
1898
1899   if (err==ZIP_OK) /* total number of entries in the central dir */
1900   {
1901     if(zi->number_entry >= 0xFFFF)
1902       err = zip64local_putValue(&zi->z_filefunc,zi->
      filestream,(uLong)0xffff,2); /* use value in ZIP64 record */
1903     else
1904       err = zip64local_putValue(&zi->z_filefunc,zi->
      filestream,(uLong)zi->number_entry,2);
1905   }
1906
1907   if (err==ZIP_OK) /* size of the central directory */
1908     err = zip64local_putValue(&zi->z_filefunc,zi->
      filestream,(uLong)size_centraldir,4);
1909
1910   if (err==ZIP_OK) /* offset of start of central directory with respect to the starting disk number
      */
1911   {
1912     ZPOS64_T pos = centraldir_pos_inzip - zi->
      add_position_when_writting_offset;
1913     if(pos >= 0xffffffff)
1914     {
1915       err = zip64local_putValue(&zi->z_filefunc,zi->
      filestream, (uLong)0xffffffff,4);
1916     }
1917     else
1918                   err = zip64local_putValue(&zi->z_filefunc,zi->
      filestream, (uLong)(centraldir_pos_inzip - zi->
      add_position_when_writting_offset),4);
1919   }
1920
1921    return err;
1922 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



13.109.4.14 int Write_GlobalComment (zip64_internal * *zi,* const char * *global_comment*)

Definition at line 1924 of file zip.c.

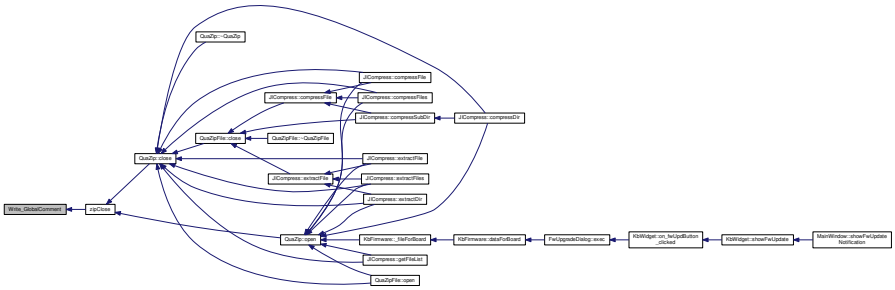
References zip64_internal::filestream, zip64_internal::z_filefunc, zip64local_putValue(), ZIP_ERRNO, ZIP_OK, and ZWRITE64.

Referenced by zipClose().

```
1925 {
1926   int err = ZIP_OK;
1927   uInt size_global_comment = 0;
1928
1929   if(global_comment != NULL)
1930     size_global_comment = (uInt)strlen(global_comment);
1931
1932   err = zip64local_putValue(&zi->z_filefunc,zi->
      filestream,(uLong)size_global_comment,2);
1933
1934   if (err == ZIP_OK && size_global_comment > 0)
1935   {
1936     if (ZWRITE64(zi->z_filefunc,zi->filestream, global_comment,
      size_global_comment) != size_global_comment)
1937       err = ZIP_ERRNO;
1938   }
1939   return err;
1940 }
```

Here is the call graph for this function:

Here is the caller graph for this function:



13.109.4.15 int Write_LocalFileHeader (zip64_internal * *zi,* const char * *filename,* uInt *size_extrafield_local,* const void * *extrafield_local,* uLong *version_to_extract*)

Definition at line 977 of file zip.c.

References zip64_internal::ci, curfile64_info::dosDate, zip64_internal::filestream, curfile64_info::flag, LOCALHEADERMAGIC, curfile64_info::method, curfile64_info::pos_zip64extrainfo, zip64_internal::z_filefunc, curfile64_info::zip64, zip64local_putValue(), ZIP_ERRNO, ZIP_OK, ZTELL64, and ZWRITE64.

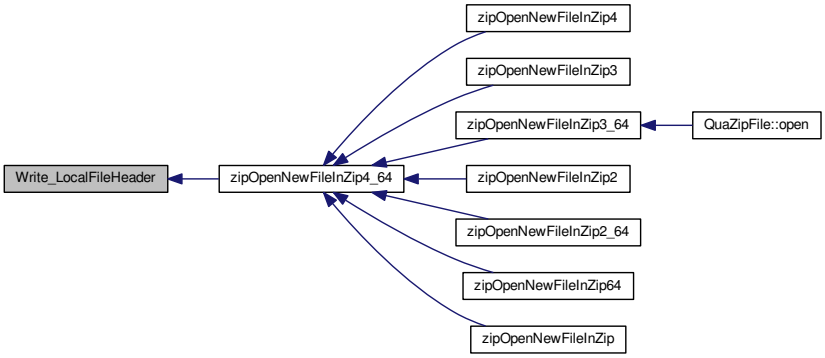
Referenced by zipOpenNewFileInZip4_64().

```
981 {
982   /* write the local header */
983   int err;
984   uInt size_filename = (uInt)strlen(filename);
985   uInt size_extrafield = size_extrafield_local;
986 
987   err = zip64local_putValue(&zi->z_filefunc,zi->
      filestream,(uLong)LOCALHEADERMAGIC, 4);
988 
989   if (err==ZIP_OK)
990   {
991     if(zi->ci.zip64)
992       err = zip64local_putValue(&zi->z_filefunc,zi->
      filestream,(uLong)45,2);/* version needed to extract */
993     else
994       err = zip64local_putValue(&zi->z_filefunc,zi->
      filestream,(uLong)version_to_extract,2);
995   }
996 
997   if (err==ZIP_OK)
998     err = zip64local_putValue(&zi->z_filefunc,zi->
      filestream,(uLong)zi->ci.flag,2);
999 
1000   if (err==ZIP_OK)
1001     err = zip64local_putValue(&zi->z_filefunc,zi->
      filestream,(uLong)zi->ci.method,2);
1002 
1003   if (err==ZIP_OK)
1004     err = zip64local_putValue(&zi->z_filefunc,zi->
      filestream,(uLong)zi->ci.dosDate,4);
1005 
1006   /* CRC / Compressed size / Uncompressed size will be filled in later and rewritten later */
1007   if (err==ZIP_OK)
1008     err = zip64local_putValue(&zi->z_filefunc,zi->
      filestream,(uLong)0,4); /* crc 32, unknown */
1009   if (err==ZIP_OK)
1010   {
1011     if(zi->ci.zip64)
1012       err = zip64local_putValue(&zi->z_filefunc,zi->
      filestream,(uLong)0xFFFFFFFF,4); /* compressed size, unknown */
1013     else
1014       err = zip64local_putValue(&zi->z_filefunc,zi->
      filestream,(uLong)0,4); /* compressed size, unknown */
1015   }
1016   if (err==ZIP_OK)
1017   {
1018     if(zi->ci.zip64)
1019       err = zip64local_putValue(&zi->z_filefunc,zi->
      filestream,(uLong)0xFFFFFFFF,4); /* uncompressed size, unknown */
1020     else
1021       err = zip64local_putValue(&zi->z_filefunc,zi->
      filestream,(uLong)0,4); /* uncompressed size, unknown */
1022   }
1023 
1024   if (err==ZIP_OK)
1025     err = zip64local_putValue(&zi->z_filefunc,zi->
      filestream,(uLong)size_filename,2);
1026 
1027   if(zi->ci.zip64)
1028   {
1029     size_extrafield += 20;
1030   }
1031 
1032   if (err==ZIP_OK)
1033     err = zip64local_putValue(&zi->z_filefunc,zi->
      filestream,(uLong)size_extrafield,2);
1034 
1035   if ((err==ZIP_OK) && (size_filename > 0))
1036   {
1037     if (ZWRITE64(zi->z_filefunc,zi->filestream,filename,size_filename)!=
      size_filename)
1038       err = ZIP_ERRNO;
1039   }
1040 
1041   if ((err==ZIP_OK) && (size_extrafield_local > 0))
1042   {
1043     if (ZWRITE64(zi->z_filefunc, zi->filestream, extrafield_local,
      size_extrafield_local) != size_extrafield_local)
1044       err = ZIP_ERRNO;
1045   }
1046 
1047 
1048   if ((err==ZIP_OK) && (zi->ci.zip64))
1049   {
1050       /* write the Zip64 extended info */
1051       short HeaderID = 1;
1052       short DataSize = 16;
```

```
1053       ZPOS64_T CompressedSize = 0;
1054       ZPOS64_T UncompressedSize = 0;
1055
1056       /* Remember position of Zip64 extended info for the local file header. (needed when we update size
     after done with file) */
1057       zi->ci.pos_zip64extrainfo = ZTELL64(zi->
    z_filefunc,zi->filestream);
1058
1059       err = zip64local_putValue(&zi->z_filefunc, zi->
    filestream, (short)HeaderID,2);
1060       err = zip64local_putValue(&zi->z_filefunc, zi->
    filestream, (short)DataSize,2);
1061
1062       err = zip64local_putValue(&zi->z_filefunc, zi->
    filestream, (ZPOS64_T)UncompressedSize,8);
1063       err = zip64local_putValue(&zi->z_filefunc, zi->
    filestream, (ZPOS64_T)CompressedSize,8);
1064     }
1065
1066     return err;
1067 }
```

Here is the call graph for this function:

Here is the caller graph for this function:



13.109.4.16 int Write_Zip64EndOfCentralDirectoryLocator (zip64_internal * *zi*, ZPOS64_T *zip64eocd_pos_inzip*)

Definition at line 1815 of file zip.c.

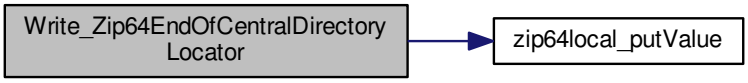
References zip64_internal::add_position_when_writting_offset, zip64_internal::filestream, zip64_internal::z_filefunc, ZIP64ENDLOCHEADERMAGIC, zip64local_putValue(), and ZIP_OK.

Referenced by zipClose().

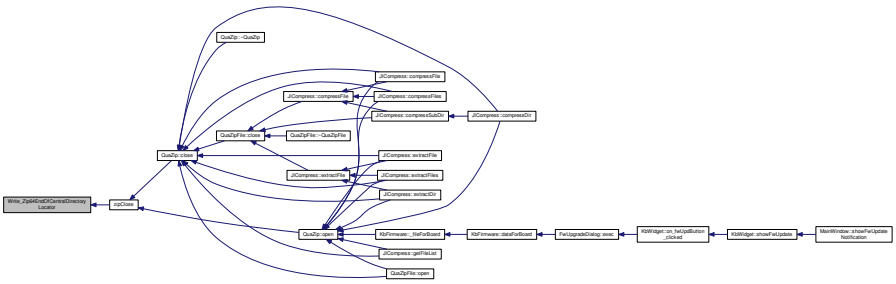
```
1816 {
1817   int err = ZIP_OK;
```

```
1818   ZPOS64_T pos = zip64eocd_pos_inzip - zi->
      add_position_when_writting_offset;
1819
1820   err = zip64local_putValue(&zi->z_filefunc,zi->
      filestream,(uLong)ZIP64ENDLOCHEADERMAGIC,4);
1821
1822   /*num disks*/
1823     if (err==ZIP_OK) /* number of the disk with the start of the central directory */
1824       err = zip64local_putValue(&zi->z_filefunc,zi->
      filestream,(uLong)0,4);
1825
1826   /*relative offset*/
1827     if (err==ZIP_OK) /* Relative offset to the Zip64EndOfCentralDirectory */
1828       err = zip64local_putValue(&zi->z_filefunc,zi->
      filestream, pos,8);
1829
1830   /*total disks*/ /* Do not support spawning of disk so always say 1 here*/
1831     if (err==ZIP_OK) /* number of the disk with the start of the central directory */
1832       err = zip64local_putValue(&zi->z_filefunc,zi->
      filestream,(uLong)1,4);
1833
1834     return err;
1835 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



13.109.4.17 int Write_Zip64EndOfCentralDirectoryRecord (zip64_internal * *zi*, uLong *size_centraldir*, ZPOS64_T *centraldir_pos_inzip*)

Definition at line 1837 of file zip.c.

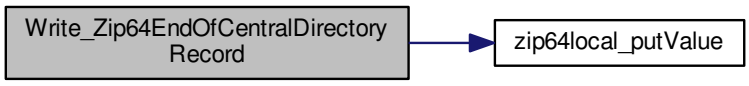
References zip64_internal::add_position_when_writting_offset, zip64_internal::filestream, zip64_internal::number-_entry, zip64_internal::z_filefunc, ZIP64ENDHEADERMAGIC, zip64local_putValue(), and ZIP_OK.

Referenced by zipClose().

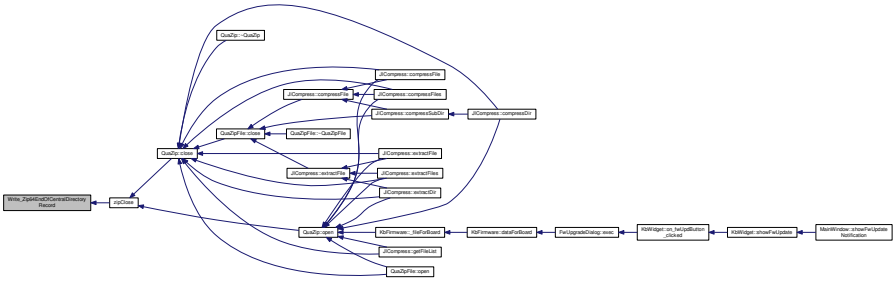
```
1838 {
1839   int err = ZIP_OK;
1840
1841   uLong Zip64DataSize = 44;
1842
1843   err = zip64local_putValue(&zi->z_filefunc,zi->
```

```
    filestream,(uLong)ZIP64ENDHEADERMAGIC,4);
1844
1845   if (err==ZIP_OK) /* size of this 'zip64 end of central directory' */
1846     err = zip64local_putValue(&zi->z_filefunc,zi->
    filestream,(ZPOS64_T)Zip64DataSize,8); /* why ZPOS64_T of this ? */
1847
1848   if (err==ZIP_OK) /* version made by */
1849     err = zip64local_putValue(&zi->z_filefunc,zi->
    filestream,(uLong)45,2);
1850
1851   if (err==ZIP_OK) /* version needed */
1852     err = zip64local_putValue(&zi->z_filefunc,zi->
    filestream,(uLong)45,2);
1853
1854   if (err==ZIP_OK) /* number of this disk */
1855     err = zip64local_putValue(&zi->z_filefunc,zi->
    filestream,(uLong)0,4);
1856
1857   if (err==ZIP_OK) /* number of the disk with the start of the central directory */
1858     err = zip64local_putValue(&zi->z_filefunc,zi->
    filestream,(uLong)0,4);
1859
1860   if (err==ZIP_OK) /* total number of entries in the central dir on this disk */
1861     err = zip64local_putValue(&zi->z_filefunc, zi->
    filestream, zi->number_entry, 8);
1862
1863   if (err==ZIP_OK) /* total number of entries in the central dir */
1864     err = zip64local_putValue(&zi->z_filefunc, zi->
    filestream, zi->number_entry, 8);
1865
1866   if (err==ZIP_OK) /* size of the central directory */
1867     err = zip64local_putValue(&zi->z_filefunc,zi->
    filestream,(ZPOS64_T)size_centraldir,8);
1868
1869   if (err==ZIP_OK) /* offset of start of central directory with respect to the starting disk number
    */
1870   {
1871     ZPOS64_T pos = centraldir_pos_inzip - zi->
    add_position_when_writting_offset;
1872     err = zip64local_putValue(&zi->z_filefunc,zi->
    filestream, (ZPOS64_T)pos,8);
1873   }
1874   return err;
1875 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



13.109.4.18 static int zip64FlushWriteBuffer (zip64_internal * *zi*) [static]

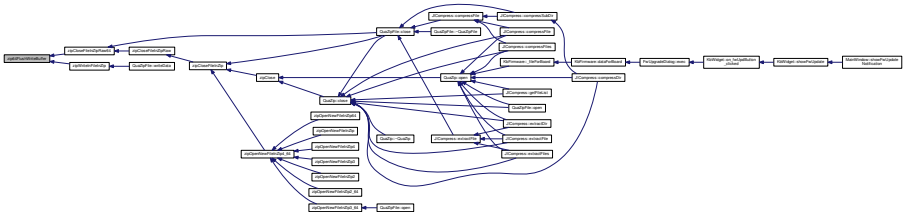
Definition at line 1403 of file zip.c.

References curfile64_info::buffered_data, zip64_internal::ci, curfile64_info::encrypt, zip64_internal::filestream, curfile64_info::keys, curfile64_info::method, curfile64_info::pcrc_32_tab, curfile64_info::pos_in_buffered_data, curfile64_info::stream, curfile64_info::totalCompressedData, curfile64_info::totalUncompressedData, Z_BZIP2ED, zip64_internal::z_filefunc, zencode, ZIP_ERRNO, ZIP_OK, and ZWRITE64.

Referenced by zipCloseFileInZipRaw64(), and zipWriteInFileInZip().

```
1404 {
1405     int err=ZIP_OK;
1406
1407     if (zi->ci.encrypt != 0)
1408     {
1409 #ifndef NOCRYPT
1410         uInt i;
1411         int t;
1412         for (i=0;i<zi->ci.pos_in_buffered_data;i++)
1413             zi->ci.buffered_data[i] = zencode(zi->ci.
      keys, zi->ci.pcrc_32_tab, zi->ci.buffered_data[i],t);
1414 #endif
1415     }
1416
1417     if (ZWRITE64(zi->z_filefunc,zi->filestream,zi->
      ci.buffered_data,zi->ci.pos_in_buffered_data) != zi->
      ci.pos_in_buffered_data)
1418       err = ZIP_ERRNO;
1419
1420     zi->ci.totalCompressedData += zi->ci.
      pos_in_buffered_data;
1421
1422 #ifdef HAVE_BZIP2
1423     if(zi->ci.method == Z_BZIP2ED)
1424     {
1425       zi->ci.totalUncompressedData += zi->ci.bstream.total_in_lo32;
1426       zi->ci.bstream.total_in_lo32 = 0;
1427       zi->ci.bstream.total_in_hi32 = 0;
1428     }
1429     else
1430 #endif
1431     {
1432       zi->ci.totalUncompressedData += zi->ci.stream.total_in;
1433       zi->ci.stream.total_in = 0;
1434     }
1435
1436
1437     zi->ci.pos_in_buffered_data = 0;
1438
1439     return err;
1440 }
```

Here is the caller graph for this function:



13.109.4.19 static int zip64local_getByte (const zlib_filefunc64_32_def * *pzlib_filefunc_def,* voidpf *filestream,* int * *pi*) [static]

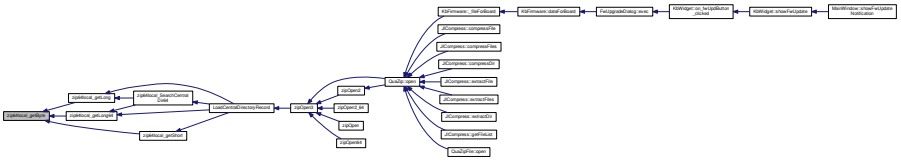
Definition at line 358 of file zip.c.

References ZERROR64, ZIP_EOF, ZIP_ERRNO, ZIP_OK, and ZREAD64.

Referenced by zip64local_getLong(), zip64local_getLong64(), and zip64local_getShort().

```
359 {
360     unsigned char c;
361     int err = (int)ZREAD64(*pzlib_filefunc_def,filestream,&c,1);
362     if (err==1)
363     {
364         *pi = (int)c;
365         return ZIP_OK;
366     }
367     else
368     {
369         if (ZERROR64(*pzlib_filefunc_def,filestream))
370             return ZIP_ERRNO;
371         else
372             return ZIP_EOF;
373     }
374 }
```

Here is the caller graph for this function:



13.109.4.20 static int zip64local_getLong (const zlib_filefunc64_32_def * *pzlib_filefunc_def,* voidpf *filestream,* uLong * *pX*) [static]

Definition at line 404 of file zip.c.

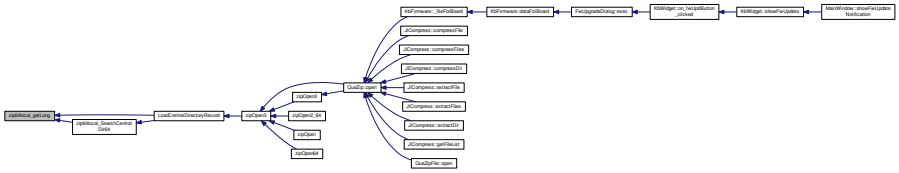
References zip64local_getByte(), and ZIP_OK.

Referenced by LoadCentralDirectoryRecord(), and zip64local_SearchCentralDir64().

```
405 {
406     uLong x ;
407     int i = 0;
408     int err;
409
410     err = zip64local_getByte(pzlib_filefunc_def,filestream,&i);
411     x = (uLong)i;
412
413     if (err==ZIP_OK)
414         err = zip64local_getByte(pzlib_filefunc_def,filestream,&i);
415     x += ((uLong)i)<<8;
416
417     if (err==ZIP_OK)
418         err = zip64local_getByte(pzlib_filefunc_def,filestream,&i);
419     x += ((uLong)i)<<16;
420
421     if (err==ZIP_OK)
422         err = zip64local_getByte(pzlib_filefunc_def,filestream,&i);
423     x += ((uLong)i)<<24;
424
425     if (err==ZIP_OK)
426         *pX = x;
427     else
428         *pX = 0;
429     return err;
430 }
```

Here is the call graph for this function:

Here is the caller graph for this function:



13.109.4.21 static int zip64local_getLong64 (const zlib_filefunc64_32_def * *pzlib_filefunc_def,* voidpf *filestream,* ZPOS64_T * *pX*) [static]

Definition at line 435 of file zip.c.

References zip64local_getByte(), and ZIP_OK.

Referenced by LoadCentralDirectoryRecord(), and zip64local_SearchCentralDir64().

```
436 {
437   ZPOS64_T x;
438   int i = 0;
439   int err;
440
441   err = zip64local_getByte(pzlib_filefunc_def,filestream,&i);
442   x = (ZPOS64_T)i;
443
444   if (err==ZIP_OK)
445     err = zip64local_getByte(pzlib_filefunc_def,filestream,&i);
446   x += ((ZPOS64_T)i)<<8;
447
448   if (err==ZIP_OK)
449     err = zip64local_getByte(pzlib_filefunc_def,filestream,&i);
450   x += ((ZPOS64_T)i)<<16;
451
452   if (err==ZIP_OK)
453     err = zip64local_getByte(pzlib_filefunc_def,filestream,&i);
454   x += ((ZPOS64_T)i)<<24;
455
456   if (err==ZIP_OK)
457     err = zip64local_getByte(pzlib_filefunc_def,filestream,&i);
458   x += ((ZPOS64_T)i)<<32;
459
460   if (err==ZIP_OK)
461     err = zip64local_getByte(pzlib_filefunc_def,filestream,&i);
462   x += ((ZPOS64_T)i)<<40;
463
464   if (err==ZIP_OK)
465     err = zip64local_getByte(pzlib_filefunc_def,filestream,&i);
466   x += ((ZPOS64_T)i)<<48;
467
468   if (err==ZIP_OK)
469     err = zip64local_getByte(pzlib_filefunc_def,filestream,&i);
470   x += ((ZPOS64_T)i)<<56;
471
472   if (err==ZIP_OK)
473     *pX = x;
```

```
474   else
475     *pX = 0;
476
477   return err;
478 }
```

Here is the call graph for this function:

Here is the caller graph for this function:

13.109.4.22 static int zip64local_getShort (const zlib_filefunc64_32_def * *pzlib_filefunc_def,* voidpf *filestream,* uLong * *pX*) [static]

Definition at line 382 of file zip.c.

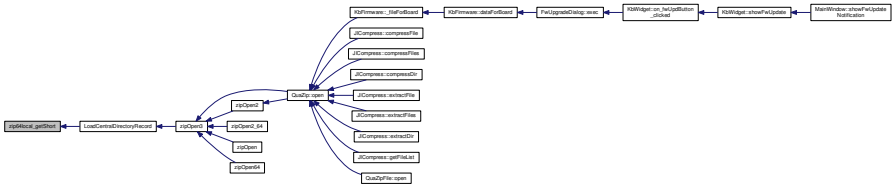
References zip64local_getByte(), and ZIP_OK.

Referenced by LoadCentralDirectoryRecord().

```
383 {
384     uLong x ;
385     int i = 0;
386     int err;
387
388     err = zip64local_getByte(pzlib_filefunc_def,filestream,&i);
389     x = (uLong)i;
390
391     if (err==ZIP_OK)
392         err = zip64local_getByte(pzlib_filefunc_def,filestream,&i);
393     x += ((uLong)i)<<8;
394
395     if (err==ZIP_OK)
396         *pX = x;
397     else
398         *pX = 0;
399     return err;
400 }
```

Here is the call graph for this function:

Here is the caller graph for this function:



13.109.4.23 static int zip64local_putValue (const zlib_filefunc64_32_def * *pzlib_filefunc_def,* voidpf *filestream,* ZPOS64_T *x,* int *nbByte*) [static]

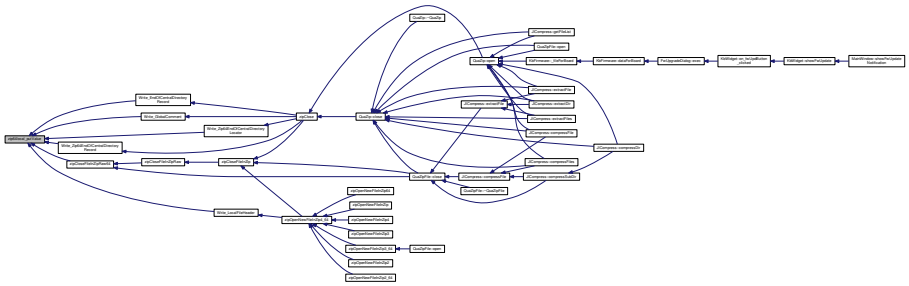
Definition at line 296 of file zip.c.

References ZIP_ERRNO, ZIP_OK, and ZWRITE64.

Referenced by Write_EndOfCentralDirectoryRecord(), Write_GlobalComment(), Write_LocalFileHeader(), Write_-Zip64EndOfCentralDirectoryLocator(), Write_Zip64EndOfCentralDirectoryRecord(), and zipCloseFileInZipRaw64().

```
297 {
298     unsigned char buf[8];
299     int n;
300     for (n = 0; n < nbByte; n++)
301     {
302         buf[n] = (unsigned char)(x & 0xff);
303         x >>= 8;
304     }
305     if (x != 0)
306       {     /* data overflow - hack for ZIP64 (X Roche) */
307       for (n = 0; n < nbByte; n++)
308         {
309           buf[n] = 0xff;
310         }
311       }
312
313     if (ZWRITE64(*pzlib_filefunc_def,filestream,buf,nbByte)!=(uLong)nbByte)
314         return ZIP_ERRNO;
315     else
316         return ZIP_OK;
317 }
```

Here is the caller graph for this function:



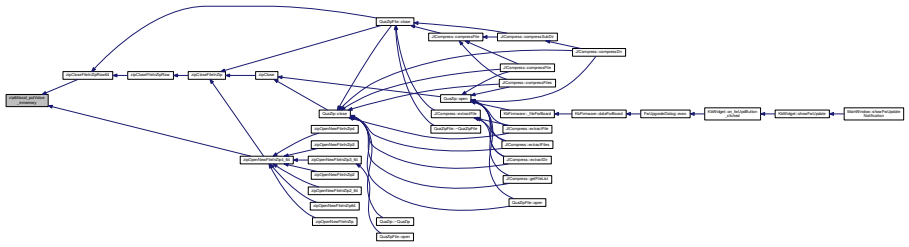
13.109.4.24 static void zip64local_putValue_inmemory (void * *dest*, ZPOS64_T *x*, int *nbByte*) [static]

Definition at line 320 of file zip.c.

Referenced by zipCloseFileInZipRaw64(), and zipOpenNewFileInZip4_64().

```
321 {
322     unsigned char* buf=(unsigned char*)dest;
323     int n;
324     for (n = 0; n < nbByte; n++) {
325         buf[n] = (unsigned char)(x & 0xff);
326         x >>= 8;
327     }
328
329     if (x != 0)
330     {     /* data overflow - hack for ZIP64 */
331        for (n = 0; n < nbByte; n++)
332        {
333           buf[n] = 0xff;
334        }
335     }
336 }
```

Here is the caller graph for this function:



13.109.4.25 static ZPOS64_T zip64local_SearchCentralDir (const zlib_filefunc64_32_def * *pzlib_filefunc_def*, voidpf *filestream*) [static]

Definition at line 489 of file zip.c.

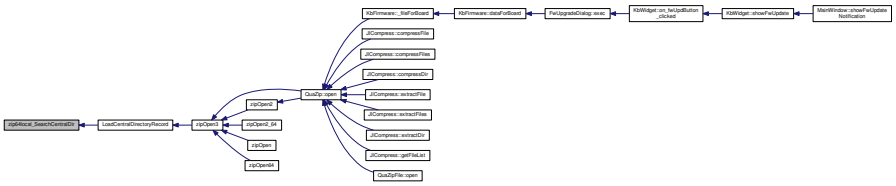
References ALLOC, BUFREADCOMMENT, TRYFREE, ZLIB_FILEFUNC_SEEK_END, ZLIB_FILEFUNC_SEEK_-SET, ZREAD64, ZSEEK64, and ZTELL64.

Referenced by LoadCentralDirectoryRecord().

```
490 {
491   unsigned char* buf;
492   ZPOS64_T uSizeFile;
```

```
493   ZPOS64_T uBackRead;
494   ZPOS64_T uMaxBack=0xffff; /* maximum size of global comment */
495   ZPOS64_T uPosFound=0;
496
497   if (ZSEEK64(*pzlib_filefunc_def,filestream,0,ZLIB_FILEFUNC_SEEK_END) != 0)
498     return 0;
499
500
501   uSizeFile = ZTELL64(*pzlib_filefunc_def,filestream);
502
503   if (uMaxBack>uSizeFile)
504     uMaxBack = uSizeFile;
505
506   buf = (unsigned char*)ALLOC(BUFREADCOMMENT+4);
507   if (buf==NULL)
508     return 0;
509
510   uBackRead = 4;
511   while (uBackRead<uMaxBack)
512   {
513     uLong uReadSize;
514     ZPOS64_T uReadPos ;
515     int i;
516     if (uBackRead+BUFREADCOMMENT>uMaxBack)
517       uBackRead = uMaxBack;
518     else
519       uBackRead+=BUFREADCOMMENT;
520     uReadPos = uSizeFile-uBackRead ;
521
522     uReadSize = ((BUFREADCOMMENT+4) < (uSizeFile-uReadPos)) ?
523       (BUFREADCOMMENT+4) : (uLong)(uSizeFile-uReadPos);
524     if (ZSEEK64(*pzlib_filefunc_def,filestream,uReadPos,
    ZLIB_FILEFUNC_SEEK_SET)!=0)
525       break;
526
527     if (ZREAD64(*pzlib_filefunc_def,filestream,buf,uReadSize)!=uReadSize)
528       break;
529
530     for (i=(int)uReadSize-3; (i--)>0;)
531       if (((*(buf+i))==0x50) && ((*(buf+i+1))==0x4b) &&
532         ((*(buf+i+2))==0x05) && ((*(buf+i+3))==0x06))
533       {
534         uPosFound = uReadPos+i;
535         break;
536       }
537
538     if (uPosFound!=0)
539       break;
540   }
541   TRYFREE(buf);
542   return uPosFound;
543 }
```

Here is the caller graph for this function:



13.109.4.26 static ZPOS64_T zip64local_SearchCentralDir64 (const zlib_filefunc64_32_def * *pzlib_filefunc_def,* voidpf *filestream*) [static]

Definition at line 551 of file zip.c.

References ALLOC, BUFREADCOMMENT, TRYFREE, zip64local_getLong(), zip64local_getLong64(), ZIP_OK, ZLIB_FILEFUNC_SEEK_END, ZLIB_FILEFUNC_SEEK_SET, ZREAD64, ZSEEK64, and ZTELL64.

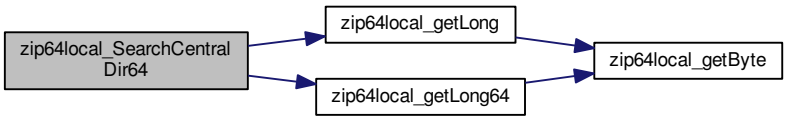
Referenced by LoadCentralDirectoryRecord().

```
552 {
```

```
553   unsigned char* buf;
554   ZPOS64_T uSizeFile;
555   ZPOS64_T uBackRead;
556   ZPOS64_T uMaxBack=0xffff; /* maximum size of global comment */
557   ZPOS64_T uPosFound=0;
558   uLong uL;
559   ZPOS64_T relativeOffset;
560
561   if (ZSEEK64(*pzlib_filefunc_def,filestream,0,ZLIB_FILEFUNC_SEEK_END) != 0)
562     return 0;
563
564   uSizeFile = ZTELL64(*pzlib_filefunc_def,filestream);
565
566   if (uMaxBack>uSizeFile)
567     uMaxBack = uSizeFile;
568
569   buf = (unsigned char*)ALLOC(BUFREADCOMMENT+4);
570   if (buf==NULL)
571     return 0;
572
573   uBackRead = 4;
574   while (uBackRead<uMaxBack)
575   {
576     uLong uReadSize;
577     ZPOS64_T uReadPos;
578     int i;
579     if (uBackRead+BUFREADCOMMENT>uMaxBack)
580       uBackRead = uMaxBack;
581     else
582       uBackRead+=BUFREADCOMMENT;
583     uReadPos = uSizeFile-uBackRead ;
584
585     uReadSize = ((BUFREADCOMMENT+4) < (uSizeFile-uReadPos)) ?
586       (BUFREADCOMMENT+4) : (uLong)(uSizeFile-uReadPos);
587     if (ZSEEK64(*pzlib_filefunc_def,filestream,uReadPos,
    ZLIB_FILEFUNC_SEEK_SET)!=0)
588       break;
589
590     if (ZREAD64(*pzlib_filefunc_def,filestream,buf,uReadSize)!=uReadSize)
591       break;
592
593     for (i=(int)uReadSize-3; (i--)>0;)
594     {
595       /* Signature "0x07064b50" Zip64 end of central directory locater */
596       if (((*(buf+i))==0x50) && ((*(buf+i+1))==0x4b) && ((*(buf+i+2))==0x06) && ((*(buf+i+3))==0x07))
597       {
598         uPosFound = uReadPos+i;
599         break;
600       }
601     }
602
603       if (uPosFound!=0)
604         break;
605   }
606
607   TRYFREE(buf);
608   if (uPosFound == 0)
609     return 0;
610
611   /* Zip64 end of central directory locator */
612   if (ZSEEK64(*pzlib_filefunc_def,filestream, uPosFound,
    ZLIB_FILEFUNC_SEEK_SET)!=0)
613     return 0;
614
615   /* the signature, already checked */
616   if (zip64local_getLong(pzlib_filefunc_def,filestream,&uL)!=
    ZIP_OK)
617     return 0;
618
619   /* number of the disk with the start of the zip64 end of  central directory */
620   if (zip64local_getLong(pzlib_filefunc_def,filestream,&uL)!=
    ZIP_OK)
621     return 0;
622   if (uL != 0)
623     return 0;
624
625   /* relative offset of the zip64 end of central directory record */
626   if (zip64local_getLong64(pzlib_filefunc_def,filestream,&relativeOffset)!=
    ZIP_OK)
627     return 0;
628
629   /* total number of disks */
630   if (zip64local_getLong(pzlib_filefunc_def,filestream,&uL)!=
    ZIP_OK)
631     return 0;
632   if (uL != 1)
633     return 0;
```

```
634
635   /* Goto Zip64 end of central directory record */
636   if (ZSEEK64(*pzlib_filefunc_def,filestream, relativeOffset,
      ZLIB_FILEFUNC_SEEK_SET)!=0)
637     return 0;
638
639   /* the signature */
640   if (zip64local_getLong(pzlib_filefunc_def,filestream,&uL)!=
      ZIP_OK)
641     return 0;
642
643   if (uL != 0x06064b50) /* signature of 'Zip64 end of central directory' */
644     return 0;
645
646   return relativeOffset;
647 }
```

Here is the call graph for this function:



Here is the caller graph for this function:

13.109.4.27 static uLong zip64local_TmzDateToDosDate (const tm_zip * *ptm*) [static]

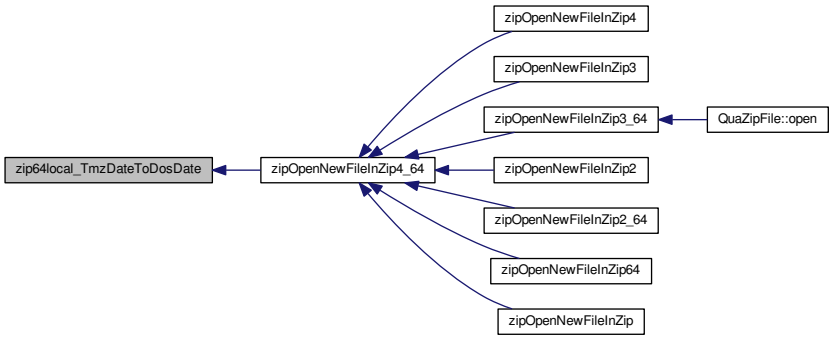
Definition at line 341 of file zip.c.

References tm_zip_s::tm_hour, tm_zip_s::tm_mday, tm_zip_s::tm_min, tm_zip_s::tm_mon, tm_zip_s::tm_sec, and tm_zip_s::tm_year.

Referenced by zipOpenNewFileInZip4_64().

```
342 {
343     uLong year = (uLong)ptm->tm_year;
344     if (year>=1980)
345         year-=1980;
346     else if (year>=80)
347         year-=80;
348     return
349       (uLong) (((ptm->tm_mday) + (32 * (ptm->tm_mon+1)) + (512 * year)) << 16) |
350         ((ptm->tm_sec/2) + (32* ptm->tm_min) + (2048 * (uLong)ptm->
      tm_hour));
351 }
```

Here is the caller graph for this function:



13.109.4.28 int ZEXPORT zipClearFlags (zipFile *file*, unsigned *flags*)

Definition at line 2091 of file zip.c.

References zip64_internal::flags, ZIP_OK, ZIP_PARAMERROR, ZIP_SEQUENTIAL, and ZIP_WRITE_DATA_DESCRIPTOR.

Referenced by QuaZipFile::open().

```
2092 {
2093     zip64_internal* zi;
2094     if (file == NULL)
2095         return ZIP_PARAMERROR;
2096     zi = (zip64_internal*)file;
2097     zi->flags &= ~flags;
2098     // If the data descriptor is not written, we can't use a non-seekable output.
2099     if ((zi->flags & ZIP_WRITE_DATA_DESCRIPTOR) == 0) {
2100         zi->flags &= ~ZIP_SEQUENTIAL;
2101     }
2102     return ZIP_OK;
2103 }
```

Here is the caller graph for this function:

13.109.4.29 int ZEXPORT zipClose (zipFile *file*, const char * *global_comment*)

Definition at line 1942 of file zip.c.

References zip64_internal::add_position_when_writting_offset, zip64_internal::central_dir, linkedlist_datablock_internal_s::data, zip64_internal::filestream, linkedlist_datablock_internal_s::filled_in_this_block, linkedlist_data_s::first_block, zip64_internal::flags, free_linkedlist(), zip64_internal::globalcomment, zip64_internal::in_opened_file_inzip, linkedlist_datablock_internal_s::next_datablock, zip64_internal::number_entry, TRYFREE, Write_EndOf-

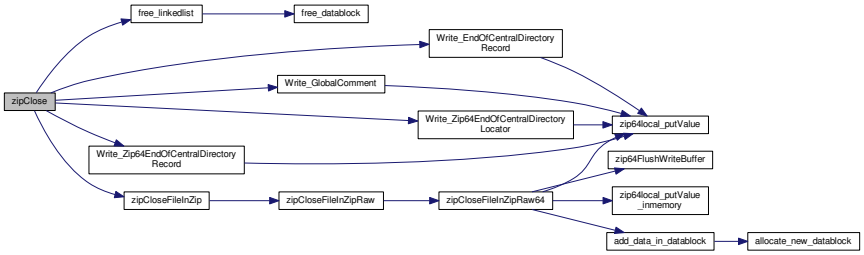
CentralDirectoryRecord(), Write_GlobalComment(), Write_Zip64EndOfCentralDirectoryLocator(), Write_Zip64EndOfCentralDirectoryRecord(), zip64_internal::z_filefunc, ZCLOSE64, ZFAKECLOSE64, ZIP_AUTO_CLOSE, ZIP_ERRNO, ZIP_OK, ZIP_PARAMERROR, zipCloseFileInZip(), ZTELL64, and ZWRITE64.

Referenced by QuaZip::close(), and QuaZip::open().

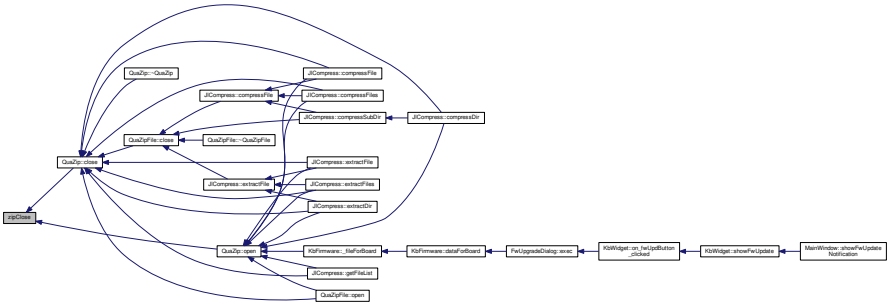
```
1943 {
1944     zip64_internal* zi;
1945     int err = 0;
1946     uLong size_centraldir = 0;
1947     ZPOS64_T centraldir_pos_inzip;
1948     ZPOS64_T pos;
1949 
1950     if (file == NULL)
1951         return ZIP_PARAMERROR;
1952 
1953     zi = (zip64_internal*)file;
1954 
1955     if (zi->in_opened_file_inzip == 1)
1956     {
1957         err = zipCloseFileInZip (file);
1958     }
1959 
1960 #ifndef NO_ADDFILEINEXISTINGZIP
1961     if (global_comment==NULL)
1962         global_comment = zi->globalcomment;
1963 #endif
1964 
1965     centraldir_pos_inzip = ZTELL64(zi->z_filefunc,zi->
     filestream);
1966 
1967     if (err==ZIP_OK)
1968     {
1969         linkedlist_datablock_internal* ldi = zi->
     central_dir.first_block;
1970         while (ldi!=NULL)
1971         {
1972             if ((err==ZIP_OK) && (ldi->filled_in_this_block>0))
1973             {
1974                 if (ZWRITE64(zi->z_filefunc,zi->filestream, ldi->
     data, ldi->filled_in_this_block) != ldi->
     filled_in_this_block)
1975                     err = ZIP_ERRNO;
1976             }
1977 
1978             size_centraldir += ldi->filled_in_this_block;
1979             ldi = ldi->next_datablock;
1980         }
1981     }
1982     free_linkedlist(&(zi->central_dir));
1983 
1984     pos = centraldir_pos_inzip - zi->add_position_when_writting_offset;
1985     if(pos >= 0xffffffff || zi->number_entry > 0xFFFF)
1986     {
1987       ZPOS64_T Zip64EOCDpos = ZTELL64(zi->z_filefunc,zi->
     filestream);
1988       Write_Zip64EndOfCentralDirectoryRecord(zi, size_centraldir,
     centraldir_pos_inzip);
1989 
1990       Write_Zip64EndOfCentralDirectoryLocator(zi, Zip64EOCDpos);
1991     }
1992 
1993     if (err==ZIP_OK)
1994       err = Write_EndOfCentralDirectoryRecord(zi, size_centraldir,
     centraldir_pos_inzip);
1995 
1996     if(err == ZIP_OK)
1997       err = Write_GlobalComment(zi, global_comment);
1998 
1999     if ((zi->flags & ZIP_AUTO_CLOSE) != 0) {
2000         if (ZCLOSE64(zi->z_filefunc,zi->filestream) != 0) {
2001             if (err == ZIP_OK)
2002                 err = ZIP_ERRNO;
2003         }
2004     } else {
2005         if (ZFAKECLOSE64(zi->z_filefunc,zi->filestream) != 0) {
2006             if (err == ZIP_OK)
2007                 err = ZIP_ERRNO;
2008         }
2009     }
2010 
2011 #ifndef NO_ADDFILEINEXISTINGZIP
2012     TRYFREE(zi->globalcomment);
2013 #endif
2014     TRYFREE(zi);
2015 
```

```
2016     return err;
2017 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



13.109.4.30 int ZEXPORT zipCloseFileInZip (zipFile *file*)

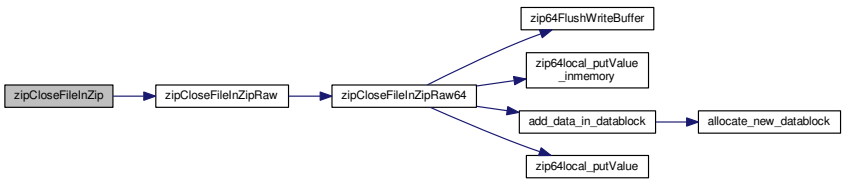
Definition at line 1810 of file zip.c.

References zipCloseFileInZipRaw().

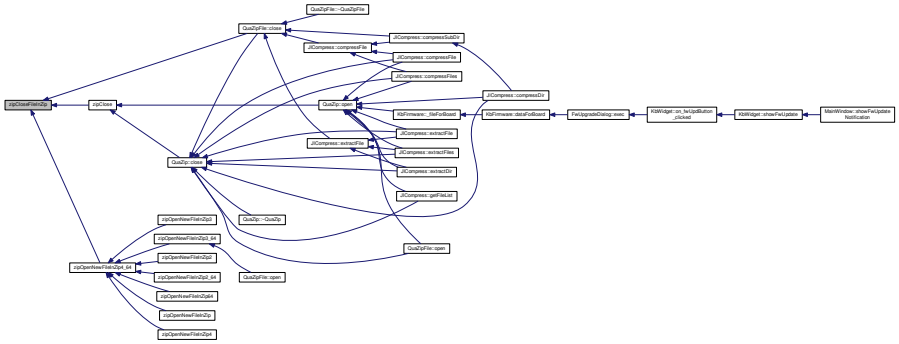
Referenced by QuaZipFile::close(), zipClose(), and zipOpenNewFileInZip4_64().

```
1811 {
1812     return zipCloseFileInZipRaw (file,0,0);
1813 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



13.109.4.31 int ZEXPORT zipCloseFileInZipRaw (zipFile *file*, uLong *uncompressed_size*, uLong *crc32*)

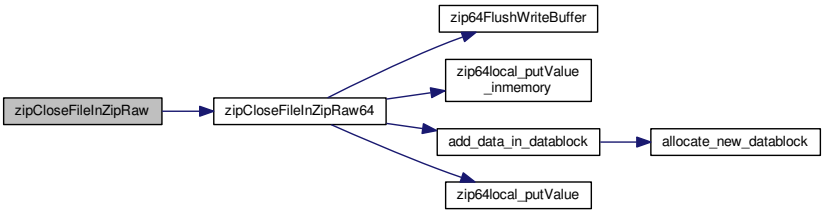
Definition at line 1549 of file zip.c.

References zipCloseFileInZipRaw64().

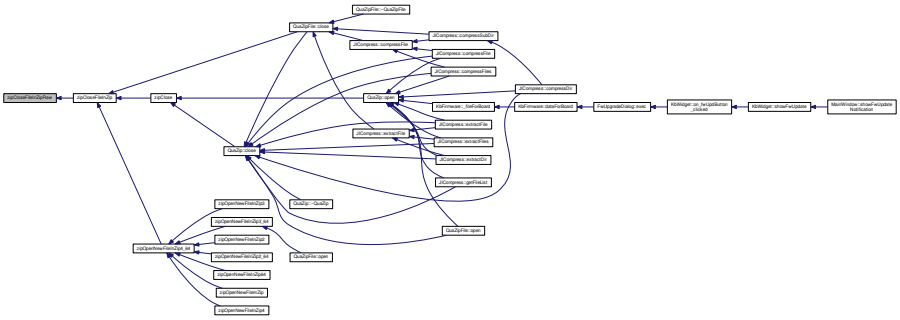
Referenced by zipCloseFileInZip().

```
1550 {
1551     return zipCloseFileInZipRaw64 (file, uncompressed_size, crc32);
1552 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



13.109.4.32 int ZEXPORT zipCloseFileInZipRaw64 (zipFile *file*, ZPOS64_T *uncompressed_size*, uLong *crc32*)

Definition at line 1554 of file zip.c.

References add_data_in_datablock(), curfile64_info::buffered_data, zip64_internal::central_dir, curfile64_info::central_header, zip64_internal::ci, curfile64_info::crc32, curfile64_info::crypt_header_size, DESCRIPTORHEADERMAGIC, zip64_internal::filestream, curfile64_info::flag, zip64_internal::flags, zip64_internal::in_opened_file_inzip, curfile64_info::method, zip64_internal::number_entry, curfile64_info::pos_in_buffered_data, curfile64_info::pos_local_header, curfile64_info::pos_zip64extrainfo, curfile64_info::raw, curfile64_info::size_centralExtra, curfile64_info::size_centralExtraFree, curfile64_info::size_centralheader, curfile64_info::stream, curfile64_info::stream_initialised, curfile64_info::totalCompressedData, curfile64_info::totalUncompressedData, Z_BUFSIZE, Z_BZIP2ED, zip64_internal::z_filefunc, curfile64_info::zip64, zip64FlushWriteBuffer(), zip64local_putValue(), zip64local_putValue_inmemory(), ZIP_BADZIPFILE, ZIP_ERRNO, ZIP_OK, ZIP_PARAMERROR, ZIP_SEQUENTIAL, ZLIB_FILEFUNC_SEEK_SET, ZSEEK64, and ZTELL64.

Referenced by QuaZipFile::close(), and zipCloseFileInZipRaw().

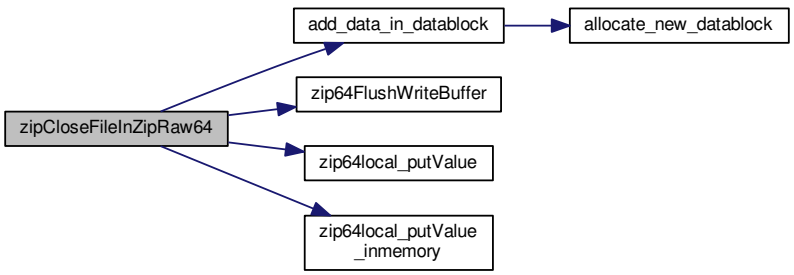
```
1555 {
1556     zip64_internal* zi;
1557     ZPOS64_T compressed_size;
1558     uLong invalidValue = 0xffffffff;
1559     short datasize = 0;
1560     int err=ZIP_OK;
1561
1562     if (file == NULL)
1563         return ZIP_PARAMERROR;
1564     zi = (zip64_internal*)file;
1565
1566     if (zi->in_opened_file_inzip == 0)
1567         return ZIP_PARAMERROR;
1568     zi->ci.stream.avail_in = 0;
1569
1570     if ((zi->ci.method == Z_DEFLATED) && (!zi->ci.raw))
1571                 {
1572                         while (err==ZIP_OK)
1573                         {
1574                                 uLong uTotalOutBefore;
1575                                 if (zi->ci.stream.avail_out == 0)
1576                                 {
1577                                         if (zip64FlushWriteBuffer(zi) ==
     ZIP_ERRNO)
1578                                                 err = ZIP_ERRNO;
1579                                         zi->ci.stream.avail_out = (uInt)
     Z_BUFSIZE;
1580                                         zi->ci.stream.next_out = zi->ci.
     buffered_data;
1581                                 }
1582                                 uTotalOutBefore = zi->ci.stream.total_out;
1583                                 err=deflate(&zi->ci.stream,  Z_FINISH);
1584                                 zi->ci.pos_in_buffered_data += (uInt)(zi->
     ci.stream.total_out - uTotalOutBefore) ;
1585                         }
1586                 }
1587     else if ((zi->ci.method == Z_BZIP2ED) && (!zi->ci.raw))
1588     {
1589 #ifdef HAVE_BZIP2
1590       err = BZ_FINISH_OK;
1591       while (err==BZ_FINISH_OK)
1592       {
1593         uLong uTotalOutBefore;
1594         if (zi->ci.bstream.avail_out == 0)
1595         {
1596           if (zip64FlushWriteBuffer(zi) == ZIP_ERRNO)
1597             err = ZIP_ERRNO;
1598           zi->ci.bstream.avail_out = (uInt)Z_BUFSIZE;
1599           zi->ci.bstream.next_out = (char*)zi->ci.buffered_data;
1600         }
1601         uTotalOutBefore = zi->ci.bstream.total_out_lo32;
1602         err=BZ2_bzCompress(&zi->ci.bstream,  BZ_FINISH);
1603         if(err == BZ_STREAM_END)
1604           err = Z_STREAM_END;
1605
1606         zi->ci.pos_in_buffered_data += (uInt)(zi->ci.bstream.total_out_lo32 -
     uTotalOutBefore);
1607       }
1608
1609       if(err == BZ_FINISH_OK)
1610         err = ZIP_OK;
1611 #endif
1612     }
1613
```

```
1614     if (err==Z_STREAM_END)
1615         err=ZIP_OK; /* this is normal */
1616
1617     if ((zi->ci.pos_in_buffered_data>0) && (err==ZIP_OK))
1618                 {
1619         if (zip64FlushWriteBuffer(zi)==ZIP_ERRNO)
1620             err = ZIP_ERRNO;
1621                 }
1622
1623     if ((zi->ci.method == Z_DEFLATED) && (!zi->ci.raw))
1624     {
1625         int tmp_err = deflateEnd(&zi->ci.stream);
1626         if (err == ZIP_OK)
1627             err = tmp_err;
1628         zi->ci.stream_initialised = 0;
1629     }
1630 #ifdef HAVE_BZIP2
1631     else if((zi->ci.method == Z_BZIP2ED) && (!zi->ci.raw))
1632     {
1633       int tmperr = BZ2_bzCompressEnd(&zi->ci.bstream);
1634                         if (err==ZIP_OK)
1635                                 err = tmperr;
1636                         zi->ci.stream_initialised = 0;
1637     }
1638 #endif
1639
1640     if (!zi->ci.raw)
1641     {
1642         crc32 = (uLong)zi->ci.crc32;
1643         uncompressed_size = zi->ci.totalUncompressedData;
1644     }
1645     compressed_size = zi->ci.totalCompressedData;
1646
1647 #    ifndef NOCRYPT
1648     compressed_size += zi->ci.crypt_header_size;
1649 #    endif
1650
1651     /* update Current Item crc and sizes, */
1652     if(compressed_size >= 0xffffffff || uncompressed_size >= 0xffffffff || zi->
      ci.pos_local_header >= 0xffffffff)
1653     {
1654       /*version Made by*/
1655       zip64local_putValue_inmemory(zi->ci.
      central_header+4,(uLong)45,2);
1656       /*version needed*/
1657       zip64local_putValue_inmemory(zi->ci.
      central_header+6,(uLong)45,2);
1658
1659     }
1660
1661     zip64local_putValue_inmemory(zi->ci.
      central_header+16,crc32,4); /*crc*/
1662
1663
1664     if(compressed_size >= 0xffffffff)
1665       zip64local_putValue_inmemory(zi->ci.
      central_header+20, invalidValue,4); /*compr size*/
1666     else
1667       zip64local_putValue_inmemory(zi->ci.
      central_header+20, compressed_size,4); /*compr size*/
1668
1669     /* set internal file attributes field */
1670     if (zi->ci.stream.data_type == Z_ASCII)
1671         zip64local_putValue_inmemory(zi->ci.
      central_header+36,(uLong)Z_ASCII,2);
1672
1673     if(uncompressed_size >= 0xffffffff)
1674       zip64local_putValue_inmemory(zi->ci.
      central_header+24, invalidValue,4); /*uncompr size*/
1675     else
1676       zip64local_putValue_inmemory(zi->ci.
      central_header+24, uncompressed_size,4); /*uncompr size*/
1677
1678     /* Add ZIP64 extra info field for uncompressed size */
1679     if(uncompressed_size >= 0xffffffff)
1680       datasize += 8;
1681
1682     /* Add ZIP64 extra info field for compressed size */
1683     if(compressed_size >= 0xffffffff)
1684       datasize += 8;
1685
1686     /* Add ZIP64 extra info field for relative offset to local file header of current file */
1687     if(zi->ci.pos_local_header >= 0xffffffff)
1688       datasize += 8;
1689
1690     if(datasize > 0)
1691     {
```

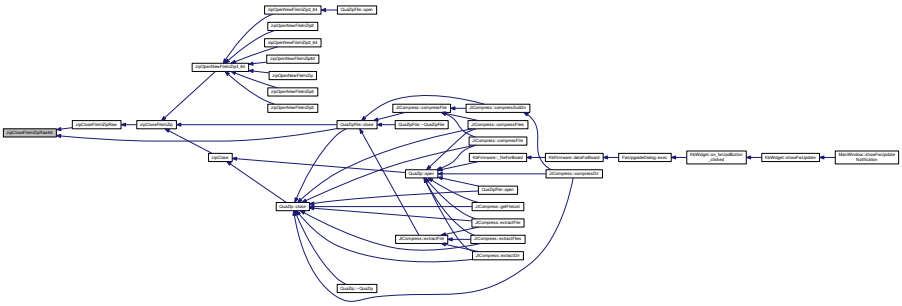
```
1692       char* p = NULL;
1693
1694       if((uLong)(datasize + 4) > zi->ci.size_centralExtraFree)
1695       {
1696         /* we can not write more data to the buffer that we have room for. */
1697         return ZIP_BADZIPFILE;
1698       }
1699
1700       p = zi->ci.central_header + zi->ci.size_centralheader;
1701
1702       /* Add Extra Information Header for 'ZIP64 information' */
1703       zip64local_putValue_inmemory(p, 0x0001, 2); /* HeaderID */
1704       p += 2;
1705       zip64local_putValue_inmemory(p, datasize, 2); /* DataSize */
1706       p += 2;
1707
1708       if(uncompressed_size >= 0xffffffff)
1709       {
1710         zip64local_putValue_inmemory(p, uncompressed_size, 8);
1711         p += 8;
1712       }
1713
1714       if(compressed_size >= 0xffffffff)
1715       {
1716         zip64local_putValue_inmemory(p, compressed_size, 8);
1717         p += 8;
1718       }
1719
1720       if(zi->ci.pos_local_header >= 0xffffffff)
1721       {
1722         zip64local_putValue_inmemory(p, zi->ci.
     pos_local_header, 8);
1723         p += 8;
1724       }
1725
1726       /* Update how much extra free space we got in the memory buffer */
1727       /* and increase the centralheader size so the new ZIP64 fields are included */
1728       /* ( 4 below is the size of HeaderID and DataSize field ) */
1729       zi->ci.size_centralExtraFree -= datasize + 4;
1730       zi->ci.size_centralheader += datasize + 4;
1731
1732       /* Update the extra info size field */
1733       zi->ci.size_centralExtra += datasize + 4;
1734       zip64local_putValue_inmemory(zi->ci.
     central_header+30,(uLong)zi->ci.size_centralExtra,2);
1735     }
1736
1737     if (err==ZIP_OK)
1738         err = add_data_in_datablock(&zi->central_dir, zi->
     ci.central_header, (uLong)zi->ci.size_centralheader);
1739
1740     free(zi->ci.central_header);
1741
1742     if (err==ZIP_OK)
1743     {
1744         if ((zi->flags & ZIP_SEQUENTIAL) == 0) {
1745             /* Update the LocalFileHeader with the new values. */
1746
1747             ZPOS64_T cur_pos_inzip = ZTELL64(zi->z_filefunc,zi->
     filestream);
1748
1749             if (ZSEEK64(zi->z_filefunc,zi->filestream, zi->
     ci.pos_local_header + 14,ZLIB_FILEFUNC_SEEK_SET)!=0)
1750                 err = ZIP_ERRNO;
1751
1752             if (err==ZIP_OK)
1753                 err = zip64local_putValue(&zi->z_filefunc,zi->
     filestream,crc32,4); /* crc 32, unknown */
1754
1755             if(uncompressed_size >= 0xffffffff || compressed_size >= 0xffffffff)
1756             {
1757                 if(zi->ci.pos_zip64extrainfo > 0)
1758                 {
1759                     /* Update the size in the ZIP64 extended field. */
1760                     if (ZSEEK64(zi->z_filefunc,zi->filestream, zi->
     ci.pos_zip64extrainfo + 4,ZLIB_FILEFUNC_SEEK_SET)!=0)
1761                         err = ZIP_ERRNO;
1762
1763                     if (err==ZIP_OK) /* compressed size, unknown */
1764                         err = zip64local_putValue(&zi->
     z_filefunc, zi->filestream, uncompressed_size, 8);
1765
1766                     if (err==ZIP_OK) /* uncompressed size, unknown */
1767                         err = zip64local_putValue(&zi->
     z_filefunc, zi->filestream, compressed_size, 8);
1768                 }
1769             }
```

```
1770            else
1771            {
1772                if (err==ZIP_OK) /* compressed size, unknown */
1773                    err = zip64local_putValue(&zi->z_filefunc,zi->
    filestream,compressed_size,4);
1774
1775                if (err==ZIP_OK) /* uncompressed size, unknown */
1776                    err = zip64local_putValue(&zi->z_filefunc,zi->
    filestream,uncompressed_size,4);
1777            }
1778
1779            if (ZSEEK64(zi->z_filefunc,zi->filestream, cur_pos_inzip,
    ZLIB_FILEFUNC_SEEK_SET)!=0)
1780                err = ZIP_ERRNO;
1781        }
1782
1783        if ((zi->ci.flag & 8) != 0) {
1784            /* Write local Descriptor after file data */
1785            if (err==ZIP_OK)
1786                err = zip64local_putValue(&zi->z_filefunc,zi->
    filestream,(uLong)DESCRIPTORHEADERMAGIC,4);
1787            if (err==ZIP_OK)
1788                err = zip64local_putValue(&zi->z_filefunc,zi->
    filestream,crc32,4); /* crc 32, unknown */
1789            if (zi->ci.zip64) {
1790                if (err==ZIP_OK) /* compressed size, unknown */
1791                    err = zip64local_putValue(&zi->z_filefunc,zi->
    filestream,compressed_size,8);
1792
1793                if (err==ZIP_OK) /* uncompressed size, unknown */
1794                    err = zip64local_putValue(&zi->z_filefunc,zi->
    filestream,uncompressed_size,8);
1795            } else {
1796                if (err==ZIP_OK) /* compressed size, unknown */
1797                    err = zip64local_putValue(&zi->z_filefunc,zi->
    filestream,compressed_size,4);
1798                if (err==ZIP_OK) /* uncompressed size, unknown */
1799                    err = zip64local_putValue(&zi->z_filefunc,zi->
    filestream,uncompressed_size,4);
1800            }
1801        }
1802    }
1803
1804    zi->number_entry ++;
1805    zi->in_opened_file_inzip = 0;
1806
1807    return err;
1808 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



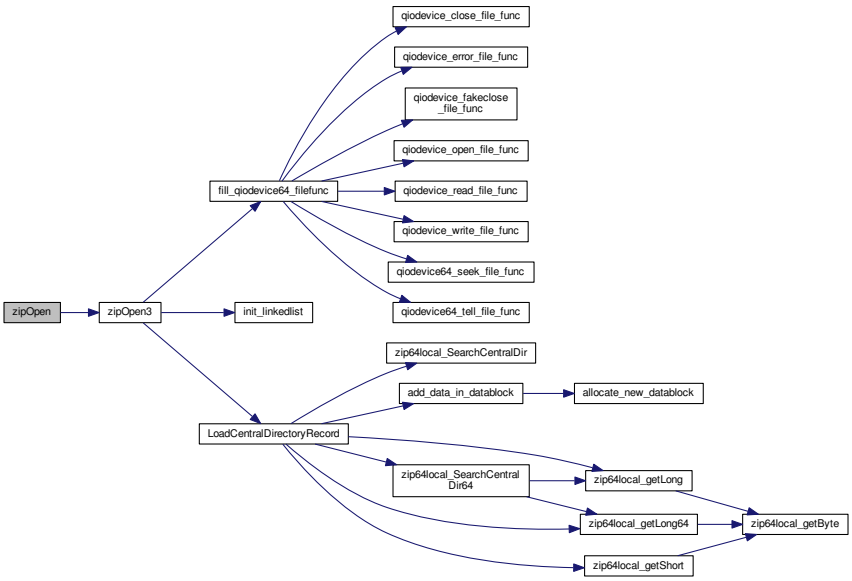
13.109.4.33 zipFile ZEXPORT zipOpen (voidpf *file*, int *append*)

Definition at line 967 of file zip.c.

References ZIP_DEFAULT_FLAGS, and zipOpen3().

```
968 {
969     return zipOpen3(file,append,NULL,NULL, ZIP_DEFAULT_FLAGS);
970 }
```

Here is the call graph for this function:



13.109.4.34 zipFile ZEXPORT zipOpen2 (voidpf *file*, int *append*, zipcharpc * *globalcomment*, zlib_filefunc_def * *pzlib_filefunc32_def*)

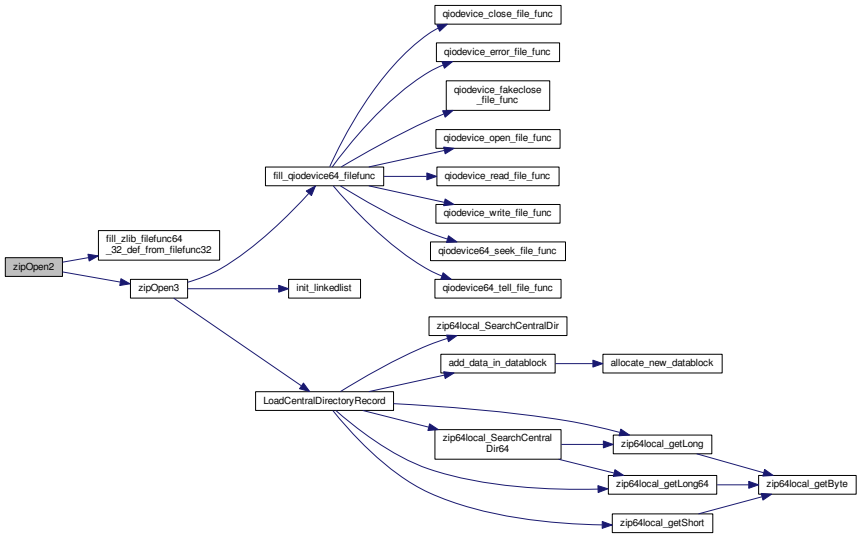
Definition at line 939 of file zip.c.

References fill_zlib_filefunc64_32_def_from_filefunc32(), ZIP_DEFAULT_FLAGS, and zipOpen3().

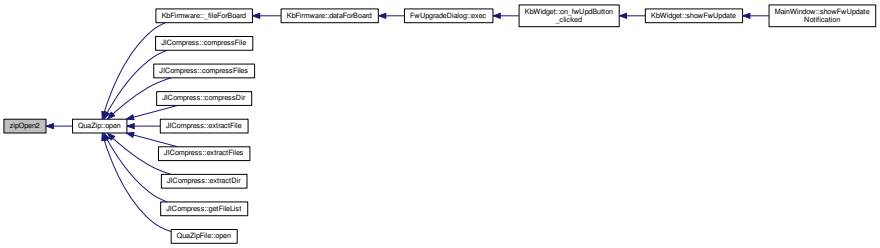
Referenced by QuaZip::open().

```
940 {
941     if (pzlib_filefunc32_def != NULL)
942     {
943         zlib_filefunc64_32_def zlib_filefunc64_32_def_fill;
944         fill_zlib_filefunc64_32_def_from_filefunc32(&
    zlib_filefunc64_32_def_fill,pzlib_filefunc32_def);
945         return zipOpen3(file, append, globalcomment, &zlib_filefunc64_32_def_fill,
    ZIP_DEFAULT_FLAGS);
946     }
947     else
948         return zipOpen3(file, append, globalcomment, NULL,
    ZIP_DEFAULT_FLAGS);
949 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



13.109.4.35 zipFile ZEXPORT zipOpen2_64 (voidpf *file*, int *append*, zipcharpc * *globalcomment*, zlib_filefunc64_def * *pzlib_filefunc_def*)

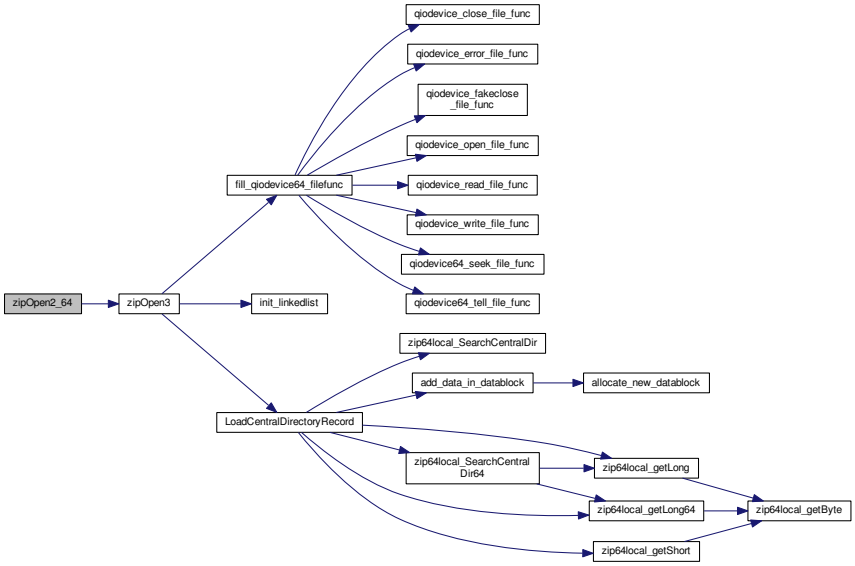
Definition at line 951 of file zip.c.

References zlib_filefunc64_32_def_s::zfile_func64, ZIP_DEFAULT_FLAGS, zipOpen3(), zlib_filefunc64_32_def_s::zseek32_file, and zlib_filefunc64_32_def_s::ztell32_file.

```
952 {
953     if (pzlib_filefunc_def != NULL)
954     {
```

```
955         zlib_filefunc64_32_def zlib_filefunc64_32_def_fill;
956         zlib_filefunc64_32_def_fill.zfile_func64 = *pzlib_filefunc_def;
957         zlib_filefunc64_32_def_fill.ztell32_file = NULL;
958         zlib_filefunc64_32_def_fill.zseek32_file = NULL;
959         return zipOpen3(file, append, globalcomment, &zlib_filefunc64_32_def_fill,
    ZIP_DEFAULT_FLAGS);
960     }
961     else
962         return zipOpen3(file, append, globalcomment, NULL,
    ZIP_DEFAULT_FLAGS);
963 }
```

Here is the call graph for this function:



13.109.4.36 zipFile ZEXPORT zipOpen3 (voidpf *file*, int *append*, zipcharpc * *globalcomment*, zlib_filefunc64_32_def * *pzlib_filefunc64_32_def*, unsigned *flags*)

Definition at line 862 of file zip.c.

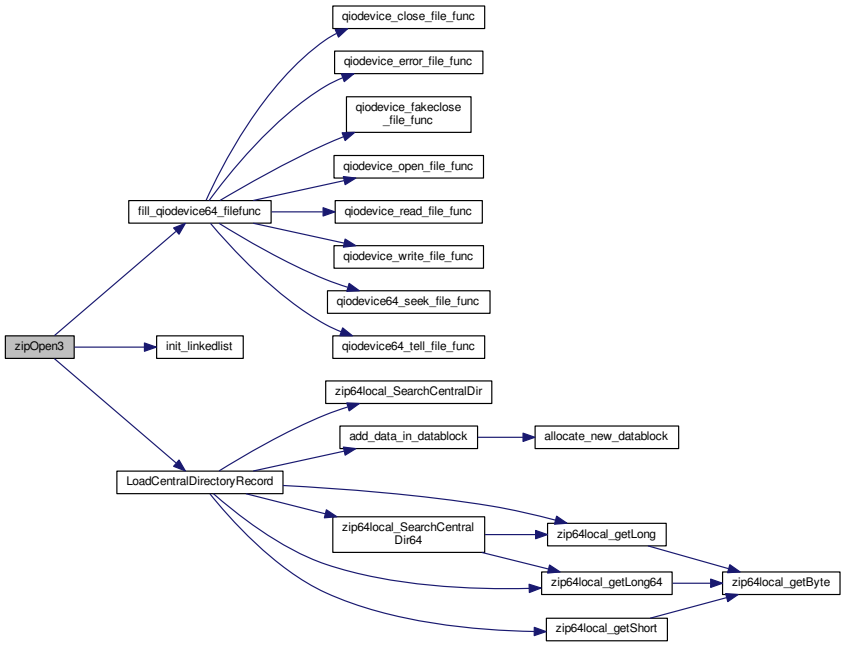
References zip64_internal::add_position_when_writting_offset, ALLOC, APPEND_STATUS_ADDINZIP, APPEND_STATUS_CREATE, APPEND_STATUS_CREATEAFTER, zip64_internal::begin_pos, zip64_internal::central_dir, zip64_internal::ci, zip64_internal::filestream, fill_qiodevice64_filefunc(), zip64_internal::flags, zip64_internal::globalcomment, zip64_internal::in_opened_file_inzip, init_linkedlist(), LoadCentralDirectoryRecord(), zip64_internal::number_entry, SEEK_END, curfile64_info::stream_initialised, TRYFREE, zip64_internal::z_filefunc, ZCLOSE64, ZFAKECLOSE64, zlib_filefunc64_32_def_s::zfile_func64, ZIP_AUTO_CLOSE, ZIP_OK, ZLIB_FILEFUNC_MODE_CREATE, ZLIB_FILEFUNC_MODE_EXISTING, ZLIB_FILEFUNC_MODE_READ, ZLIB_FILEFUNC_MODE_WRITE, ZOPEN64, zlib_filefunc64_32_def_s::zseek32_file, ZSEEK64, zlib_filefunc64_32_def_s::ztell32_file, and ZTELL64.

Referenced by QuaZip::open(), zipOpen(), zipOpen2(), zipOpen2_64(), and zipOpen64().

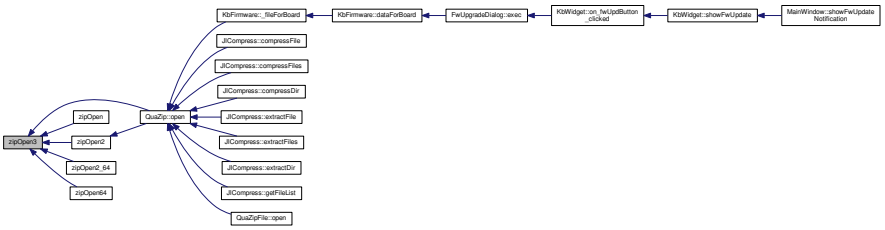
```
864 {
865     zip64_internal ziinit;
866     zip64_internal* zi;
867     int err=ZIP_OK;
868
869     ziinit.flags = flags;
870     ziinit.z_filefunc.zseek32_file = NULL;
871     ziinit.z_filefunc.ztell32_file = NULL;
872     if (pzlib_filefunc64_32_def==NULL)
873         fill_qiodevice64_filefunc(&ziinit.z_filefunc.
    zfile_func64);
```

```
874     else
875         ziinit.z_filefunc = *pzlib_filefunc64_32_def;
876 
877     ziinit.filestream = ZOPEN64(ziinit.z_filefunc,
878                   file,
879                   (append == APPEND_STATUS_CREATE) ?
880                   (ZLIB_FILEFUNC_MODE_READ |
    ZLIB_FILEFUNC_MODE_WRITE | ZLIB_FILEFUNC_MODE_CREATE) :
881                     (ZLIB_FILEFUNC_MODE_READ |
    ZLIB_FILEFUNC_MODE_WRITE | ZLIB_FILEFUNC_MODE_EXISTING))
    ;
882 
883     if (ziinit.filestream == NULL)
884         return NULL;
885 
886     if (append == APPEND_STATUS_CREATEAFTER)
887         ZSEEK64(ziinit.z_filefunc,ziinit.filestream,0,
    SEEK_END);
888 
889     ziinit.begin_pos = ZTELL64(ziinit.z_filefunc,ziinit.
    filestream);
890     ziinit.in_opened_file_inzip = 0;
891     ziinit.ci.stream_initialised = 0;
892     ziinit.number_entry = 0;
893     ziinit.add_position_when_writting_offset = 0;
894     init_linkedlist(&(ziinit.central_dir));
895 
896 
897 
898     zi = (zip64_internal*)ALLOC(sizeof(zip64_internal));
899     if (zi==NULL)
900     {
901         if ((ziinit.flags & ZIP_AUTO_CLOSE) != 0) {
902             ZCLOSE64(ziinit.z_filefunc,ziinit.filestream);
903         } else {
904             ZFAKECLOSE64(ziinit.z_filefunc,ziinit.
    filestream);
905         }
906         return NULL;
907     }
908 
909     /* now we add file in a zipfile */
910 #    ifndef NO_ADDFILEINEXISTINGZIP
911     ziinit.globalcomment = NULL;
912     if (append == APPEND_STATUS_ADDINZIP)
913     {
914       /* Read and Cache Central Directory Records */
915       err = LoadCentralDirectoryRecord(&ziinit);
916     }
917 
918     if (globalcomment)
919     {
920       *globalcomment = ziinit.globalcomment;
921     }
922 #    endif /* !NO_ADDFILEINEXISTINGZIP*/
923 
924     if (err != ZIP_OK)
925     {
926 #    ifndef NO_ADDFILEINEXISTINGZIP
927         TRYFREE(ziinit.globalcomment);
928 #    endif /* !NO_ADDFILEINEXISTINGZIP*/
929         TRYFREE(zi);
930         return NULL;
931     }
932     else
933     {
934         *zi = ziinit;
935         return (zipFile)zi;
936     }
937 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



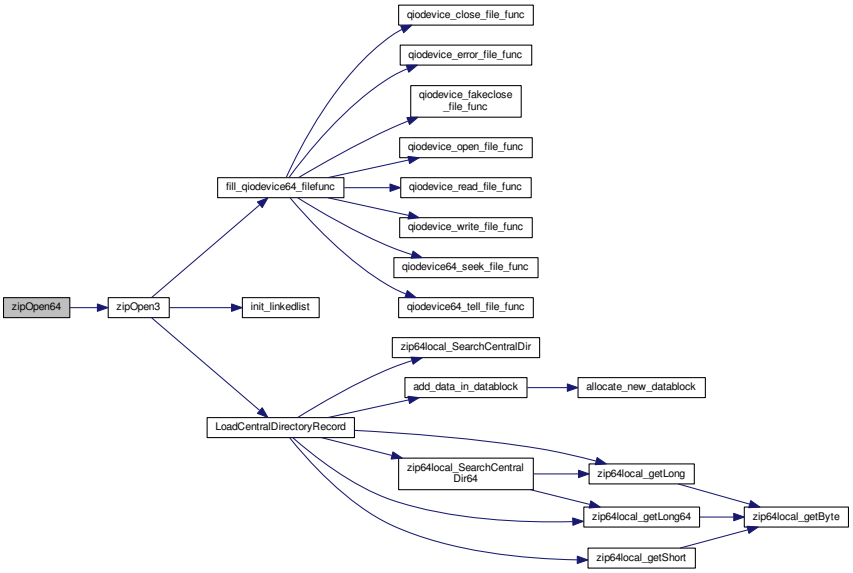
13.109.4.37 zipFile ZEXPORT zipOpen64 (voidpf *file,* int *append*)

Definition at line 972 of file zip.c.

References ZIP_DEFAULT_FLAGS, and zipOpen3().

```
973 {
974     return zipOpen3(file,append,NULL,NULL, ZIP_DEFAULT_FLAGS);
975 }
```

Here is the call graph for this function:



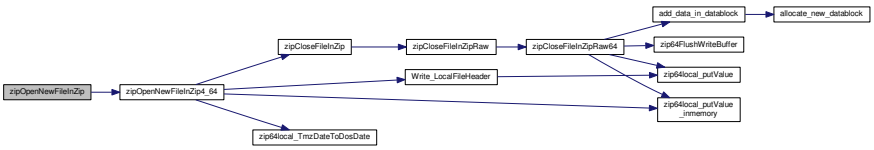
13.109.4.38 int ZEXPORT zipOpenNewFileInZip (zipFile *file*, const char * *filename*, const zip_fileinfo * *zipfi*, const void * *extrafield_local*, uInt *size_extrafield_local*, const void * *extrafield_global*, uInt *size_extrafield_global*, const char * *comment*, int *method*, int *level*)

Definition at line 1390 of file zip.c.

References DEF_MEM_LEVEL, VERSIONMADEBY, and zipOpenNewFileInZip4_64().

```
1394 {
1395     return zipOpenNewFileInZip4_64 (file, filename, zipfi,
1396                                     extrafield_local, size_extrafield_local,
1397                                     extrafield_global, size_extrafield_global,
1398                                     comment, method, level, 0,
1399                                     -MAX_WBITS, DEF_MEM_LEVEL, Z_DEFAULT_STRATEGY,
1400                                     NULL, 0, VERSIONMADEBY, 0, 0);
1401 }
```

Here is the call graph for this function:



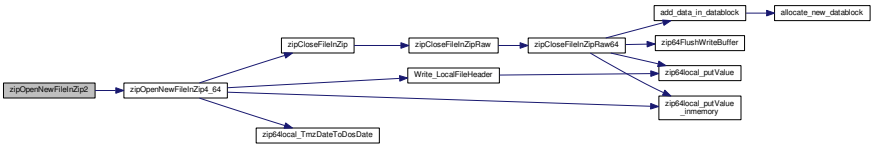
13.109.4.39 int ZEXPORT zipOpenNewFileInZip2 (zipFile *file*, const char * *filename*, const zip_fileinfo * *zipfi*, const void * *extrafield_local*, uInt *size_extrafield_local*, const void * *extrafield_global*, uInt *size_extrafield_global*, const char * *comment*, int *method*, int *level*, int *raw*)

Definition at line 1351 of file zip.c.

References DEF_MEM_LEVEL, VERSIONMADEBY, and zipOpenNewFileInZip4_64().

```
1355 {
1356     return zipOpenNewFileInZip4_64 (file, filename, zipfi,
1357                                     extrafield_local, size_extrafield_local,
1358                                     extrafield_global, size_extrafield_global,
1359                                     comment, method, level, raw,
1360                                     -MAX_WBITS, DEF_MEM_LEVEL, Z_DEFAULT_STRATEGY,
1361                                     NULL, 0, VERSIONMADEBY, 0, 0);
1362 }
```

Here is the call graph for this function:



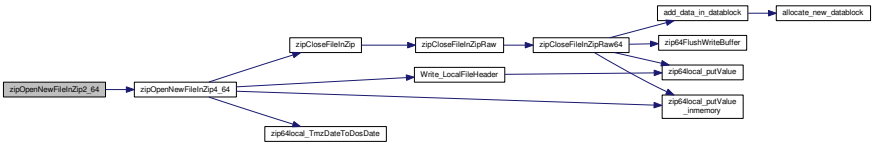
13.109.4.40 int ZEXPORT zipOpenNewFileInZip2_64 (zipFile *file*, const char * *filename*, const zip_fileinfo * *zipfi*, const void * *extrafield_local*, uInt *size_extrafield_local*, const void * *extrafield_global*, uInt *size_extrafield_global*, const char * *comment*, int *method*, int *level*, int *raw*, int *zip64*)

Definition at line 1364 of file zip.c.

References DEF_MEM_LEVEL, VERSIONMADEBY, and zipOpenNewFileInZip4_64().

```
1368 {
1369     return zipOpenNewFileInZip4_64 (file, filename, zipfi,
1370                                     extrafield_local, size_extrafield_local,
1371                                     extrafield_global, size_extrafield_global,
1372                                     comment, method, level, raw,
1373                                     -MAX_WBITS, DEF_MEM_LEVEL, Z_DEFAULT_STRATEGY,
1374                                     NULL, 0, VERSIONMADEBY, 0, zip64);
1375 }
```

Here is the call graph for this function:



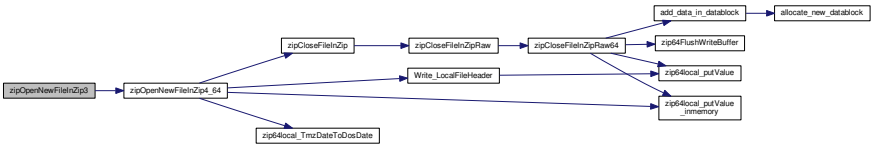
13.109.4.41 int ZEXPORT zipOpenNewFileInZip3 (zipFile *file*, const char * *filename*, const zip_fileinfo * *zipfi*, const void * *extrafield_local*, uInt *size_extrafield_local*, const void * *extrafield_global*, uInt *size_extrafield_global*, const char * *comment*, int *method*, int *level*, int *raw*, int *windowBits*, int *memLevel*, int *strategy*, const char * *password*, uLong *crcForCrypting*)

Definition at line 1321 of file zip.c.

References VERSIONMADEBY, and zipOpenNewFileInZip4_64().

```
1327 {
1328     return zipOpenNewFileInZip4_64 (file, filename, zipfi,
1329                                     extrafield_local, size_extrafield_local,
1330                                     extrafield_global, size_extrafield_global,
1331                                     comment, method, level, raw,
1332                                     windowBits, memLevel, strategy,
1333                                     password, crcForCrypting, VERSIONMADEBY, 0, 0);
1334 }
```

Here is the call graph for this function:



13.109.4.42 int ZEXPORT zipOpenNewFileInZip3_64 (zipFile *file*, const char * *filename*, const zip_fileinfo * *zipfi*, const void * *extrafield_local*, uInt *size_extrafield_local*, const void * *extrafield_global*, uInt *size_extrafield_global*, const char * *comment*, int *method*, int *level*, int *raw*, int *windowBits*, int *memLevel*, int *strategy*, const char * *password*, uLong *crcForCrypting*, int *zip64*)

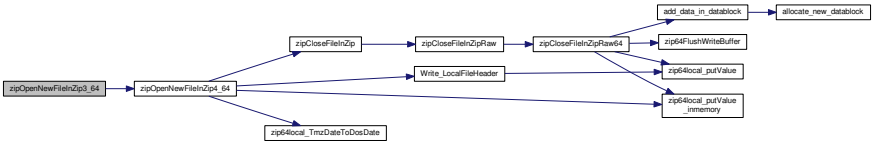
Definition at line 1336 of file zip.c.

References VERSIONMADEBY, and zipOpenNewFileInZip4_64().

Referenced by QuaZipFile::open().

```
1342 {
1343     return zipOpenNewFileInZip4_64 (file, filename, zipfi,
1344                                     extrafield_local, size_extrafield_local,
1345                                     extrafield_global, size_extrafield_global,
1346                                     comment, method, level, raw,
1347                                     windowBits, memLevel, strategy,
1348                                     password, crcForCrypting, VERSIONMADEBY, 0, zip64);
1349 }
```

Here is the call graph for this function:



Here is the caller graph for this function:

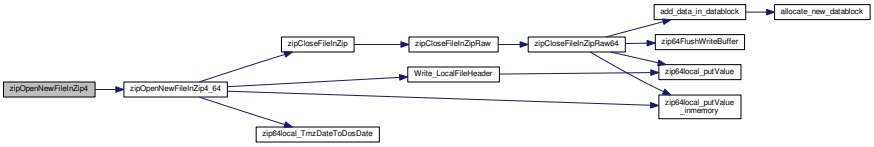
13.109.4.43 int ZEXPORT zipOpenNewFileInZip4 (zipFile *file*, const char * *filename*, const zip_fileinfo * *zipfi*, const void * *extrafield_local*, uInt *size_extrafield_local*, const void * *extrafield_global*, uInt *size_extrafield_global*, const char * *comment*, int *method*, int *level*, int *raw*, int *windowBits*, int *memLevel*, int *strategy*, const char * *password*, uLong *crcForCrypting*, uLong *versionMadeBy*, uLong *flagBase*)

Definition at line 1305 of file zip.c.

References zipOpenNewFileInZip4_64().

```
1312 {
1313     return zipOpenNewFileInZip4_64 (file, filename, zipfi,
1314                                     extrafield_local, size_extrafield_local,
1315                                     extrafield_global, size_extrafield_global,
1316                                     comment, method, level, raw,
1317                                     windowBits, memLevel, strategy,
1318                                     password, crcForCrypting, versionMadeBy, flagBase, 0);
1319 }
```

Here is the call graph for this function:



13.109.4.44 int ZEXPORT zipOpenNewFileInZip4_64 (zipFile *file*, const char * *filename*, const zip_fileinfo * *zipfi*, const void * *extrafield_local*, uInt *size_extrafield_local*, const void * *extrafield_global*, uInt *size_extrafield_global*, const char * *comment*, int *method*, int *level*, int *raw*, int *windowBits*, int *memLevel*, int *strategy*, const char * *password*, uLong *crcForCrypting*, uLong *versionMadeBy*, uLong *flagBase*, int *zip64*)

Definition at line 1077 of file zip.c.

References zip64_internal::add_position_when_writting_offset, ALLOC, curfile64_info::buffered_data, curfile64_info::central_header, CENTRALHEADERMAGIC, zip64_internal::ci, curfile64_info::crc32, curfile64_info::crypt_header_size, zip_fileinfo::dosDate, curfile64_info::dosDate, curfile64_info::encrypt, zip_fileinfo::external_fa, zip64_internal::filestream, curfile64_info::flag, zip64_internal::flags, zip64_internal::in_opened_file_inzip, zip_fileinfo::internal_fa, curfile64_info::keys, curfile64_info::method, curfile64_info::pcrc_32_tab, curfile64_info::pos_in_buffered_data, curfile64_info::pos_local_header, curfile64_info::pos_zip64extrainfo, curfile64_info::raw, curfile64_info::size_centralExtra, curfile64_info::size_centralExtraFree, curfile64_info::size_centralheader, SIZECENTRALHEADER, curfile64_info::stream, curfile64_info::stream_initialised, zip_fileinfo::tmz_date, curfile64_info::totalCompressedData, curfile64_info::totalUncompressedData, Write_LocalFileHeader(), Z_BUFSIZE, Z_BZIP2ED, zip64_internal::z_filefunc, curfile64_info::zip64, zip64local_putValue_inmemory(), zip64local_TmzDateToDosDate(), ZIP_ERRNO, ZIP_INTERNALERROR, ZIP_OK, ZIP_PARAMERROR, ZIP_SEQUENTIAL, ZIP_WRITE_DATA_DESCRIPTOR, zipCloseFileInZip(), ZTELL64, and ZWRITE64.

Referenced by zipOpenNewFileInZip(), zipOpenNewFileInZip2(), zipOpenNewFileInZip2_64(), zipOpenNewFileInZip3(), zipOpenNewFileInZip3_64(), zipOpenNewFileInZip4(), and zipOpenNewFileInZip64().

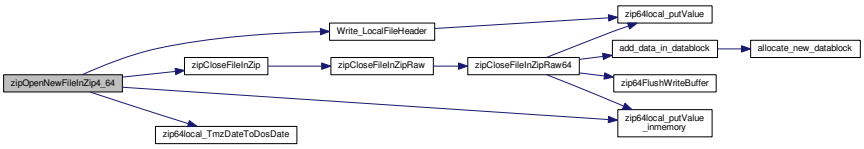
```
1084 {
1085     zip64_internal* zi;
1086     uInt size_filename;
1087     uInt size_comment;
1088     uInt i;
1089     int err = ZIP_OK;
1090     uLong version_to_extract;
1091
1092 #    ifdef NOCRYPT
1093     if (password != NULL)
1094         return ZIP_PARAMERROR;
1095 #    endif
1096
1097     if (file == NULL)
1098         return ZIP_PARAMERROR;
1099
1100 #ifdef HAVE_BZIP2
1101     if ((method!=0) && (method!=Z_DEFLATED) && (method!=Z_BZIP2ED))
1102       return ZIP_PARAMERROR;
1103 #else
1104     if ((method!=0) && (method!=Z_DEFLATED))
1105       return ZIP_PARAMERROR;
1106 #endif
```

```
1107
1108    zi = (zip64_internal*)file;
1109
1110    if (zi->in_opened_file_inzip == 1)
1111    {
1112        err = zipCloseFileInZip (file);
1113        if (err != ZIP_OK)
1114            return err;
1115    }
1116
1117    if (method == 0
1118            && (level == 0 || (zi->flags & ZIP_WRITE_DATA_DESCRIPTOR) == 0)
1119            && (zi->flags & ZIP_SEQUENTIAL) == 0)
1120    {
1121        version_to_extract = 10;
1122    }
1123    else
1124    {
1125        version_to_extract = 20;
1126    }
1127
1128    if (filename==NULL)
1129        filename="-";
1130
1131    if (comment==NULL)
1132        size_comment = 0;
1133    else
1134        size_comment = (uInt)strlen(comment);
1135
1136    size_filename = (uInt)strlen(filename);
1137
1138    if (zipfi == NULL)
1139        zi->ci.dosDate = 0;
1140    else
1141    {
1142        if (zipfi->dosDate != 0)
1143            zi->ci.dosDate = zipfi->dosDate;
1144        else
1145          zi->ci.dosDate = zip64local_TmzDateToDosDate(&zipfi->
    tmz_date);
1146    }
1147
1148    zi->ci.flag = flagBase;
1149    if ((level==8) || (level==9))
1150      zi->ci.flag |= 2;
1151    if (level==2)
1152      zi->ci.flag |= 4;
1153    if (level==1)
1154      zi->ci.flag |= 6;
1155    if (password != NULL)
1156      zi->ci.flag |= 1;
1157    if (version_to_extract >= 20
1158            && ((zi->flags & ZIP_WRITE_DATA_DESCRIPTOR) != 0
1159                || (zi->flags & ZIP_SEQUENTIAL) != 0))
1160      zi->ci.flag |= 8;
1161
1162    zi->ci.crc32 = 0;
1163    zi->ci.method = method;
1164    zi->ci.encrypt = 0;
1165    zi->ci.stream_initialised = 0;
1166    zi->ci.pos_in_buffered_data = 0;
1167    zi->ci.raw = raw;
1168    zi->ci.pos_local_header = ZTELL64(zi->z_filefunc,zi->
    filestream);
1169
1170    zi->ci.size_centralheader = SIZECENTRALHEADER + size_filename +
    size_extrafield_global + size_comment;
1171    zi->ci.size_centralExtraFree = 32; /* Extra space we have reserved in case we
     need to add ZIP64 extra info data */
1172
1173    zi->ci.central_header = (char*)ALLOC((uInt)zi->ci.
    size_centralheader + zi->ci.size_centralExtraFree);
1174
1175    zi->ci.size_centralExtra = size_extrafield_global;
1176    zip64local_putValue_inmemory(zi->ci.
    central_header,(uLong)CENTRALHEADERMAGIC,4);
1177    /* version info */
1178    zip64local_putValue_inmemory(zi->ci.
    central_header+4,(uLong)versionMadeBy,2);
1179    zip64local_putValue_inmemory(zi->ci.
    central_header+6,(uLong)version_to_extract,2);
1180    zip64local_putValue_inmemory(zi->ci.
    central_header+8,(uLong)zi->ci.flag,2);
1181    zip64local_putValue_inmemory(zi->ci.
    central_header+10,(uLong)zi->ci.method,2);
1182    zip64local_putValue_inmemory(zi->ci.
    central_header+12,(uLong)zi->ci.dosDate,4);
```

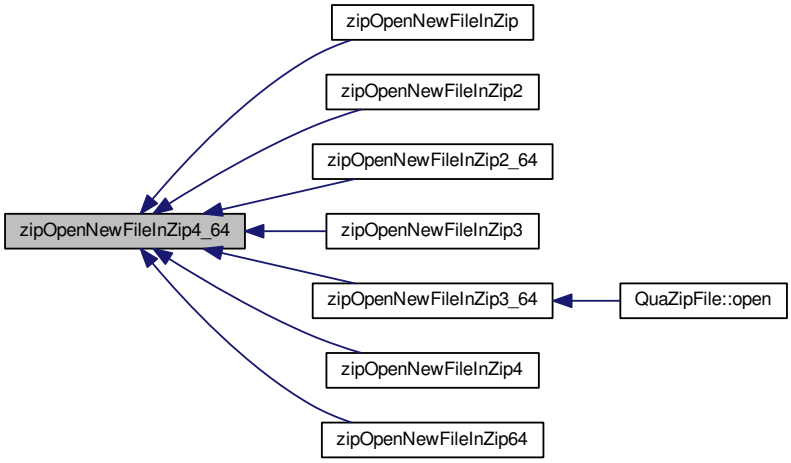
```
1183     zip64local_putValue_inmemory(zi->ci.
     central_header+16,(uLong)0,4); /*crc*/
1184     zip64local_putValue_inmemory(zi->ci.
     central_header+20,(uLong)0,4); /*compr size*/
1185     zip64local_putValue_inmemory(zi->ci.
     central_header+24,(uLong)0,4); /*uncompr size*/
1186     zip64local_putValue_inmemory(zi->ci.
     central_header+28,(uLong)size_filename,2);
1187     zip64local_putValue_inmemory(zi->ci.
     central_header+30,(uLong)size_extrafield_global,2);
1188     zip64local_putValue_inmemory(zi->ci.
     central_header+32,(uLong)size_comment,2);
1189     zip64local_putValue_inmemory(zi->ci.
     central_header+34,(uLong)0,2); /*disk nm start*/
1190
1191     if (zipfi==NULL)
1192         zip64local_putValue_inmemory(zi->ci.
     central_header+36,(uLong)0,2);
1193     else
1194         zip64local_putValue_inmemory(zi->ci.
     central_header+36,(uLong)zipfi->internal_fa,2);
1195
1196     if (zipfi==NULL)
1197         zip64local_putValue_inmemory(zi->ci.
     central_header+38,(uLong)0,4);
1198     else
1199         zip64local_putValue_inmemory(zi->ci.
     central_header+38,(uLong)zipfi->external_fa,4);
1200
1201     if(zi->ci.pos_local_header >= 0xffffffff)
1202       zip64local_putValue_inmemory(zi->ci.
     central_header+42,(uLong)0xffffffff,4);
1203     else
1204       zip64local_putValue_inmemory(zi->ci.
     central_header+42,(uLong)zi->ci.pos_local_header - zi->
     add_position_when_writting_offset,4);
1205
1206     for (i=0;i<size_filename;i++)
1207         *(zi->ci.central_header+SIZECENTRALHEADER+i) = *(filename+i);
1208
1209     for (i=0;i<size_extrafield_global;i++)
1210         *(zi->ci.central_header+SIZECENTRALHEADER+size_filename+i) =
1211               *(((const char*)extrafield_global)+i);
1212
1213     for (i=0;i<size_comment;i++)
1214         *(zi->ci.central_header+SIZECENTRALHEADER+size_filename+
1215               size_extrafield_global+i) = *(comment+i);
1216     if (zi->ci.central_header == NULL)
1217         return ZIP_INTERNALERROR;
1218
1219     zi->ci.zip64 = zip64;
1220     zi->ci.totalCompressedData = 0;
1221     zi->ci.totalUncompressedData = 0;
1222     zi->ci.pos_zip64extrainfo = 0;
1223
1224     err = Write_LocalFileHeader(zi, filename, size_extrafield_local,
1225                                 extrafield_local, version_to_extract);
1226
1227 #ifdef HAVE_BZIP2
1228     zi->ci.bstream.avail_in = (uInt)0;
1229     zi->ci.bstream.avail_out = (uInt)Z_BUFSIZE;
1230     zi->ci.bstream.next_out = (char*)zi->ci.buffered_data;
1231     zi->ci.bstream.total_in_hi32 = 0;
1232     zi->ci.bstream.total_in_lo32 = 0;
1233     zi->ci.bstream.total_out_hi32 = 0;
1234     zi->ci.bstream.total_out_lo32 = 0;
1235 #endif
1236
1237     zi->ci.stream.avail_in = (uInt)0;
1238     zi->ci.stream.avail_out = (uInt)Z_BUFSIZE;
1239     zi->ci.stream.next_out = zi->ci.buffered_data;
1240     zi->ci.stream.total_in = 0;
1241     zi->ci.stream.total_out = 0;
1242     zi->ci.stream.data_type = Z_BINARY;
1243
1244 #ifdef HAVE_BZIP2
1245     if ((err==ZIP_OK) && (zi->ci.method == Z_DEFLATED || zi->ci.
     method == Z_BZIP2ED) && (!zi->ci.raw))
1246 #else
1247     if ((err==ZIP_OK) && (zi->ci.method == Z_DEFLATED) && (!zi->ci.
     raw))
1248 #endif
1249     {
1250         if(zi->ci.method == Z_DEFLATED)
1251         {
1252           zi->ci.stream.zalloc = (alloc_func)0;
1253           zi->ci.stream.zfree = (free_func)0;
```

```
1254            zi->ci.stream.opaque = (voidpf)0;
1255
1256            if (windowBits>0)
1257                windowBits = -windowBits;
1258
1259            err = deflateInit2(&zi->ci.stream, level, Z_DEFLATED, windowBits, memLevel, strategy);
1260
1261            if (err==Z_OK)
1262                zi->ci.stream_initialised = Z_DEFLATED;
1263          }
1264          else if(zi->ci.method == Z_BZIP2ED)
1265          {
1266 #ifdef HAVE_BZIP2
1267              /* Init BZip stuff here */
1268            zi->ci.bstream.bzalloc = 0;
1269            zi->ci.bstream.bzfree = 0;
1270            zi->ci.bstream.opaque = (voidpf)0;
1271
1272            err = BZ2_bzCompressInit(&zi->ci.bstream, level, 0,35);
1273            if(err == BZ_OK)
1274              zi->ci.stream_initialised = Z_BZIP2ED;
1275 #endif
1276          }
1277
1278      }
1279
1280 #    ifndef NOCRYPT
1281      zi->ci.crypt_header_size = 0;
1282      if ((err==Z_OK) && (password != NULL))
1283      {
1284          unsigned char bufHead[RAND_HEAD_LEN];
1285          unsigned int sizeHead;
1286          zi->ci.encrypt = 1;
1287          zi->ci.pcrc_32_tab = get_crc_table();
1288          /*init_keys(password,zi->ci.keys,zi->ci.pcrc_32_tab);*/
1289          if (crcForCrypting == 0) {
1290              crcForCrypting = (uLong)zi->ci.dosDate << 16; /* ATTANTION! Without this row, you
     don't unpack your password protected archive in other app. */
1291          }
1292          sizeHead=crypthead(password,bufHead,RAND_HEAD_LEN,zi->ci.keys,zi->
      ci.pcrc_32_tab,crcForCrypting);
1293          zi->ci.crypt_header_size = sizeHead;
1294
1295          if (ZWRITE64(zi->z_filefunc,zi->filestream,bufHead,sizeHead) !=
     sizeHead)
1296                  err = ZIP_ERRNO;
1297      }
1298 #    endif
1299
1300      if (err==Z_OK)
1301          zi->in_opened_file_inzip = 1;
1302      return err;
1303 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



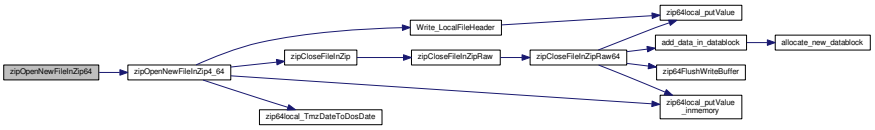
13.109.4.45 int ZEXPORT zipOpenNewFileInZip64 (zipFile *file*, const char * *filename*, const zip_fileinfo * *zipfi*, const void * *extrafield_local*, uInt *size_extrafield_local*, const void * *extrafield_global*, uInt *size_extrafield_global*, const char * *comment*, int *method*, int *level*, int *zip64*)

Definition at line 1377 of file zip.c.

References DEF_MEM_LEVEL, VERSIONMADEBY, and zipOpenNewFileInZip4_64().

```
1381 {
1382     return zipOpenNewFileInZip4_64 (file, filename, zipfi,
1383                                     extrafield_local, size_extrafield_local,
1384                                     extrafield_global, size_extrafield_global,
1385                                     comment, method, level, 0,
1386                                     -MAX_WBITS, DEF_MEM_LEVEL, Z_DEFAULT_STRATEGY,
1387                                     NULL, 0, VERSIONMADEBY, 0, zip64);
1388 }
```

Here is the call graph for this function:



13.109.4.46 int ZEXPORT zipRemoveExtraInfoBlock (char * *pData*, int * *dataLen*, short *sHeader*)

Definition at line 2019 of file zip.c.

References ALLOC, TRYFREE, ZIP_ERRNO, ZIP_OK, and ZIP_PARAMERROR.

```
2020 {
2021   char* p = pData;
2022   int size = 0;
```

```
2023   char* pNewHeader;
2024   char* pTmp;
2025   short header;
2026   short dataSize;
2027
2028   int retVal = ZIP_OK;
2029
2030   if(pData == NULL || *dataLen < 4)
2031     return ZIP_PARAMERROR;
2032
2033   pNewHeader = (char*)ALLOC(*dataLen);
2034   pTmp = pNewHeader;
2035
2036   while(p < (pData + *dataLen))
2037   {
2038     header = *(short*)p;
2039     dataSize = *(((short*)p)+1);
2040
2041     if( header == sHeader ) /* Header found. */
2042     {
2043       p += dataSize + 4; /* skip it. do not copy to temp buffer */
2044     }
2045     else
2046     {
2047       /* Extra Info block should not be removed, So copy it to the temp buffer. */
2048       memcpy(pTmp, p, dataSize + 4);
2049       p += dataSize + 4;
2050       size += dataSize + 4;
2051     }
2052
2053   }
2054
2055   if(size < *dataLen)
2056   {
2057     /* clean old extra info block. */
2058     memset(pData,0, *dataLen);
2059
2060     /* copy the new extra info block over the old */
2061     if(size > 0)
2062       memcpy(pData, pNewHeader, size);
2063
2064     /* set the new extra info size */
2065     *dataLen = size;
2066
2067     retVal = ZIP_OK;
2068   }
2069   else
2070     retVal = ZIP_ERRNO;
2071
2072   TRYFREE(pNewHeader);
2073
2074   return retVal;
2075 }
```

13.109.4.47 int ZEXPORT zipSetFlags (zipFile *file*, unsigned *flags*)

Definition at line 2077 of file zip.c.

References zip64_internal::flags, ZIP_OK, ZIP_PARAMERROR, ZIP_SEQUENTIAL, and ZIP_WRITE_DATA_DESCRIPTOR.

Referenced by QuaZip::open(), and QuaZipFile::open().

```
2078 {
2079     zip64_internal* zi;
2080     if (file == NULL)
2081         return ZIP_PARAMERROR;
2082     zi = (zip64_internal*)file;
2083     zi->flags |= flags;
2084     // If the output is non-seekable, the data descriptor is needed.
2085     if ((zi->flags & ZIP_SEQUENTIAL) != 0) {
2086         zi->flags |= ZIP_WRITE_DATA_DESCRIPTOR;
2087     }
2088     return ZIP_OK;
2089 }
```

Here is the caller graph for this function:

13.109.4.48 int ZEXPORT zipWriteInFileInZip (zipFile *file*, const void * *buf*, unsigned int *len*)

Definition at line 1442 of file zip.c.

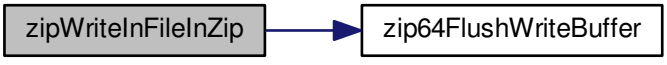
References curfile64_info::buffered_data, zip64_internal::ci, curfile64_info::crc32, zip64_internal::in_opened_file_inzip, curfile64_info::method, curfile64_info::pos_in_buffered_data, curfile64_info::raw, curfile64_info::stream, Z_BUFSIZE, Z_BZIP2ED, zip64FlushWriteBuffer(), ZIP_ERRNO, ZIP_OK, and ZIP_PARAMERROR.

Referenced by QuaZipFile::writeData().

```
1443 {
1444     zip64_internal* zi;
1445     int err=ZIP_OK;
1446
1447     if (file == NULL)
1448         return ZIP_PARAMERROR;
1449     zi = (zip64_internal*)file;
1450
1451     if (zi->in_opened_file_inzip == 0)
1452         return ZIP_PARAMERROR;
1453
1454     zi->ci.crc32 = crc32(zi->ci.crc32,buf,(uInt)len);
1455
1456 #ifdef HAVE_BZIP2
1457     if(zi->ci.method == Z_BZIP2ED && (!zi->ci.raw))
1458     {
1459       zi->ci.bstream.next_in = (void*)buf;
1460       zi->ci.bstream.avail_in = len;
1461       err = BZ_RUN_OK;
1462
1463       while ((err==BZ_RUN_OK) && (zi->ci.bstream.avail_in>0))
1464       {
1465         if (zi->ci.bstream.avail_out == 0)
1466         {
1467           if (zip64FlushWriteBuffer(zi) == ZIP_ERRNO)
1468             err = ZIP_ERRNO;
1469           zi->ci.bstream.avail_out = (uInt)Z_BUFSIZE;
1470           zi->ci.bstream.next_out = (char*)zi->ci.buffered_data;
1471         }
1472
1473
1474         if(err != BZ_RUN_OK)
1475           break;
1476
1477         if ((zi->ci.method == Z_BZIP2ED) && (!zi->ci.raw))
1478         {
1479           uLong uTotalOutBefore_lo = zi->ci.bstream.total_out_lo32;
1480 /*          uLong uTotalOutBefore_hi = zi->ci.bstream.total_out_hi32; */
1481           err=BZ2_bzCompress(&zi->ci.bstream,  BZ_RUN);
1482
1483           zi->ci.pos_in_buffered_data += (uInt)(zi->ci.bstream.total_out_lo32 -
     uTotalOutBefore_lo) ;
1484         }
1485       }
1486
1487       if(err == BZ_RUN_OK)
1488         err = ZIP_OK;
1489     }
1490     else
1491 #endif
1492     {
1493       zi->ci.stream.next_in = (Bytef*)buf;
```

```
1494        zi->ci.stream.avail_in = len;
1495
1496        while ((err==ZIP_OK) && (zi->ci.stream.avail_in>0))
1497        {
1498            if (zi->ci.stream.avail_out == 0)
1499            {
1500                if (zip64FlushWriteBuffer(zi) == ZIP_ERRNO)
1501                    err = ZIP_ERRNO;
1502                zi->ci.stream.avail_out = (uInt)Z_BUFSIZE;
1503                zi->ci.stream.next_out = zi->ci.buffered_data;
1504            }
1505
1506
1507            if(err != ZIP_OK)
1508                break;
1509
1510            if ((zi->ci.method == Z_DEFLATED) && (!zi->ci.raw))
1511            {
1512                uLong uTotalOutBefore = zi->ci.stream.total_out;
1513                err=deflate(&zi->ci.stream,  Z_NO_FLUSH);
1514                if(uTotalOutBefore > zi->ci.stream.total_out)
1515                {
1516                  int bBreak = 0;
1517                  bBreak++;
1518                }
1519
1520                zi->ci.pos_in_buffered_data += (uInt)(zi->ci.
    stream.total_out - uTotalOutBefore) ;
1521            }
1522            else
1523            {
1524                uInt copy_this,i;
1525                if (zi->ci.stream.avail_in < zi->ci.stream.avail_out)
1526                    copy_this = zi->ci.stream.avail_in;
1527                else
1528                    copy_this = zi->ci.stream.avail_out;
1529
1530                for (i = 0; i < copy_this; i++)
1531                    *(((char*)zi->ci.stream.next_out)+i) =
1532                        *(((const char*)zi->ci.stream.next_in)+i);
1533                {
1534                    zi->ci.stream.avail_in -= copy_this;
1535                    zi->ci.stream.avail_out-= copy_this;
1536                    zi->ci.stream.next_in+= copy_this;
1537                    zi->ci.stream.next_out+= copy_this;
1538                    zi->ci.stream.total_in+= copy_this;
1539                    zi->ci.stream.total_out+= copy_this;
1540                    zi->ci.pos_in_buffered_data += copy_this;
1541                }
1542            }
1543        }/* while(...) */
1544    }
1545
1546    return err;
1547 }
```

Here is the call graph for this function:



Here is the caller graph for this function:

13.109.5 Variable Documentation

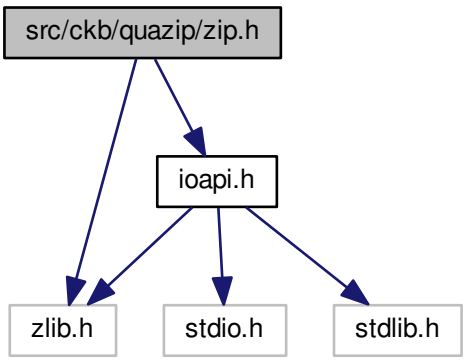
13.109.5.1 const char zip_copyright[ ] =" zip 1.01 Copyright 1998-2004 Gilles Vollant - http://www.winimage.com/zLibDll"

Definition at line 104 of file zip.c.

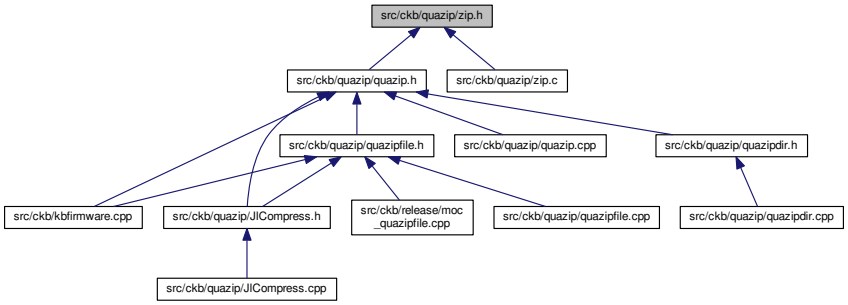
13.110 src/ckb/quazip/zip.h File Reference

```
#include "zlib.h"
#include "ioapi.h"
```

Include dependency graph for zip.h:



This graph shows which files directly or indirectly include this file:



Data Structures

- struct tm_zip_s
- struct zip_fileinfo

Macros

- #define Z_BZIP2ED 12
- #define ZIP_OK (0)
- #define ZIP_EOF (0)
- #define ZIP_ERRNO (Z_ERRNO)
- #define ZIP_PARAMERROR (-102)
- #define ZIP_BADZIPFILE (-103)
- #define ZIP_INTERNALERROR (-104)
- #define ZIP_WRITE_DATA_DESCRIPTOR 0x8u
- #define ZIP_AUTO_CLOSE 0x1u
- #define ZIP_SEQUENTIAL 0x2u
- #define ZIP_DEFAULT_FLAGS (ZIP_AUTO_CLOSE | ZIP_WRITE_DATA_DESCRIPTOR)
- #define DEF_MEM_LEVEL MAX_MEM_LEVEL
- #define APPEND_STATUS_CREATE (0)
- #define APPEND_STATUS_CREATEAFTER (1)
- #define APPEND_STATUS_ADDINZIP (2)

Typedefs

- typedef voidp zipFile
- typedef struct tm_zip_s tm_zip
- typedef const char * zipcharpc

Functions

- zipFile ZEXPORT zipOpen _Z_OF ((voidpf file, int append))
- zipFile ZEXPORT zipOpen2 _Z_OF ((voidpf file, int append, zipcharpc *globalcomment, zlib_filefunc_def *pzlib_filefunc_def))
- zipFile ZEXPORT zipOpen2_64 _Z_OF ((voidpf file, int append, zipcharpc *globalcomment, zlib_filefunc64_def *pzlib_filefunc_def))
- zipFile ZEXPORT zipOpen3 (voidpf file, int append, zipcharpc *globalcomment, zlib_filefunc64_32_def *pzlib_filefunc64_32_def, unsigned flags)

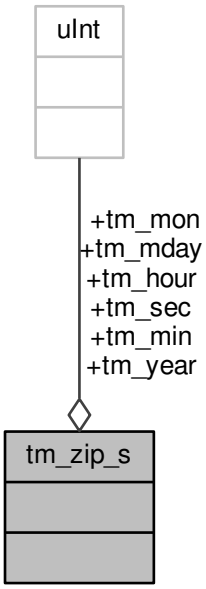
- int ZEXPORT zipOpenNewFileInZip _Z_OF ((zipFile file, const char ∗filename, const zip_fileinfo ∗zipfi, const void ∗extrafield_local, uInt size_extrafield_local, const void ∗extrafield_global, uInt size_extrafield_global, const char ∗comment, int method, int level))
- int ZEXPORT zipOpenNewFileInZip64 _Z_OF ((zipFile file, const char ∗filename, const zip_fileinfo ∗zipfi, const void ∗extrafield_local, uInt size_extrafield_local, const void ∗extrafield_global, uInt size_extrafield_global, const char ∗comment, int method, int level, int zip64))
- int ZEXPORT zipOpenNewFileInZip2 _Z_OF ((zipFile file, const char ∗filename, const zip_fileinfo ∗zipfi, const void ∗extrafield_local, uInt size_extrafield_local, const void ∗extrafield_global, uInt size_extrafield_global, const char ∗comment, int method, int level, int raw))
- int ZEXPORT zipOpenNewFileInZip2_64 _Z_OF ((zipFile file, const char ∗filename, const zip_fileinfo ∗zipfi, const void ∗extrafield_local, uInt size_extrafield_local, const void ∗extrafield_global, uInt size_extrafield_global, const char ∗comment, int method, int level, int raw, int zip64))
- int ZEXPORT zipOpenNewFileInZip3 _Z_OF ((zipFile file, const char ∗filename, const zip_fileinfo ∗zipfi, const void ∗extrafield_local, uInt size_extrafield_local, const void ∗extrafield_global, uInt size_extrafield_global, const char ∗comment, int method, int level, int raw, int windowBits, int memLevel, int strategy, const char ∗password, uLong crcForCrypting))
- int ZEXPORT zipOpenNewFileInZip3_64 _Z_OF ((zipFile file, const char ∗filename, const zip_fileinfo ∗zipfi, const void ∗extrafield_local, uInt size_extrafield_local, const void ∗extrafield_global, uInt size_extrafield_global, const char ∗comment, int method, int level, int raw, int windowBits, int memLevel, int strategy, const char ∗password, uLong crcForCrypting, int zip64))
- int ZEXPORT zipOpenNewFileInZip4 _Z_OF ((zipFile file, const char ∗filename, const zip_fileinfo ∗zipfi, const void ∗extrafield_local, uInt size_extrafield_local, const void ∗extrafield_global, uInt size_extrafield_global, const char ∗comment, int method, int level, int raw, int windowBits, int memLevel, int strategy, const char ∗password, uLong crcForCrypting, uLong versionMadeBy, uLong flagBase))
- int ZEXPORT zipOpenNewFileInZip4_64 _Z_OF ((zipFile file, const char ∗filename, const zip_fileinfo ∗zipfi, const void ∗extrafield_local, uInt size_extrafield_local, const void ∗extrafield_global, uInt size_extrafield_global, const char ∗comment, int method, int level, int raw, int windowBits, int memLevel, int strategy, const char ∗password, uLong crcForCrypting, uLong versionMadeBy, uLong flagBase, int zip64))
- int ZEXPORT zipWriteInFileInZip _Z_OF ((zipFile file, const void ∗buf, unsigned len))
- int ZEXPORT zipCloseFileInZip _Z_OF ((zipFile file))
- int ZEXPORT zipCloseFileInZipRaw _Z_OF ((zipFile file, uLong uncompressed_size, uLong crc32))
- int ZEXPORT zipCloseFileInZipRaw64 _Z_OF ((zipFile file, ZPOS64_T uncompressed_size, uLong crc32))
- int ZEXPORT zipClose _Z_OF ((zipFile file, const char ∗global_comment))
- int ZEXPORT zipRemoveExtraInfoBlock _Z_OF ((char ∗pData, int ∗dataLen, short sHeader))
- int ZEXPORT zipSetFlags (zipFile file, unsigned flags)
- int ZEXPORT zipClearFlags (zipFile file, unsigned flags)

13.110.1 Data Structure Documentation

13.110.1.1 struct tm_zip_s

Definition at line 100 of file zip.h.

Collaboration diagram for tm_zip_s:



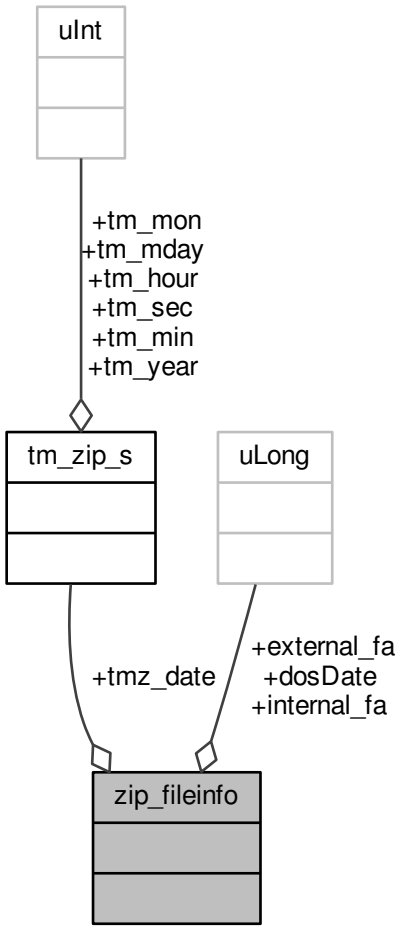
Data Fields

uInt	tm_hour	
uInt	tm_mday	
uInt	tm_min	
uInt	tm_mon	
uInt	tm_sec	
uInt	tm_year	

13.110.1.2 struct zip_fileinfo

Definition at line 110 of file zip.h.

Collaboration diagram for zip_fileinfo:



Data Fields

uLong	dosDate	
uLong	external_fa	
uLong	internal_fa	
tm_zip	tmz_date	

13.110.2 Macro Definition Documentation

13.110.2.1 #define APPEND_STATUS_ADDINZIP (2)

Definition at line 125 of file zip.h.

Referenced by QuaZip::open(), and zipOpen3().

13.110.2.2 #define APPEND_STATUS_CREATE (0)

Definition at line 123 of file zip.h.

Referenced by QuaZip::open(), and zipOpen3().

13.110.2.3 #define APPEND_STATUS_CREATEAFTER (1)

Definition at line 124 of file zip.h.

Referenced by QuaZip::open(), and zipOpen3().

13.110.2.4 #define DEF_MEM_LEVEL MAX_MEM_LEVEL

Definition at line 94 of file zip.h.

Referenced by zipOpenNewFileInZip(), zipOpenNewFileInZip2(), zipOpenNewFileInZip2_64(), and zipOpenNewFileInZip64().

13.110.2.5 #define Z_BZIP2ED 12

Definition at line 67 of file zip.h.

13.110.2.6 #define ZIP_AUTO_CLOSE 0x1u

Definition at line 86 of file zip.h.

Referenced by LoadCentralDirectoryRecord(), QuaZip::open(), zipClose(), and zipOpen3().

13.110.2.7 #define ZIP_BADZIPFILE (-103)

Definition at line 82 of file zip.h.

Referenced by LoadCentralDirectoryRecord(), and zipCloseFileInZipRaw64().

13.110.2.8 #define ZIP_DEFAULT_FLAGS (ZIP_AUTO_CLOSE | ZIP_WRITE_DATA_DESCRIPTOR)

Definition at line 88 of file zip.h.

Referenced by zipOpen(), zipOpen2(), zipOpen2_64(), and zipOpen64().

13.110.2.9 #define ZIP_EOF (0)

Definition at line 79 of file zip.h.

Referenced by zip64local_getByte().

13.110.2.10 #define ZIP_ERRNO (Z_ERRNO)

Definition at line 80 of file zip.h.

Referenced by LoadCentralDirectoryRecord(), Write_GlobalComment(), Write_LocalFileHeader(), zip64FlushWriteBuffer(), zip64local_getByte(), zip64local_putValue(), zipClose(), zipCloseFileInZipRaw64(), zipOpenNewFileInZip4_64(), zipRemoveExtraInfoBlock(), and zipWriteInFileInZip().

13.110.2.11 #define ZIP_INTERNALERROR (-104)

Definition at line 83 of file zip.h.

Referenced by add_data_in_datablock(), and zipOpenNewFileInZip4_64().

13.110.2.12 #define ZIP_OK (0)

Definition at line 78 of file zip.h.

Referenced by add_data_in_datablock(), LoadCentralDirectoryRecord(), Write_EndOfCentralDirectoryRecord(), Write_GlobalComment(), Write_LocalFileHeader(), Write_Zip64EndOfCentralDirectoryLocator(), Write_Zip64EndOfCentralDirectoryRecord(), QuaZipFile::writeData(), zip64FlushWriteBuffer(), zip64local_getByte(), zip64local_getLong(), zip64local_getLong64(), zip64local_getShort(), zip64local_putValue(), zip64local_SearchCentralDir64(), zipClearFlags(), zipClose(), zipCloseFileInZipRaw64(), zipOpen3(), zipOpenNewFileInZip4_64(), zipRemoveExtraInfoBlock(), zipSetFlags(), and zipWriteInFileInZip().

13.110.2.13 #define ZIP_PARAMERROR (-102)

Definition at line 81 of file zip.h.

Referenced by zipClearFlags(), zipClose(), zipCloseFileInZipRaw64(), zipOpenNewFileInZip4_64(), zipRemoveExtraInfoBlock(), zipSetFlags(), and zipWriteInFileInZip().

13.110.2.14 #define ZIP_SEQUENTIAL 0x2u

Definition at line 87 of file zip.h.

Referenced by QuaZip::open(), zipClearFlags(), zipCloseFileInZipRaw64(), zipOpenNewFileInZip4_64(), and zipSetFlags().

13.110.2.15 #define ZIP_WRITE_DATA_DESCRIPTOR 0x8u

Definition at line 85 of file zip.h.

Referenced by QuaZip::open(), QuaZipFile::open(), zipClearFlags(), zipOpenNewFileInZip4_64(), and zipSetFlags().

13.110.3 Typedef Documentation

13.110.3.1 typedef struct tm_zip_s tm_zip

13.110.3.2 typedef const char* zipcharpc

Definition at line 120 of file zip.h.

13.110.3.3 typedef voidp zipFile

Definition at line 75 of file zip.h.

13.110.4 Function Documentation

13.110.4.1 zipFile ZEXPORT zipOpen _Z_OF ((voidpf file, int append))

13.110.4.2 zipFile ZEXPORT zipOpen2 _Z_OF ((voidpf file, int append, zipcharpc *globalcomment, zlib_filefunc_def *pzlib_filefunc_def))

13.110.4.3 zipFile ZEXPORT zipOpen2_64 _Z_OF ((voidpf file, int append, zipcharpc *globalcomment, zlib_filefunc64_def *pzlib_filefunc_def))

13.110.4.4 int ZEXPORT zipOpenNewFileInZip _Z_OF ((zipFile file, const char ∗filename, const zip_fileinfo ∗zipfi, const void ∗extrafield_local, uInt size_extrafield_local, const void ∗extrafield_global, uInt size_extrafield_global, const char ∗comment, int method, int level))

13.110.4.5 int ZEXPORT zipOpenNewFileInZip64 _Z_OF ((zipFile file, const char ∗filename, const zip_fileinfo ∗zipfi, const void ∗extrafield_local, uInt size_extrafield_local, const void ∗extrafield_global, uInt size_extrafield_global, const char ∗comment, int method, int level, int zip64))

13.110.4.6 int ZEXPORT zipOpenNewFileInZip2 _Z_OF ((zipFile file, const char ∗filename, const zip_fileinfo ∗zipfi, const void ∗extrafield_local, uInt size_extrafield_local, const void ∗extrafield_global, uInt size_extrafield_global, const char ∗comment, int method, int level, int raw))

13.110.4.7 int ZEXPORT zipOpenNewFileInZip2_64 _Z_OF ((zipFile file, const char ∗filename, const zip_fileinfo ∗zipfi, const void ∗extrafield_local, uInt size_extrafield_local, const void ∗extrafield_global, uInt size_extrafield_global, const char ∗comment, int method, int level, int raw, int zip64))

13.110.4.8 int ZEXPORT zipOpenNewFileInZip3 _Z_OF ((zipFile file, const char ∗filename, const zip_fileinfo ∗zipfi, const void ∗extrafield_local, uInt size_extrafield_local, const void ∗extrafield_global, uInt size_extrafield_global, const char ∗comment, int method, int level, int raw, int windowBits, int memLevel, int strategy, const char ∗password, uLong crcForCrypting))

13.110.4.9 int ZEXPORT zipOpenNewFileInZip3_64 _Z_OF ((zipFile file, const char ∗filename, const zip_fileinfo ∗zipfi, const void ∗extrafield_local, uInt size_extrafield_local, const void ∗extrafield_global, uInt size_extrafield_global, const char ∗comment, int method, int level, int raw, int windowBits, int memLevel, int strategy, const char ∗password, uLong crcForCrypting, int zip64))

13.110.4.10 int ZEXPORT zipOpenNewFileInZip4 _Z_OF ((zipFile file, const char ∗filename, const zip_fileinfo ∗zipfi, const void ∗extrafield_local, uInt size_extrafield_local, const void ∗extrafield_global, uInt size_extrafield_global, const char ∗comment, int method, int level, int raw, int windowBits, int memLevel, int strategy, const char ∗password, uLong crcForCrypting, uLong versionMadeBy, uLong flagBase))

13.110.4.11 int ZEXPORT zipOpenNewFileInZip4_64 _Z_OF ((zipFile file, const char ∗filename, const zip_fileinfo ∗zipfi, const void ∗extrafield_local, uInt size_extrafield_local, const void ∗extrafield_global, uInt size_extrafield_global, const char ∗comment, int method, int level, int raw, int windowBits, int memLevel, int strategy, const char ∗password, uLong crcForCrypting, uLong versionMadeBy, uLong flagBase, int zip64))

13.110.4.12 int ZEXPORT zipWriteInFileInZip _Z_OF ((zipFile file, const void ∗buf, unsigned len))

13.110.4.13 int ZEXPORT zipCloseFileInZip _Z_OF ((zipFile file))

13.110.4.14 int ZEXPORT zipCloseFileInZipRaw _Z_OF ((zipFile file, uLong uncompressed_size, uLong crc32))

13.110.4.15 int ZEXPORT zipCloseFileInZipRaw64 _Z_OF ((zipFile file, ZPOS64_T uncompressed_size, uLong crc32))

13.110.4.16 int ZEXPORT zipClose _Z_OF ((zipFile file, const char ∗global_comment))

13.110.4.17 int ZEXPORT zipRemoveExtraInfoBlock _Z_OF ((char ∗pData, int ∗dataLen, short sHeader))

13.110.4.18 int ZEXPORT zipClearFlags (zipFile *file,* unsigned *flags*)

Definition at line 2091 of file zip.c.

References zip64_internal::flags, ZIP_OK, ZIP_PARAMERROR, ZIP_SEQUENTIAL, and ZIP_WRITE_DATA_DESCRIPTOR.

Referenced by QuaZipFile::open().

```
2092 {
2093     zip64_internal* zi;
```

```
2094     if (file == NULL)
2095         return ZIP_PARAMERROR;
2096     zi = (zip64_internal*)file;
2097     zi->flags &= ~flags;
2098     // If the data descriptor is not written, we can't use a non-seekable output.
2099     if ((zi->flags & ZIP_WRITE_DATA_DESCRIPTOR) == 0) {
2100         zi->flags &= ~ZIP_SEQUENTIAL;
2101     }
2102     return ZIP_OK;
2103 }
```

Here is the caller graph for this function:

13.110.4.19 zipFile ZEXPORT zipOpen3 (voidpf *file*, int *append*, zipcharpc * *globalcomment*, zlib_filefunc64_32_def * *pzlib_filefunc64_32_def*, unsigned *flags*)

Definition at line 862 of file zip.c.

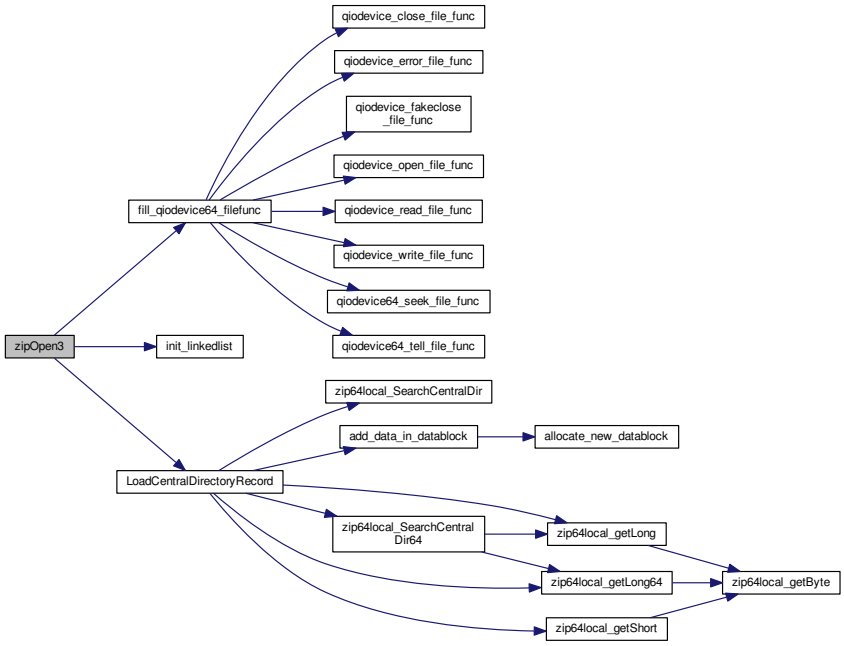
References zip64_internal::add_position_when_writting_offset, ALLOC, APPEND_STATUS_ADDINZIP, APPEND_STATUS_CREATE, APPEND_STATUS_CREATEAFTER, zip64_internal::begin_pos, zip64_internal::central_dir, zip64_internal::ci, zip64_internal::filestream, fill_qiodevice64_filefunc(), zip64_internal::flags, zip64_internal::globalcomment, zip64_internal::in_opened_file_inzip, init_linkedlist(), LoadCentralDirectoryRecord(), zip64_internal::number_entry, SEEK_END, curfile64_info::stream_initialised, TRYFREE, zip64_internal::z_filefunc, ZCLOSE64, ZFAKECLOSE64, zlib_filefunc64_32_def_s::zfile_func64, ZIP_AUTO_CLOSE, ZIP_OK, ZLIB_FILEFUNC_MODE_CREATE, ZLIB_FILEFUNC_MODE_EXISTING, ZLIB_FILEFUNC_MODE_READ, ZLIB_FILEFUNC_MODE_WRITE, ZOPEN64, zlib_filefunc64_32_def_s::zseek32_file, ZSEEK64, zlib_filefunc64_32_def_s::ztell32_file, and ZTELL64.

Referenced by QuaZip::open(), zipOpen(), zipOpen2(), zipOpen2_64(), and zipOpen64().

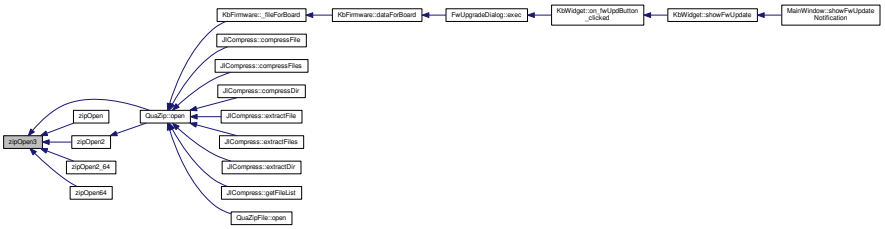
```
864 {
865     zip64_internal ziinit;
866     zip64_internal* zi;
867     int err=ZIP_OK;
868 
869     ziinit.flags = flags;
870     ziinit.z_filefunc.zseek32_file = NULL;
871     ziinit.z_filefunc.ztell32_file = NULL;
872     if (pzlib_filefunc64_32_def==NULL)
873         fill_qiodevice64_filefunc(&ziinit.z_filefunc.
      zfile_func64);
874     else
875         ziinit.z_filefunc = *pzlib_filefunc64_32_def;
876 
877     ziinit.filestream = ZOPEN64(ziinit.z_filefunc,
878                 file,
879                 (append == APPEND_STATUS_CREATE) ?
880                 (ZLIB_FILEFUNC_MODE_READ |
      ZLIB_FILEFUNC_MODE_WRITE | ZLIB_FILEFUNC_MODE_CREATE) :
881                   (ZLIB_FILEFUNC_MODE_READ |
      ZLIB_FILEFUNC_MODE_WRITE | ZLIB_FILEFUNC_MODE_EXISTING))
      ;
882 
883     if (ziinit.filestream == NULL)
884         return NULL;
885 
886     if (append == APPEND_STATUS_CREATEAFTER)
887         ZSEEK64(ziinit.z_filefunc,ziinit.filestream,0,
      SEEK_END);
888 
```

```
889     ziinit.begin_pos = ZTELL64(ziinit.z_filefunc,ziinit.
      filestream);
890     ziinit.in_opened_file_inzip = 0;
891     ziinit.ci.stream_initialised = 0;
892     ziinit.number_entry = 0;
893     ziinit.add_position_when_writting_offset = 0;
894     init_linkedlist(&(ziinit.central_dir));
895
896
897
898     zi = (zip64_internal*)ALLOC(sizeof(zip64_internal));
899     if (zi==NULL)
900     {
901         if ((ziinit.flags & ZIP_AUTO_CLOSE) != 0) {
902             ZCLOSE64(ziinit.z_filefunc,ziinit.filestream);
903         } else {
904             ZFAKECLOSE64(ziinit.z_filefunc,ziinit.
      filestream);
905         }
906         return NULL;
907     }
908
909     /* now we add file in a zipfile */
910 #    ifndef NO_ADDFILEINEXISTINGZIP
911     ziinit.globalcomment = NULL;
912     if (append == APPEND_STATUS_ADDINZIP)
913     {
914       /* Read and Cache Central Directory Records */
915       err = LoadCentralDirectoryRecord(&ziinit);
916     }
917
918     if (globalcomment)
919     {
920       *globalcomment = ziinit.globalcomment;
921     }
922 #    endif /* !NO_ADDFILEINEXISTINGZIP*/
923
924     if (err != ZIP_OK)
925     {
926 #    ifndef NO_ADDFILEINEXISTINGZIP
927         TRYFREE(ziinit.globalcomment);
928 #    endif /* !NO_ADDFILEINEXISTINGZIP*/
929         TRYFREE(zi);
930         return NULL;
931     }
932     else
933     {
934         *zi = ziinit;
935         return (zipFile)zi;
936     }
937 }
```

Here is the call graph for this function:



Here is the caller graph for this function:



13.110.4.20 int ZEXPORT zipSetFlags (zipFile *file*, unsigned *flags*)

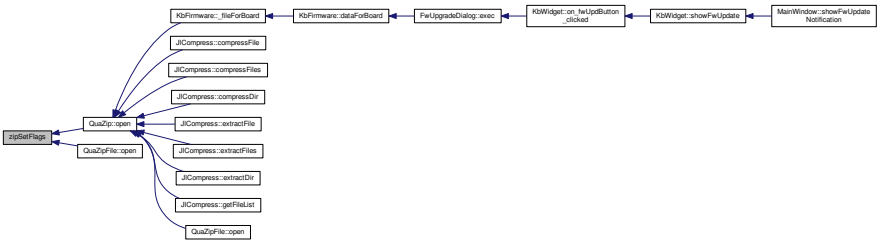
Definition at line 2077 of file zip.c.

References zip64_internal::flags, ZIP_OK, ZIP_PARAMERROR, ZIP_SEQUENTIAL, and ZIP_WRITE_DATA_DESCRIPTOR.

Referenced by QuaZip::open(), and QuaZipFile::open().

```
2078 {
2079     zip64_internal* zi;
2080     if (file == NULL)
2081         return ZIP_PARAMERROR;
2082     zi = (zip64_internal*)file;
2083     zi->flags |= flags;
2084     // If the output is non-seekable, the data descriptor is needed.
2085     if ((zi->flags & ZIP_SEQUENTIAL) != 0) {
2086         zi->flags |= ZIP_WRITE_DATA_DESCRIPTOR;
2087     }
2088     return ZIP_OK;
2089 }
```

Here is the caller graph for this function:



13.111 src/ckb/rebindwidget.cpp File Reference

```
#include <QStandardPaths>
#include "rebindwidget.h"
#include "ui_rebindwidget.h"
#include <qdebug.h>
```

Include dependency graph for rebindwidget.cpp:

Variables

- static const int DPI_OFFSET = -KeyAction::DPI_CYCLE_UP + 1
- static const int DPI_CUST_IDX = KeyAction::DPI_CUSTOM + DPI_OFFSET

13.111.1 Variable Documentation

13.111.1.1 const int DPI_CUST_IDX = KeyAction::DPI_CUSTOM + DPI_OFFSET [static]

Definition at line 7 of file rebindwidget.cpp.

Referenced by RebindWidget::on_dpiBox_currentIndexChanged().

13.111.1.2 const int DPI_OFFSET = -KeyAction::DPI_CYCLE_UP + 1 [static]

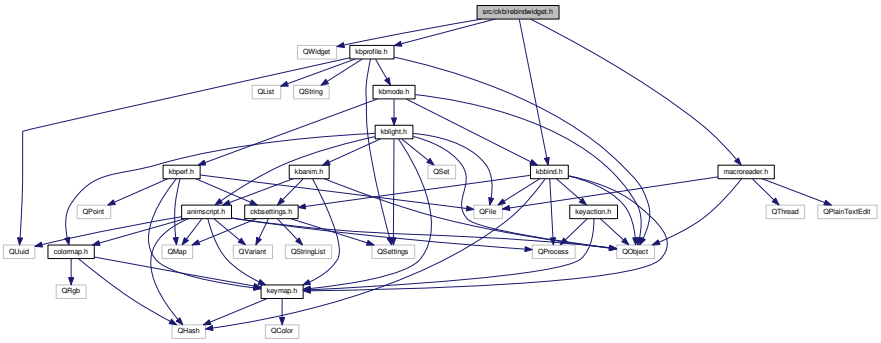
Definition at line 6 of file rebindwidget.cpp.

Referenced by RebindWidget::applyChanges(), RebindWidget::setBind(), and RebindWidget::setSelection().

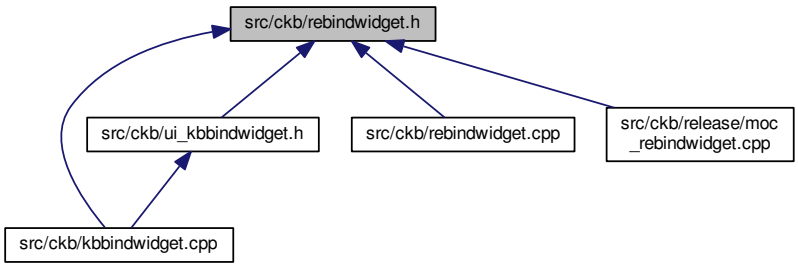
13.112 src/ckb/rebindwidget.h File Reference

```
#include <QWidget>
#include "kbbind.h"
#include "kbprofile.h"
#include "macroreader.h"
```

Include dependency graph for rebindwidget.h:



This graph shows which files directly or indirectly include this file:



Data Structures

- class RebindWidget

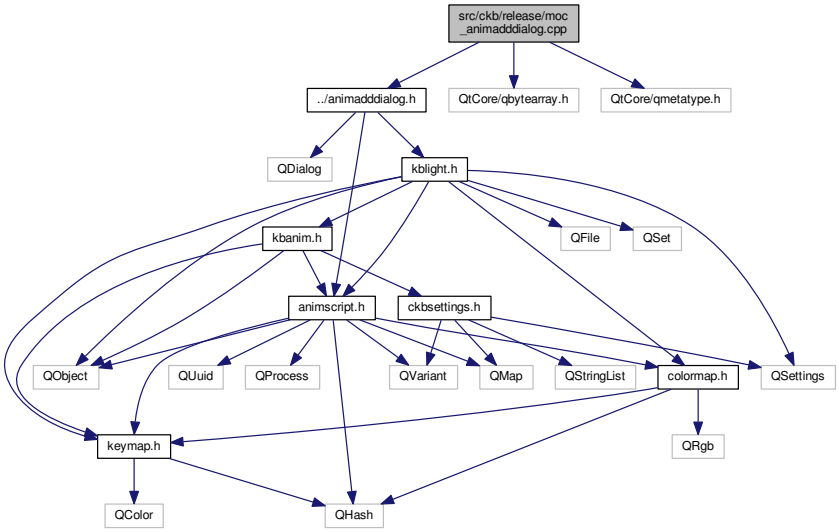
Namespaces

- Ui

13.113 src/ckb/release/moc_animadddialog.cpp File Reference

```
#include "../animadddialog.h"
#include <QtCore/qbytearray.h>
#include <QtCore/qmetatype.h>
```

Include dependency graph for moc_animadddialog.cpp:



Data Structures

- struct qt_meta_stringdata_AnimAddDialog_t

Macros

- #define QT_MOC_LITERAL(idx, ofs, len)

Variables

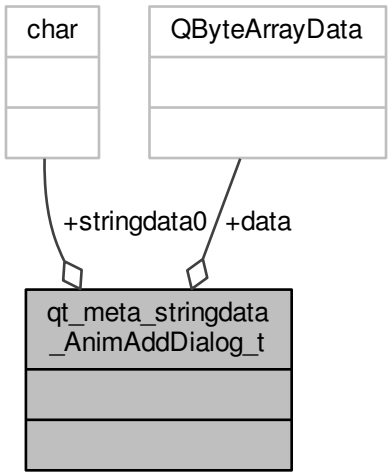
- static const qt_meta_stringdata_AnimAddDialog_t qt_meta_stringdata_AnimAddDialog
- static const uint qt_meta_data_AnimAddDialog []

13.113.1 Data Structure Documentation

13.113.1.1 struct qt_meta_stringdata_AnimAddDialog_t

Definition at line 21 of file moc_animadddialog.cpp.

Collaboration diagram for qt_meta_stringdata_AnimAddDialog_t:



Data Fields

QByteArrayData	data[7]	
char	stringdata0[95]	

13.113.2 Macro Definition Documentation

13.113.2.1 #define QT_MOC_LITERAL(*idx, ofs, len*)

Value:

```
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET(len, \
    qptrdiff(offsetof(qt_meta_stringdata_AnimAddDialog_t, stringdata0) +
      ofs \
        - idx * sizeof(QByteArrayData)) \
    )
```

Definition at line 25 of file moc_animadddialog.cpp.

13.113.3 Variable Documentation

13.113.3.1 const uint qt_meta_data_AnimAddDialog[] [static]

Initial value:

```
= {

       7,
       0,
       0,    0,
       3,   14,
       0,    0,
       0,    0,
       0,    0,
```

```
       0,
       0,


       1,    1,   29,    2, 0x08 ,
       4,    1,   32,    2, 0x08 ,
       5,    1,   35,    2, 0x08 ,


    QMetaType::Void, QMetaType::Int,    3,
    QMetaType::Void, QMetaType::Int,    3,
    QMetaType::Void, QMetaType::Bool,    6,

       0
}
```

Definition at line 47 of file moc_animadddialog.cpp.

13.113.3.2 const qt_meta_stringdata_AnimAddDialog_t qt_meta_stringdata_AnimAddDialog [static]

Initial value:

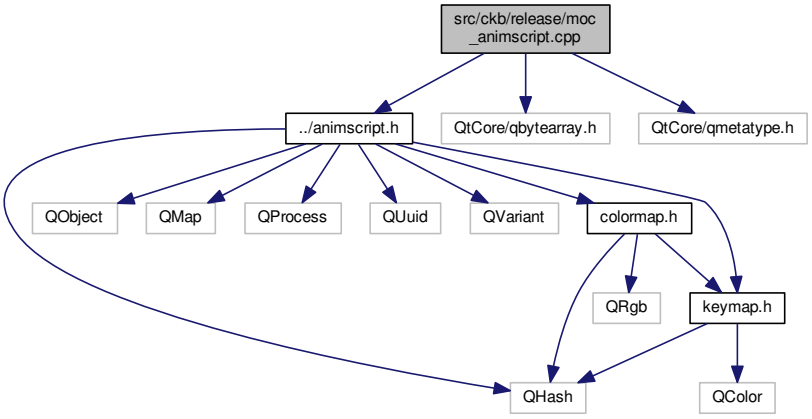
```
= {
    {
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 13 ,   qptrdiff(offsetof(
      qt_meta_stringdata_AnimAddDialog_t, stringdata0) +  0    -  0  * sizeof(
      QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 20 ,   qptrdiff(offsetof(
      qt_meta_stringdata_AnimAddDialog_t, stringdata0) +  14    -  1  * sizeof(
      QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 0 ,   qptrdiff(offsetof(
      qt_meta_stringdata_AnimAddDialog_t, stringdata0) +  35    -  2  * sizeof(
      QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 5 ,   qptrdiff(offsetof(
      qt_meta_stringdata_AnimAddDialog_t, stringdata0) +  36    -  3  * sizeof(
      QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 22 ,   qptrdiff(offsetof(
      qt_meta_stringdata_AnimAddDialog_t, stringdata0) +  42    -  4  * sizeof(
      QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 21 ,   qptrdiff(offsetof(
      qt_meta_stringdata_AnimAddDialog_t, stringdata0) +  65    -  5  * sizeof(
      QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 7 ,   qptrdiff(offsetof(
      qt_meta_stringdata_AnimAddDialog_t, stringdata0) +  87    -  6  * sizeof(
      QByteArrayData))   )

    },
    "AnimAddDialog\0on_animBox_activated\0\0"
    "index\0on_presetBox_activated\0"
    "on_previewBox_clicked\0checked"
}
```

Definition at line 30 of file moc_animadddialog.cpp.

13.114 src/ckb/release/moc_animscript.cpp File Reference

```
#include "../animscript.h"
#include <QtCore/qbytearray.h>
#include <QtCore/qmetatype.h>
```

Include dependency graph for moc_animscript.cpp:



Data Structures

- struct qt_meta_stringdata_AnimScript_t

Macros

- #define QT_MOC_LITERAL(idx, ofs, len)

Variables

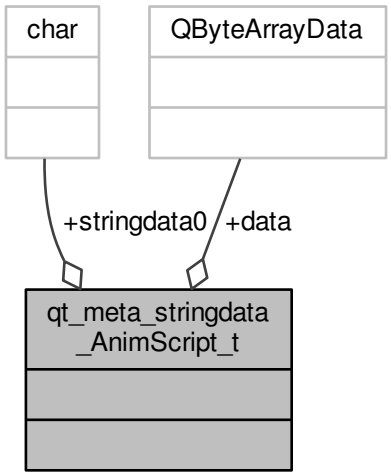
- static const qt_meta_stringdata_AnimScript_t qt_meta_stringdata_AnimScript
- static const uint qt_meta_data_AnimScript [ ]

13.114.1 Data Structure Documentation

13.114.1.1 struct qt_meta_stringdata_AnimScript_t

Definition at line 21 of file moc_animscript.cpp.

Collaboration diagram for qt_meta_stringdata_AnimScript_t:



Data Fields

QByteArrayData	data[3]	
char	stringdata0[24]	

13.114.2 Macro Definition Documentation

13.114.2.1 #define QT_MOC_LITERAL(*idx, ofs, len*)

Value:

```
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET(len, \
    qptrdiff(offsetof(qt_meta_stringdata_AnimScript_t, stringdata0) + ofs \
        - idx * sizeof(QByteArrayData)) \
    )
```

Definition at line 25 of file moc_animscript.cpp.

13.114.3 Variable Documentation

13.114.3.1 const uint qt_meta_data_AnimScript[] [static]

Initial value:

```
= {

       7,
       0,
       0,    0,
       1,   14,
       0,    0,
       0,    0,
       0,    0,
       0,
```

```
        0,

        1,    0,   19,    2, 0x08 ,

    QMetaType::Void,

        0
}
```

Definition at line 41 of file moc_animscript.cpp.

13.114.3.2 const qt_meta_stringdata_AnimScript_t qt_meta_stringdata_AnimScript [static]

Initial value:

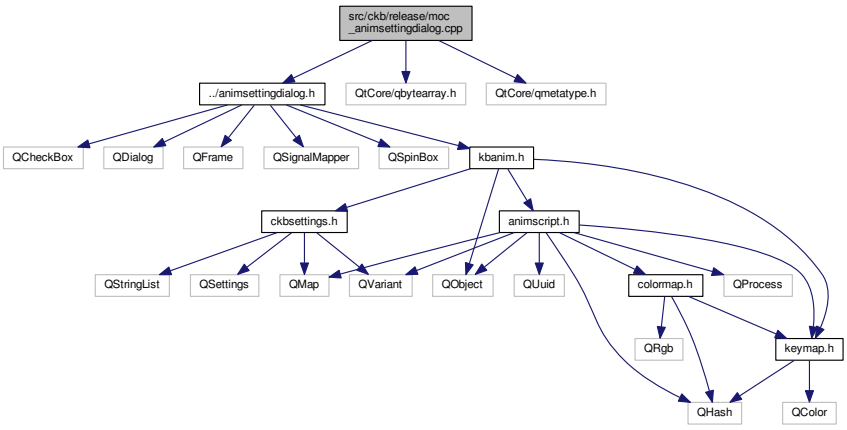
```
= {
    {
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 10 ,   qptrdiff(offsetof(
      qt_meta_stringdata_AnimScript_t, stringdata0) +  0    -  0  * sizeof(
      QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 11 ,   qptrdiff(offsetof(
      qt_meta_stringdata_AnimScript_t, stringdata0) +  11    -  1  * sizeof(
      QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 0 ,   qptrdiff(offsetof(
      qt_meta_stringdata_AnimScript_t, stringdata0) +  23    -  2  * sizeof(
      QByteArrayData))   )

    },
    "AnimScript\0readProcess\0"
}
```

Definition at line 30 of file moc_animscript.cpp.

13.115 src/ckb/release/moc_animsettingdialog.cpp File Reference

```
#include "../animsettingdialog.h"
#include <QtCore/qbytearray.h>
#include <QtCore/qmetatype.h>
```

Include dependency graph for moc_animsettingdialog.cpp:



Data Structures

- struct qt_meta_stringdata_AnimSettingDialog_t

Macros

- #define QT_MOC_LITERAL(idx, ofs, len)

Variables

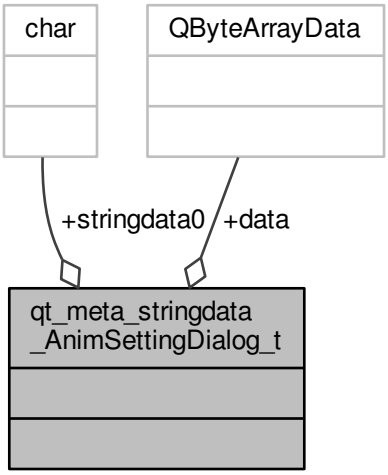
- static const qt_meta_stringdata_AnimSettingDialog_t qt_meta_stringdata_AnimSettingDialog
- static const uint qt_meta_data_AnimSettingDialog []

13.115.1 Data Structure Documentation

13.115.1.1 struct qt_meta_stringdata_AnimSettingDialog_t

Definition at line 21 of file moc_animsettingdialog.cpp.

Collaboration diagram for qt_meta_stringdata_AnimSettingDialog_t:



Data Fields

QByteArrayData	data[15]	
char	stringdata0[246]	

13.115.2 Macro Definition Documentation

13.115.2.1 #define QT_MOC_LITERAL(*idx, ofs, len*)

Value:

```
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET(len, \
    qptrdiff(offsetof(qt_meta_stringdata_AnimSettingDialog_t,
      stringdata0) + ofs \
        - idx * sizeof(QByteArrayData)) \
    )
```

Definition at line 25 of file moc_animsettingdialog.cpp.

13.115.3 Variable Documentation

13.115.3.1 const uint qt_meta_data_AnimSettingDialog[] [static]

Definition at line 60 of file moc_animsettingdialog.cpp.

13.115.3.2 const qt_meta_stringdata_AnimSettingDialog_t qt_meta_stringdata_AnimSettingDialog [static]

Initial value:

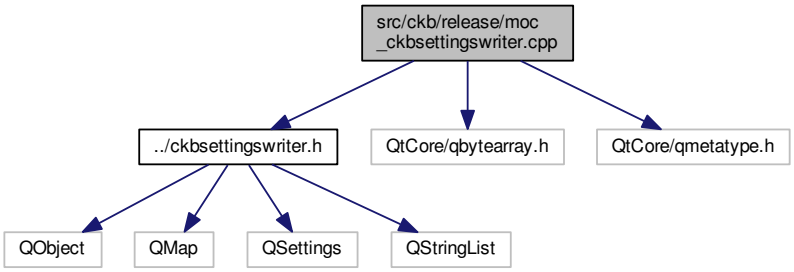
```
= {
    {
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 17 ,   qptrdiff(offsetof(
      qt_meta_stringdata_AnimSettingDialog_t, stringdata0) +  0    -  0  *
      sizeof(QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 11 ,   qptrdiff(offsetof(
      qt_meta_stringdata_AnimSettingDialog_t, stringdata0) +  18    -  1  *
       sizeof(QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 0 ,   qptrdiff(offsetof(
      qt_meta_stringdata_AnimSettingDialog_t, stringdata0) +  30    -  2  *
       sizeof(QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 8 ,   qptrdiff(offsetof(
      qt_meta_stringdata_AnimSettingDialog_t, stringdata0) +  31    -  3  *
       sizeof(QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 11 ,   qptrdiff(offsetof(
      qt_meta_stringdata_AnimSettingDialog_t, stringdata0) +  40    -  4  *
       sizeof(QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 16 ,   qptrdiff(offsetof(
      qt_meta_stringdata_AnimSettingDialog_t, stringdata0) +  52    -  5  *
       sizeof(QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 4 ,   qptrdiff(offsetof(
      qt_meta_stringdata_AnimSettingDialog_t, stringdata0) +  69    -  6  *
       sizeof(QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 19 ,   qptrdiff(offsetof(
      qt_meta_stringdata_AnimSettingDialog_t, stringdata0) +  74    -  7  *
       sizeof(QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 11 ,   qptrdiff(offsetof(
      qt_meta_stringdata_AnimSettingDialog_t, stringdata0) +  94    -  8  *
       sizeof(QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 24 ,   qptrdiff(offsetof(
      qt_meta_stringdata_AnimSettingDialog_t, stringdata0) +  106    -  9
      * sizeof(QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 4 ,   qptrdiff(offsetof(
      qt_meta_stringdata_AnimSettingDialog_t, stringdata0) +  131    -  10
       * sizeof(QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 25 ,   qptrdiff(offsetof(
      qt_meta_stringdata_AnimSettingDialog_t, stringdata0) +  136    -  11
       * sizeof(QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 26 ,   qptrdiff(offsetof(
      qt_meta_stringdata_AnimSettingDialog_t, stringdata0) +  162    -  12
       * sizeof(QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 27 ,   qptrdiff(offsetof(
      qt_meta_stringdata_AnimSettingDialog_t, stringdata0) +  189    -  13
       * sizeof(QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 28 ,   qptrdiff(offsetof(
      qt_meta_stringdata_AnimSettingDialog_t, stringdata0) +  217    -  14
       * sizeof(QByteArrayData))   )

    },
    "AnimSettingDialog\0newDuration\0\0duration\0"
    "updateStops\0angleDialChanged\0name\0"
    "angleSpinnerChanged\0updateParam\0"
    "on_delayBox_valueChanged\0arg1\0"
    "on_repeatBox_valueChanged\0"
    "on_kpDelayBox_valueChanged\0"
    "on_kpRepeatBox_valueChanged\0"
    "on_kpReleaseBox_stateChanged"
}
```

Definition at line 30 of file moc_animsettingdialog.cpp.

13.116 src/ckb/release/moc_ckbsettingswriter.cpp File Reference

```
#include "../ckbsettingswriter.h"
#include <QtCore/qbytearray.h>
#include <QtCore/qmetatype.h>
```

Include dependency graph for moc_ckbsettingswriter.cpp:



Data Structures

- struct qt_meta_stringdata_CkbSettingsWriter_t

Macros

- #define QT_MOC_LITERAL(idx, ofs, len)

Variables

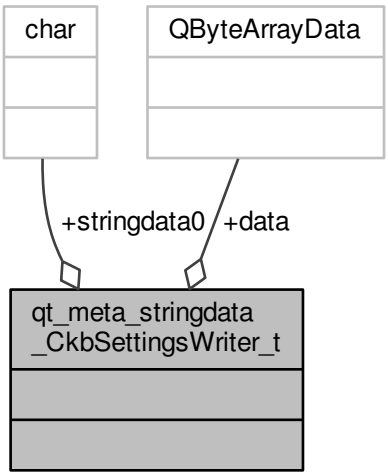
- static const qt_meta_stringdata_CkbSettingsWriter_t qt_meta_stringdata_CkbSettingsWriter
- static const uint qt_meta_data_CkbSettingsWriter [ ]

13.116.1 Data Structure Documentation

13.116.1.1 struct qt_meta_stringdata_CkbSettingsWriter_t

Definition at line 21 of file moc_ckbsettingswriter.cpp.

Collaboration diagram for qt_meta_stringdata_CkbSettingsWriter_t:



Data Fields

QByteArrayData	data[3]	
char	stringdata0[23]	

13.116.2 Macro Definition Documentation

13.116.2.1 #define QT_MOC_LITERAL(*idx, ofs, len*)

Value:

```
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET(len, \
    qptrdiff(offsetof(qt_meta_stringdata_CkbSettingsWriter_t,
      stringdata0) + ofs \
        - idx * sizeof(QByteArrayData)) \
    )
```

Definition at line 25 of file moc_ckbsettingswriter.cpp.

13.116.3 Variable Documentation

13.116.3.1 const uint qt_meta_data_CkbSettingsWriter[] [static]

Initial value:

```
= {

       7,
       0,
       0,    0,
       1,   14,
       0,    0,
       0,    0,
       0,    0,
```

```
        0,
        0,

        1,    0,   19,    2, 0x0a ,

    QMetaType::Void,

        0
}
```

Definition at line 41 of file moc_ckbsettingswriter.cpp.

13.116.3.2 const qt_meta_stringdata_CkbSettingsWriter_t qt_meta_stringdata_CkbSettingsWriter [static]

Initial value:

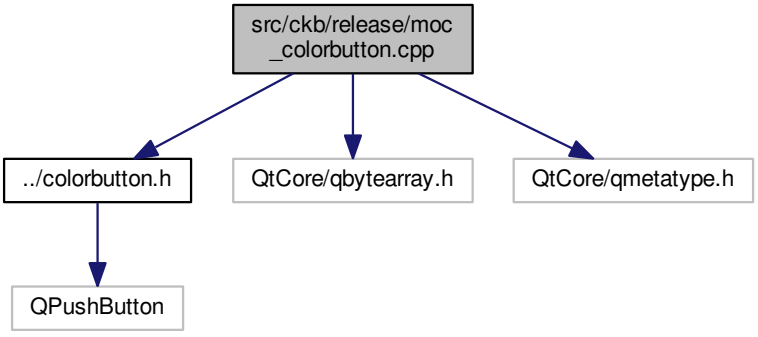
```
= {
    {
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 17 ,   qptrdiff(offsetof(
      qt_meta_stringdata_CkbSettingsWriter_t, stringdata0) +  0    -  0  *
      sizeof(QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 3 ,   qptrdiff(offsetof(
      qt_meta_stringdata_CkbSettingsWriter_t, stringdata0) +  18    -  1  *
       sizeof(QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 0 ,   qptrdiff(offsetof(
      qt_meta_stringdata_CkbSettingsWriter_t, stringdata0) +  22    -  2  *
       sizeof(QByteArrayData))   )

    },
    "CkbSettingsWriter\0run\0"
}
```

Definition at line 30 of file moc_ckbsettingswriter.cpp.

13.117 src/ckb/release/moc_colorbutton.cpp File Reference

```
#include "../colorbutton.h"
#include <QtCore/qbytearray.h>
#include <QtCore/qmetatype.h>
```

Include dependency graph for moc_colorbutton.cpp:



Data Structures

- struct qt_meta_stringdata_ColorButton_t

Macros

- #define QT_MOC_LITERAL(idx, ofs, len)

Variables

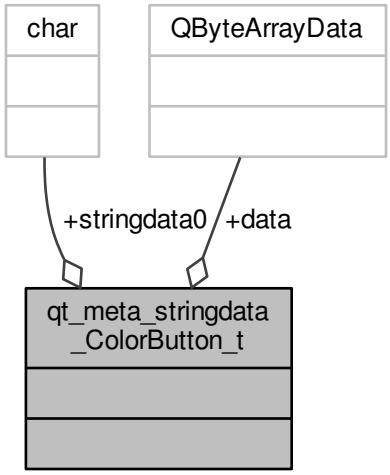
- static const qt_meta_stringdata_ColorButton_t qt_meta_stringdata_ColorButton
- static const uint qt_meta_data_ColorButton [ ]

13.117.1 Data Structure Documentation

13.117.1.1 struct qt_meta_stringdata_ColorButton_t

Definition at line 21 of file moc_colorbutton.cpp.

Collaboration diagram for qt_meta_stringdata_ColorButton_t:



Data Fields

QByteArrayData	data[5]	
char	stringdata0[45]	

13.117.2 Macro Definition Documentation

13.117.2.1 #define QT_MOC_LITERAL(*idx, ofs, len*)

Value:

```
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET(len, \
    qptrdiff(offsetof(qt_meta_stringdata_ColorButton_t, stringdata0) + ofs
      \
        - idx * sizeof(QByteArrayData)) \
    )
```

Definition at line 25 of file moc_colorbutton.cpp.

13.117.3 Variable Documentation

13.117.3.1 const uint qt_meta_data_ColorButton[] [static]

Initial value:

```
= {

       7,
       0,
       0,    0,
       2,   14,
       0,    0,
       0,    0,
       0,    0,
       0,
       1,


       1,    1,   24,    2, 0x06 ,


       4,    0,   27,    2, 0x08 ,


    QMetaType::Void, QMetaType::QColor,    3,


    QMetaType::Void,

       0
}
```

Definition at line 44 of file moc_colorbutton.cpp.

13.117.3.2 const qt_meta_stringdata_ColorButton_t qt_meta_stringdata_ColorButton [static]

Initial value:

```
= {
    {
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 11 ,   qptrdiff(offsetof(
      qt_meta_stringdata_ColorButton_t, stringdata0) +  0    -  0  * sizeof(
      QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 12 ,   qptrdiff(offsetof(
      qt_meta_stringdata_ColorButton_t, stringdata0) +  12    -  1  * sizeof(
      QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 0 ,   qptrdiff(offsetof(
      qt_meta_stringdata_ColorButton_t, stringdata0) +  25    -  2  * sizeof(
      QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 8 ,   qptrdiff(offsetof(
      qt_meta_stringdata_ColorButton_t, stringdata0) +  26    -  3  * sizeof(
      QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 9 ,   qptrdiff(offsetof(
      qt_meta_stringdata_ColorButton_t, stringdata0) +  35    -  4  * sizeof(
      QByteArrayData))   )

    },
    "ColorButton\0colorChanged\0\0newColor\0"
    "pickColor"
}
```

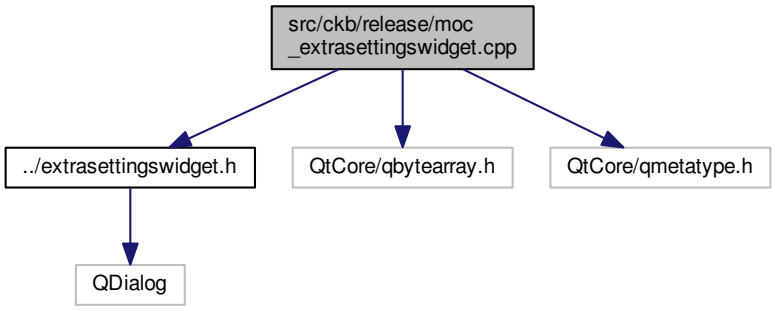
Definition at line 30 of file moc_colorbutton.cpp.

13.118 src/ckb/release/moc_extrasettingswidget.cpp File Reference

```
#include "../extrasettingswidget.h"
```

```
#include <QtCore/qbytearray.h>
#include <QtCore/qmetatype.h>
```

Include dependency graph for moc_extrasettingswidget.cpp:



Data Structures

- struct qt_meta_stringdata_ExtraSettingsWidget_t

Macros

- #define QT_MOC_LITERAL(idx, ofs, len)

Variables

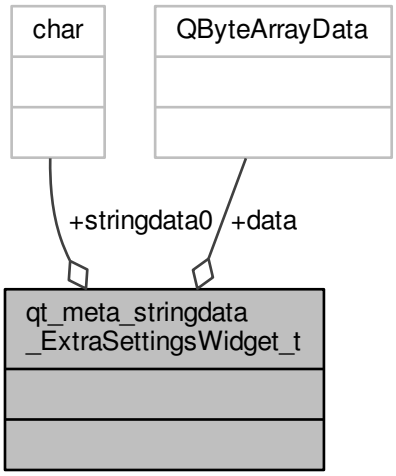
- static const qt_meta_stringdata_ExtraSettingsWidget_t qt_meta_stringdata_ExtraSettingsWidget
- static const uint qt_meta_data_ExtraSettingsWidget [ ]

13.118.1 Data Structure Documentation

13.118.1.1 struct qt_meta_stringdata_ExtraSettingsWidget_t

Definition at line 21 of file moc_extrasettingswidget.cpp.

Collaboration diagram for qt_meta_stringdata_ExtraSettingsWidget_t:



Data Fields

QByteArrayData	data[13]	
char	stringdata0[236]	

13.118.2 Macro Definition Documentation

13.118.2.1 #define QT_MOC_LITERAL(*idx, ofs, len*)

Value:

```
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET(len, \
    qptrdiff(offsetof(qt_meta_stringdata_ExtraSettingsWidget_t,
      stringdata0) + ofs \
        - idx * sizeof(QByteArrayData)) \
    )
```

Definition at line 25 of file moc_extrasettingswidget.cpp.

13.118.3 Variable Documentation

13.118.3.1 const uint qt_meta_data_ExtraSettingsWidget[ ] [static]

Definition at line 57 of file moc_extrasettingswidget.cpp.

13.118.3.2 const qt_meta_stringdata_ExtraSettingsWidget_t qt_meta_stringdata_ExtraSettingsWidget [static]

Initial value:

```
= {
    {
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 19 ,   qptrdiff(offsetof(
```

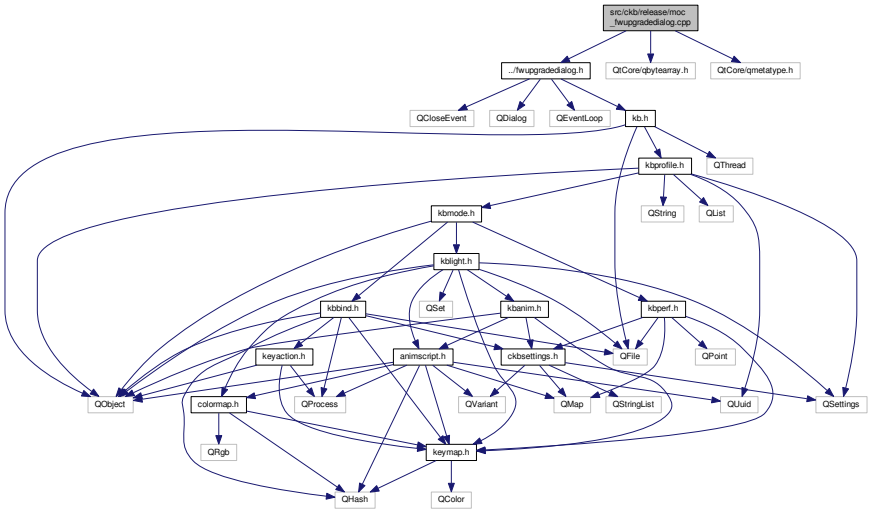
```
      qt_meta_stringdata_ExtraSettingsWidget_t, stringdata0) +  0    -  0
        * sizeof(QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 18 ,   qptrdiff(offsetof(
      qt_meta_stringdata_ExtraSettingsWidget_t, stringdata0) +  20    -
      1  * sizeof(QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 0 ,   qptrdiff(offsetof(
      qt_meta_stringdata_ExtraSettingsWidget_t, stringdata0) +  39    -
      2  * sizeof(QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 7 ,   qptrdiff(offsetof(
      qt_meta_stringdata_ExtraSettingsWidget_t, stringdata0) +  40    -
      3  * sizeof(QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 24 ,   qptrdiff(offsetof(
      qt_meta_stringdata_ExtraSettingsWidget_t, stringdata0) +  48    -
      4  * sizeof(QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 25 ,   qptrdiff(offsetof(
      qt_meta_stringdata_ExtraSettingsWidget_t, stringdata0) +  73    -
      5  * sizeof(QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 22 ,   qptrdiff(offsetof(
      qt_meta_stringdata_ExtraSettingsWidget_t, stringdata0) +  99    -
      6  * sizeof(QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 4 ,   qptrdiff(offsetof(
      qt_meta_stringdata_ExtraSettingsWidget_t, stringdata0) +  122    -
       7  * sizeof(QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 20 ,   qptrdiff(offsetof(
      qt_meta_stringdata_ExtraSettingsWidget_t, stringdata0) +  127    -
       8  * sizeof(QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 20 ,   qptrdiff(offsetof(
      qt_meta_stringdata_ExtraSettingsWidget_t, stringdata0) +  148    -
       9  * sizeof(QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 20 ,   qptrdiff(offsetof(
      qt_meta_stringdata_ExtraSettingsWidget_t, stringdata0) +  169    -
       10  * sizeof(QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 25 ,   qptrdiff(offsetof(
      qt_meta_stringdata_ExtraSettingsWidget_t, stringdata0) +  190    -
       11  * sizeof(QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 19 ,   qptrdiff(offsetof(
      qt_meta_stringdata_ExtraSettingsWidget_t, stringdata0) +  216    -
       12  * sizeof(QByteArrayData))   )

    },
    "ExtraSettingsWidget\0on_trayBox_clicked\0"
    "\0checked\0on_brightnessBox_clicked\0"
    "on_animScanButton_clicked\0"
    "on_fpsBox_valueChanged\0arg1\0"
    "on_ditherBox_clicked\0on_mAccelBox_clicked\0"
    "on_sAccelBox_clicked\0on_sSpeedBox_valueChanged\0"
    "on_delayBox_clicked"
}
```

Definition at line 30 of file moc_extrasettingswidget.cpp.

13.119 src/ckb/release/moc_fwupgradedialog.cpp File Reference

```
#include "../fwupgradedialog.h"
#include <QtCore/qbytearray.h>
#include <QtCore/qmetatype.h>
```

Include dependency graph for moc_fwupgradedialog.cpp:



Data Structures

- struct qt_meta_stringdata_FwUpgradeDialog_t

Macros

- #define QT_MOC_LITERAL(idx, ofs, len)

Variables

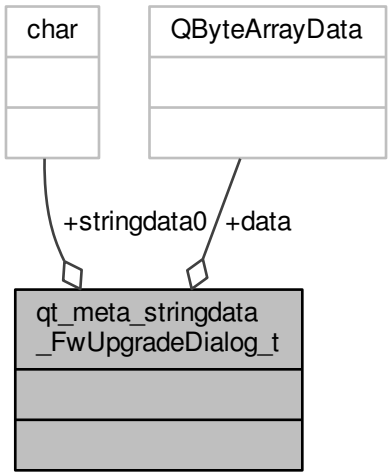
- static const qt_meta_stringdata_FwUpgradeDialog_t qt_meta_stringdata_FwUpgradeDialog
- static const uint qt_meta_data_FwUpgradeDialog [ ]

13.119.1 Data Structure Documentation

13.119.1.1 struct qt_meta_stringdata_FwUpgradeDialog_t

Definition at line 21 of file moc_fwupgradedialog.cpp.

Collaboration diagram for qt_meta_stringdata_FwUpgradeDialog_t:



Data Fields

QByteArrayData	data[10]	
char	stringdata0[133]	

13.119.2 Macro Definition Documentation

13.119.2.1 #define QT_MOC_LITERAL(*idx, ofs, len*)

Value:

```
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET(len, \
    qptrdiff(offsetof(qt_meta_stringdata_FwUpgradeDialog_t, stringdata0
      ) + ofs \
        - idx * sizeof(QByteArrayData)) \
    )
```

Definition at line 25 of file moc_fwupgradedialog.cpp.

13.119.3 Variable Documentation

13.119.3.1 const uint qt_meta_data_FwUpgradeDialog[ ] [static]

Initial value:

```
= {

       7,
       0,
       0,    0,
       5,   14,
       0,    0,
       0,    0,
       0,    0,
```

```
       0,
       0,


       1,    2,   39,    2, 0x08 ,
       5,    1,   44,    2, 0x08 ,
       7,    0,   47,    2, 0x08 ,
       8,    0,   48,    2, 0x08 ,
       9,    0,   49,    2, 0x08 ,


    QMetaType::Void, QMetaType::Int, QMetaType::Int,    3,    4,
    QMetaType::Void, QMetaType::Bool,    6,
    QMetaType::Void,
    QMetaType::Void,
    QMetaType::Void,

       0
}
```

Definition at line 51 of file moc_fwupgradedialog.cpp.

13.119.3.2 const qt_meta_stringdata_FwUpgradeDialog_t qt_meta_stringdata_FwUpgradeDialog [static]

Initial value:

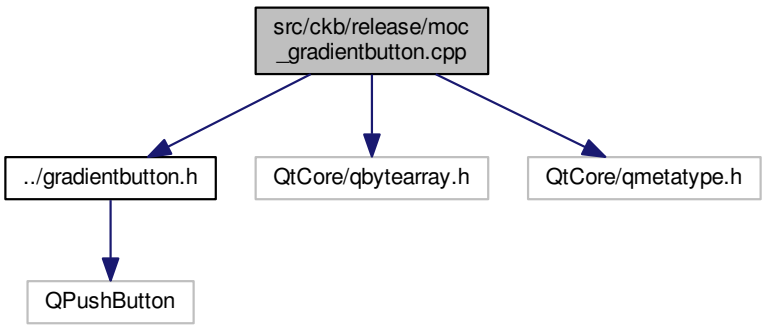
```
= {
    {
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 15 ,   qptrdiff(offsetof(
      qt_meta_stringdata_FwUpgradeDialog_t, stringdata0) +  0    -  0  *
      sizeof(QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 16 ,   qptrdiff(offsetof(
      qt_meta_stringdata_FwUpgradeDialog_t, stringdata0) +  16    -  1  *
      sizeof(QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 0 ,   qptrdiff(offsetof(
      qt_meta_stringdata_FwUpgradeDialog_t, stringdata0) +  33    -  2  *
      sizeof(QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 7 ,   qptrdiff(offsetof(
      qt_meta_stringdata_FwUpgradeDialog_t, stringdata0) +  34    -  3  *
      sizeof(QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 5 ,   qptrdiff(offsetof(
      qt_meta_stringdata_FwUpgradeDialog_t, stringdata0) +  42    -  4  *
      sizeof(QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 16 ,   qptrdiff(offsetof(
      qt_meta_stringdata_FwUpgradeDialog_t, stringdata0) +  48    -  5  *
      sizeof(QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 9 ,   qptrdiff(offsetof(
      qt_meta_stringdata_FwUpgradeDialog_t, stringdata0) +  65    -  6  *
      sizeof(QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 9 ,   qptrdiff(offsetof(
      qt_meta_stringdata_FwUpgradeDialog_t, stringdata0) +  75    -  7  *
      sizeof(QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 23 ,   qptrdiff(offsetof(
      qt_meta_stringdata_FwUpgradeDialog_t, stringdata0) +  85    -  8  *
      sizeof(QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 23 ,   qptrdiff(offsetof(
      qt_meta_stringdata_FwUpgradeDialog_t, stringdata0) +  109    -  9  *
      sizeof(QByteArrayData))   )

    },
    "FwUpgradeDialog\0fwUpdateProgress\0\0"
    "current\0total\0fwUpdateFinished\0succeeded\0"
    "removeDev\0on_cancelButton_clicked\0"
    "on_actionButton_clicked"
}
```

Definition at line 30 of file moc_fwupgradedialog.cpp.

13.120 src/ckb/release/moc_gradientbutton.cpp File Reference

```
#include "../gradientbutton.h"
#include <QtCore/qbytearray.h>
#include <QtCore/qmetatype.h>
```

Include dependency graph for moc_gradientbutton.cpp:



Data Structures

- struct qt_meta_stringdata_GradientButton_t

Macros

- #define QT_MOC_LITERAL(idx, ofs, len)

Variables

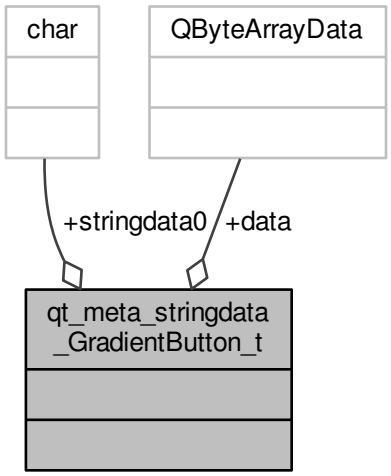
- static const qt_meta_stringdata_GradientButton_t qt_meta_stringdata_GradientButton
- static const uint qt_meta_data_GradientButton []

13.120.1 Data Structure Documentation

13.120.1.1 struct qt_meta_stringdata_GradientButton_t

Definition at line 21 of file moc_gradientbutton.cpp.

Collaboration diagram for qt_meta_stringdata_GradientButton_t:



Data Fields

QByteArrayData	data[4]	
char	stringdata0[45]	

13.120.2 Macro Definition Documentation

13.120.2.1 #define QT_MOC_LITERAL(*idx, ofs, len*)

Value:

```
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET(len, \
    qptrdiff(offsetof(qt_meta_stringdata_GradientButton_t, stringdata0)
      + ofs \
        - idx * sizeof(QByteArrayData)) \
    )
```

Definition at line 25 of file moc_gradientbutton.cpp.

13.120.3 Variable Documentation

13.120.3.1 const uint qt_meta_data_GradientButton[] [static]

Initial value:

```
= {

       7,
       0,
       0,    0,
       2,   14,
       0,    0,
       0,    0,
       0,    0,
```

```
        0,
        1,

        1,    0,   24,    2, 0x06 ,

        3,    0,   25,    2, 0x08 ,

    QMetaType::Void,

    QMetaType::Void,

        0
}
```

Definition at line 43 of file moc_gradientbutton.cpp.

13.120.3.2 const qt_meta_stringdata_GradientButton_t qt_meta_stringdata_GradientButton [static]

Initial value:

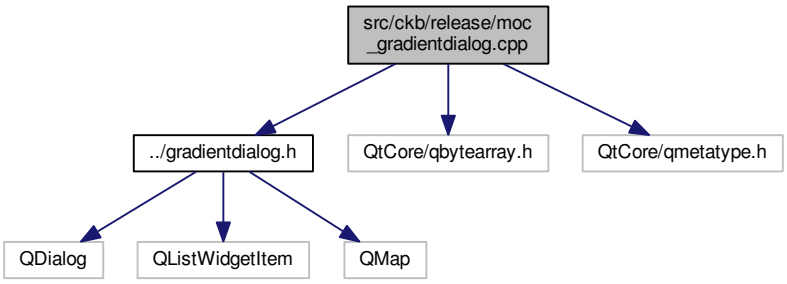
```
= {
    {
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 14 ,   qptrdiff(offsetof(
      qt_meta_stringdata_GradientButton_t, stringdata0) +  0    -  0  * sizeof
      (QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 15 ,   qptrdiff(offsetof(
      qt_meta_stringdata_GradientButton_t, stringdata0) +  15    -  1  *
      sizeof(QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 0 ,   qptrdiff(offsetof(
      qt_meta_stringdata_GradientButton_t, stringdata0) +  31    -  2  *
      sizeof(QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 12 ,   qptrdiff(offsetof(
      qt_meta_stringdata_GradientButton_t, stringdata0) +  32    -  3  *
      sizeof(QByteArrayData))   )

    },
    "GradientButton\0gradientChanged\0\0"
    "pickGradient"
}
```

Definition at line 30 of file moc_gradientbutton.cpp.

13.121 src/ckb/release/moc_gradientdialog.cpp File Reference

```
#include "../gradientdialog.h"
#include <QtCore/qbytearray.h>
#include <QtCore/qmetatype.h>
```

Include dependency graph for moc_gradientdialog.cpp:



Data Structures

- struct qt_meta_stringdata_GradientDialog_t

Macros

- #define QT_MOC_LITERAL(idx, ofs, len)

Variables

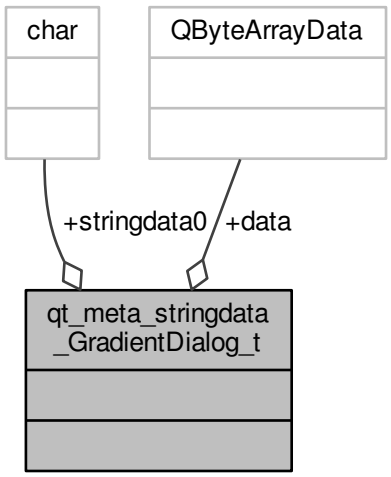
- static const qt_meta_stringdata_GradientDialog_t qt_meta_stringdata_GradientDialog
- static const uint qt_meta_data_GradientDialog [ ]

13.121.1 Data Structure Documentation

13.121.1.1 struct qt_meta_stringdata_GradientDialog_t

Definition at line 21 of file moc_gradientdialog.cpp.

Collaboration diagram for qt_meta_stringdata_GradientDialog_t:



Data Fields

QByteArrayData	data[17]	
char	stringdata0[266]	

13.121.2 Macro Definition Documentation

13.121.2.1 #define QT_MOC_LITERAL(*idx, ofs, len*)

Value:

```
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET(len, \
    qptrdiff(offsetof(qt_meta_stringdata_GradientDialog_t, stringdata0)
      + ofs \
        - idx * sizeof(QByteArrayData)) \
    )
```

Definition at line 25 of file moc_gradientdialog.cpp.

13.121.3 Variable Documentation

13.121.3.1 const uint qt_meta_data_GradientDialog[] [static]

Definition at line 62 of file moc_gradientdialog.cpp.

13.121.3.2 const qt_meta_stringdata_GradientDialog_t qt_meta_stringdata_GradientDialog [static]

Initial value:

```
= {
    {
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 14 ,   qptrdiff(offsetof(
      qt_meta_stringdata_GradientDialog_t, stringdata0) +  0    -  0  * sizeof
      (QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 14 ,   qptrdiff(offsetof(
      qt_meta_stringdata_GradientDialog_t, stringdata0) +  15    -  1  *
      sizeof(QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 0 ,   qptrdiff(offsetof(
      qt_meta_stringdata_GradientDialog_t, stringdata0) +  30    -  2  *
      sizeof(QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 5 ,   qptrdiff(offsetof(
      qt_meta_stringdata_GradientDialog_t, stringdata0) +  31    -  3  *
      sizeof(QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 11 ,   qptrdiff(offsetof(
      qt_meta_stringdata_GradientDialog_t, stringdata0) +  37    -  4  *
      sizeof(QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 8 ,   qptrdiff(offsetof(
      qt_meta_stringdata_GradientDialog_t, stringdata0) +  49    -  5  *
      sizeof(QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 12 ,   qptrdiff(offsetof(
      qt_meta_stringdata_GradientDialog_t, stringdata0) +  58    -  6  *
      sizeof(QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 32 ,   qptrdiff(offsetof(
      qt_meta_stringdata_GradientDialog_t, stringdata0) +  71    -  7  *
      sizeof(QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 16 ,   qptrdiff(offsetof(
      qt_meta_stringdata_GradientDialog_t, stringdata0) +  104    -  8  *
      sizeof(QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 7 ,   qptrdiff(offsetof(
      qt_meta_stringdata_GradientDialog_t, stringdata0) +  121    -  9  *
      sizeof(QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 8 ,   qptrdiff(offsetof(
      qt_meta_stringdata_GradientDialog_t, stringdata0) +  129    -  10  *
      sizeof(QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 24 ,   qptrdiff(offsetof(
      qt_meta_stringdata_GradientDialog_t, stringdata0) +  138    -  11  *
      sizeof(QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 4 ,   qptrdiff(offsetof(
      qt_meta_stringdata_GradientDialog_t, stringdata0) +  163    -  12  *
      sizeof(QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 21 ,   qptrdiff(offsetof(
      qt_meta_stringdata_GradientDialog_t, stringdata0) +  168    -  13  *
      sizeof(QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 23 ,   qptrdiff(offsetof(
      qt_meta_stringdata_GradientDialog_t, stringdata0) +  190    -  14  *
      sizeof(QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 23 ,   qptrdiff(offsetof(
      qt_meta_stringdata_GradientDialog_t, stringdata0) +  214    -  15  *
      sizeof(QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 27 ,   qptrdiff(offsetof(
      qt_meta_stringdata_GradientDialog_t, stringdata0) +  238    -  16  *
      sizeof(QByteArrayData))   )

    },
    "GradientDialog\0currentChanged\0\0color\0"
    "spontaneous\0position\0colorChanged\0"
    "on_presetList_currentItemChanged\0"
    "QListWidgetItem*\0current\0previous\0"
```

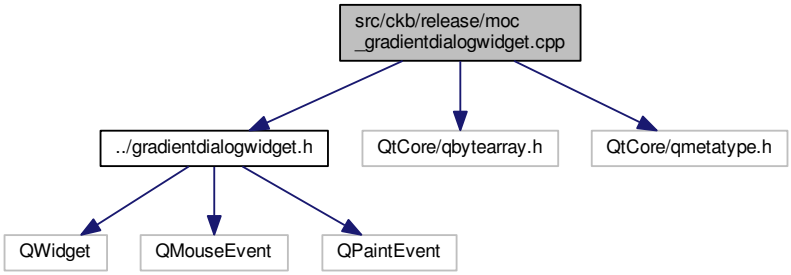
```
    "on_presetName_textEdited\0arg1\0"
    "on_presetSave_clicked\0on_presetDelete_clicked\0"
    "on_stopPos_valueChanged\0"
    "on_stopOpacity_valueChanged"
}
```

Definition at line 30 of file moc_gradientdialog.cpp.

13.122 src/ckb/release/moc_gradientdialogwidget.cpp File Reference

```
#include "../gradientdialogwidget.h"
#include <QtCore/qbytearray.h>
#include <QtCore/qmetatype.h>
```

Include dependency graph for moc_gradientdialogwidget.cpp:



Data Structures

- struct qt_meta_stringdata_GradientDialogWidget_t

Macros

- #define QT_MOC_LITERAL(idx, ofs, len)

Variables

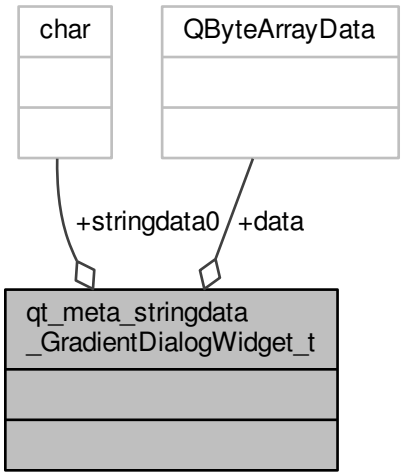
- static const qt_meta_stringdata_GradientDialogWidget_t qt_meta_stringdata_GradientDialogWidget
- static const uint qt_meta_data_GradientDialogWidget [ ]

13.122.1 Data Structure Documentation

13.122.1.1 struct qt_meta_stringdata_GradientDialogWidget_t

Definition at line 21 of file moc_gradientdialogwidget.cpp.

Collaboration diagram for qt_meta_stringdata_GradientDialogWidget_t:



Data Fields

QByteArrayData	data[6]	
char	stringdata0[64]	

13.122.2 Macro Definition Documentation

13.122.2.1 #define QT_MOC_LITERAL(*idx, ofs, len*)

Value:

```
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET(len, \
    qptrdiff(offsetof(qt_meta_stringdata_GradientDialogWidget_t,
      stringdata0) + ofs \
        - idx * sizeof(QByteArrayData)) \
    )
```

Definition at line 25 of file moc_gradientdialogwidget.cpp.

13.122.3 Variable Documentation

13.122.3.1 const uint qt_meta_data_GradientDialogWidget[] [static]

Initial value:

```
= {

       7,
       0,
       0,    0,
       1,   14,
       0,    0,
       0,    0,
       0,    0,
```

```
       0,
       1,

       1,    3,   19,    2, 0x06 ,

    QMetaType::Void, QMetaType::QColor, QMetaType::Bool, QMetaType::Int,    3,    4,    5,

       0
}
```

Definition at line 45 of file moc_gradientdialogwidget.cpp.

13.122.3.2 const qt_meta_stringdata_GradientDialogWidget_t qt_meta_stringdata_GradientDialogWidget [static]

Initial value:

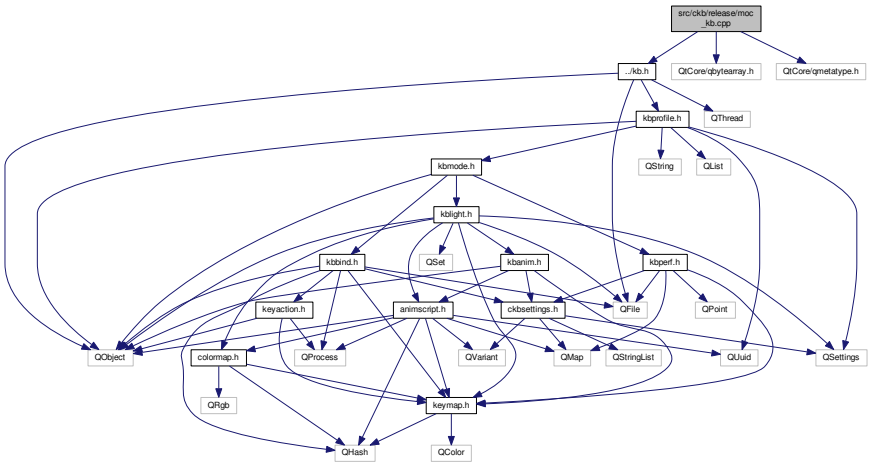
```
= {
    {
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 20 ,   qptrdiff(offsetof(
      qt_meta_stringdata_GradientDialogWidget_t, stringdata0) +  0    -
       0  * sizeof(QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 14 ,   qptrdiff(offsetof(
      qt_meta_stringdata_GradientDialogWidget_t, stringdata0) +  21    -
        1  * sizeof(QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 0 ,   qptrdiff(offsetof(
      qt_meta_stringdata_GradientDialogWidget_t, stringdata0) +  36    -
        2  * sizeof(QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 5 ,   qptrdiff(offsetof(
      qt_meta_stringdata_GradientDialogWidget_t, stringdata0) +  37    -
        3  * sizeof(QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 11 ,   qptrdiff(offsetof(
      qt_meta_stringdata_GradientDialogWidget_t, stringdata0) +  43    -
        4  * sizeof(QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 8 ,   qptrdiff(offsetof(
      qt_meta_stringdata_GradientDialogWidget_t, stringdata0) +  55    -
        5  * sizeof(QByteArrayData))   )

    },
    "GradientDialogWidget\0currentChanged\0"
    "\0color\0spontaneous\0position"
}
```

Definition at line 30 of file moc_gradientdialogwidget.cpp.

13.123 src/ckb/release/moc_kb.cpp File Reference

```
#include "../kb.h"
#include <QtCore/qbytearray.h>
#include <QtCore/qmetatype.h>
```

Include dependency graph for moc_kb.cpp:



Data Structures

- struct qt_meta_stringdata_Kb_t

Macros

- #define QT_MOC_LITERAL(idx, ofs, len)

Variables

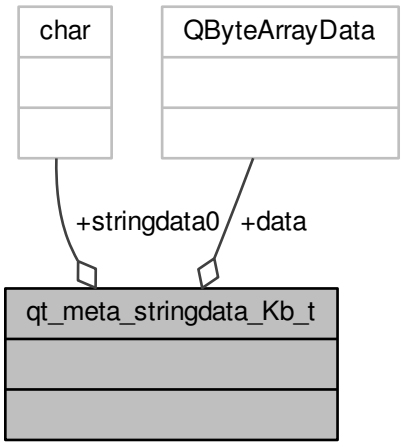
- static const qt_meta_stringdata_Kb_t qt_meta_stringdata_Kb
- static const uint qt_meta_data_Kb [ ]

13.123.1 Data Structure Documentation

13.123.1.1 struct qt_meta_stringdata_Kb_t

Definition at line 21 of file moc_kb.cpp.

Collaboration diagram for qt_meta_stringdata_Kb_t:



Data Fields

QByteArrayData	data[20]	
char	stringdata0[214]	

13.123.2 Macro Definition Documentation

13.123.2.1 #define QT_MOC_LITERAL(*idx, ofs, len*)

Value:

```
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET(len, \
    qptrdiff(offsetof(qt_meta_stringdata_Kb_t, stringdata0) + ofs \
        - idx * sizeof(QByteArrayData)) \
    )
```

Definition at line 25 of file moc_kb.cpp.

13.123.3 Variable Documentation

13.123.3.1 const uint qt_meta_data_Kb[] [static]

Definition at line 63 of file moc_kb.cpp.

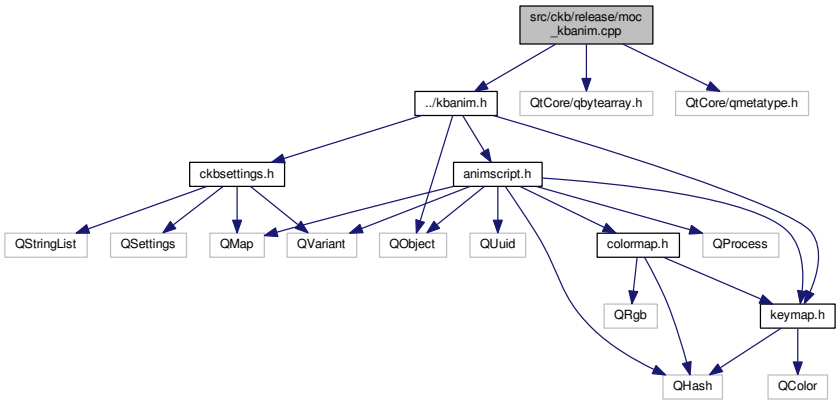
13.123.3.2 const qt_meta_stringdata_Kb_t qt_meta_stringdata_Kb [static]

Definition at line 30 of file moc_kb.cpp.

13.124 src/ckb/release/moc_kbanim.cpp File Reference

```
#include "../kbanim.h"
#include <QtCore/qbytearray.h>
#include <QtCore/qmetatype.h>
```

Include dependency graph for moc_kbanim.cpp:



Data Structures

- struct qt_meta_stringdata_KbAnim_t

Macros

- #define QT_MOC_LITERAL(idx, ofs, len)

Variables

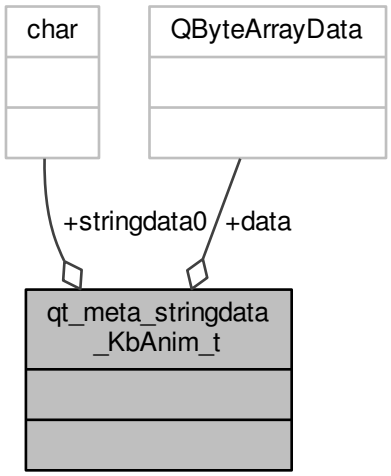
- static const qt_meta_stringdata_KbAnim_t qt_meta_stringdata_KbAnim
- static const uint qt_meta_data_KbAnim [ ]

13.124.1 Data Structure Documentation

13.124.1.1 struct qt_meta_stringdata_KbAnim_t

Definition at line 21 of file moc_kbanim.cpp.

Collaboration diagram for qt_meta_stringdata_KbAnim_t:



Data Fields

QByteArrayData	data[7]	
char	stringdata0[48]	

13.124.2 Macro Definition Documentation

13.124.2.1 #define QT_MOC_LITERAL(*idx, ofs, len*)

Value:

```
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET(len, \
    qptrdiff(offsetof(qt_meta_stringdata_KbAnim_t, stringdata0) + ofs \
        - idx * sizeof(QByteArrayData)) \
    )
```

Definition at line 25 of file moc_kbanim.cpp.

13.124.3 Variable Documentation

13.124.3.1 const uint qt_meta_data_KbAnim[] [static]

Initial value:

```
= {

       7,
       0,
       0,    0,
       0,    0,
       0,    0,
       1,   14,
       0,    0,
       0,
```

```
        0,

        1, 0x0,    5,   18,

        2, uint(KbAnim::Normal),
        3, uint(KbAnim::Add),
        4, uint(KbAnim::Subtract),
        5, uint(KbAnim::Multiply),
        6, uint(KbAnim::Divide),

        0
}
```

Definition at line 46 of file moc_kbanim.cpp.

13.124.3.2 const qt_meta_stringdata_KbAnim_t qt_meta_stringdata_KbAnim [static]

Initial value:

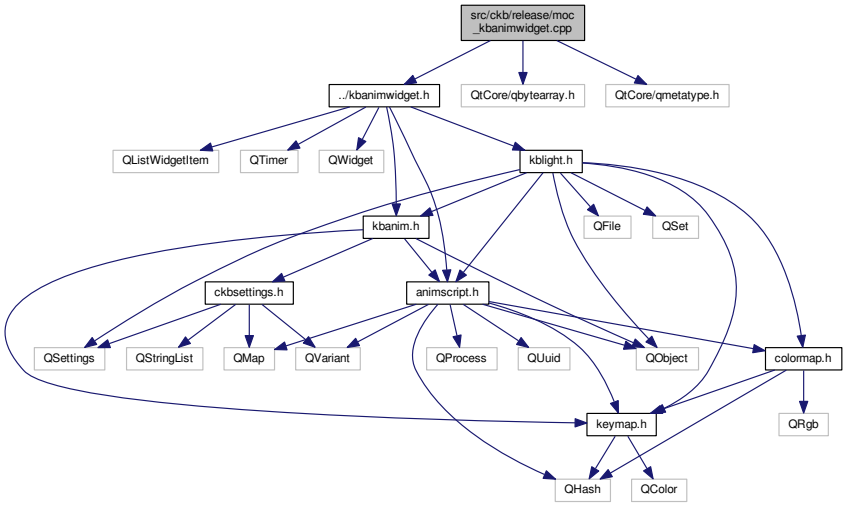
```
= {
    {
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 6 ,   qptrdiff(offsetof(
      qt_meta_stringdata_KbAnim_t, stringdata0) +  0    -  0  * sizeof(QByteArrayData)
      )   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 4 ,   qptrdiff(offsetof(
      qt_meta_stringdata_KbAnim_t, stringdata0) +  7    -  1  * sizeof(QByteArrayData)
      )   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 6 ,   qptrdiff(offsetof(
      qt_meta_stringdata_KbAnim_t, stringdata0) +  12    -  2  * sizeof(QByteArrayData
      ))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 3 ,   qptrdiff(offsetof(
      qt_meta_stringdata_KbAnim_t, stringdata0) +  19    -  3  * sizeof(QByteArrayData
      ))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 8 ,   qptrdiff(offsetof(
      qt_meta_stringdata_KbAnim_t, stringdata0) +  23    -  4  * sizeof(QByteArrayData
      ))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 8 ,   qptrdiff(offsetof(
      qt_meta_stringdata_KbAnim_t, stringdata0) +  32    -  5  * sizeof(QByteArrayData
      ))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 6 ,   qptrdiff(offsetof(
      qt_meta_stringdata_KbAnim_t, stringdata0) +  41    -  6  * sizeof(QByteArrayData
      ))   )

    },
    "KbAnim\0Mode\0Normal\0Add\0Subtract\0"
    "Multiply\0Divide"
}
```

Definition at line 30 of file moc_kbanim.cpp.

13.125 src/ckb/release/moc_kbanimwidget.cpp File Reference

```
#include "../kbanimwidget.h"
#include <QtCore/qbytearray.h>
#include <QtCore/qmetatype.h>
```

Include dependency graph for moc_kbanimwidget.cpp:



Data Structures

- struct qt_meta_stringdata_KbAnimWidget_t

Macros

- #define QT_MOC_LITERAL(idx, ofs, len)

Variables

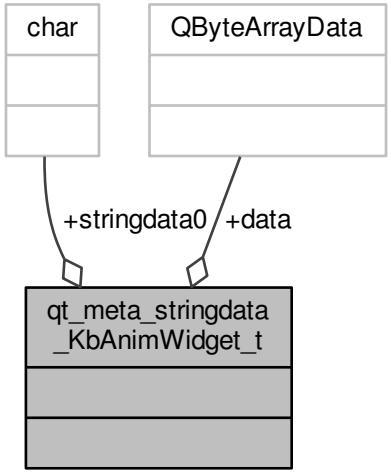
- static const qt_meta_stringdata_KbAnimWidget_t qt_meta_stringdata_KbAnimWidget
- static const uint qt_meta_data_KbAnimWidget [ ]

13.125.1 Data Structure Documentation

13.125.1.1 struct qt_meta_stringdata_KbAnimWidget_t

Definition at line 21 of file moc_kbanimwidget.cpp.

Collaboration diagram for qt_meta_stringdata_KbAnimWidget_t:



Data Fields

QByteArrayData	data[25]	
char	stringdata0[383]	

13.125.2 Macro Definition Documentation

13.125.2.1 #define QT_MOC_LITERAL(*idx, ofs, len*)

Value:

```
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET(len, \
    qptrdiff(offsetof(qt_meta_stringdata_KbAnimWidget_t, stringdata0) +
      ofs \
        - idx * sizeof(QByteArrayData)) \
    )
```

Definition at line 25 of file moc_kbanimwidget.cpp.

13.125.3 Variable Documentation

13.125.3.1 const uint qt_meta_data_KbAnimWidget[] [static]

Definition at line 74 of file moc_kbanimwidget.cpp.

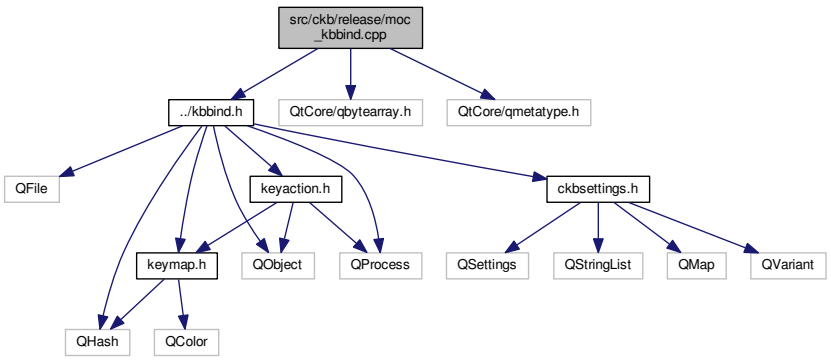
13.125.3.2 const qt_meta_stringdata_KbAnimWidget_t qt_meta_stringdata_KbAnimWidget [static]

Definition at line 30 of file moc_kbanimwidget.cpp.

13.126 src/ckb/release/moc_kbbind.cpp File Reference

```
#include "../kbbind.h"
#include <QtCore/qbytearray.h>
#include <QtCore/qmetatype.h>
```

Include dependency graph for moc_kbbind.cpp:



Data Structures

- struct qt_meta_stringdata_KbBind_t

Macros

- #define QT_MOC_LITERAL(idx, ofs, len)

Variables

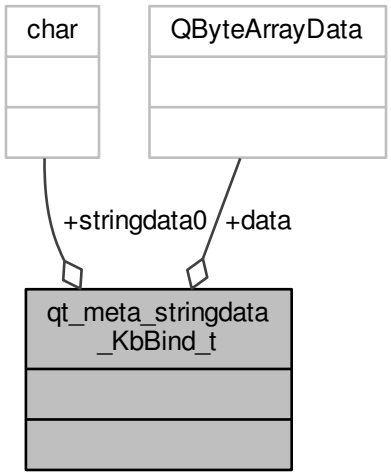
- static const qt_meta_stringdata_KbBind_t qt_meta_stringdata_KbBind
- static const uint qt_meta_data_KbBind []

13.126.1 Data Structure Documentation

13.126.1.1 struct qt_meta_stringdata_KbBind_t

Definition at line 21 of file moc_kbbind.cpp.

Collaboration diagram for qt_meta_stringdata_KbBind_t:



Data Fields

QByteArrayData	data[8]	
char	stringdata0[56]	

13.126.2 Macro Definition Documentation

13.126.2.1 #define QT_MOC_LITERAL(*idx, ofs, len*)

Value:

```
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET(len, \
    qptrdiff(offsetof(qt_meta_stringdata_KbBind_t, stringdata0) + ofs \
        - idx * sizeof(QByteArrayData)) \
    )
```

Definition at line 25 of file moc_kbbind.cpp.

13.126.3 Variable Documentation

13.126.3.1 const uint qt_meta_data_KbBind[] [static]

Definition at line 47 of file moc_kbbind.cpp.

13.126.3.2 const qt_meta_stringdata_KbBind_t qt_meta_stringdata_KbBind [static]

Initial value:

```
= {
    {
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 6 ,   qptrdiff(offsetof(
      qt_meta_stringdata_KbBind_t, stringdata0) +  0    -  0  * sizeof(QByteArrayData)
```

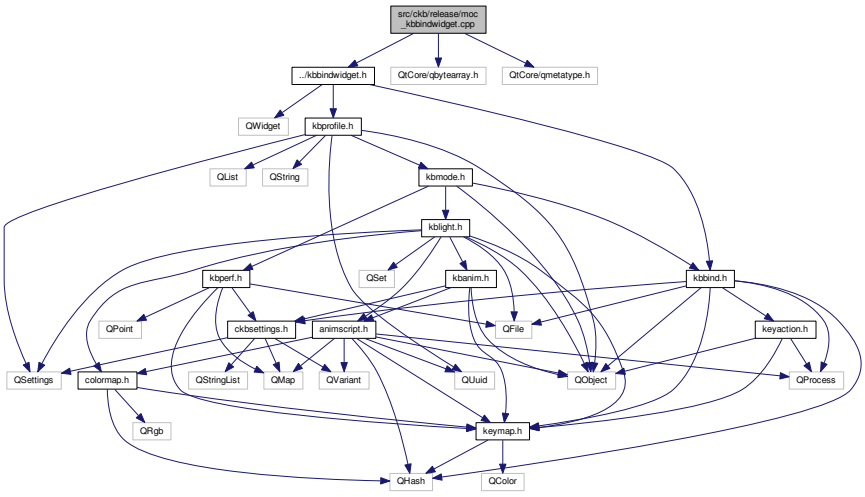
```
      )   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 7 ,   qptrdiff(offsetof(
      qt_meta_stringdata_KbBind_t, stringdata0) +  7    -  1  * sizeof(QByteArrayData)
      )   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 0 ,   qptrdiff(offsetof(
      qt_meta_stringdata_KbBind_t, stringdata0) +  15    -  2  * sizeof(QByteArrayData
      ))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 13 ,   qptrdiff(offsetof(
      qt_meta_stringdata_KbBind_t, stringdata0) +  16    -  3  * sizeof(QByteArrayData
      ))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 7 ,   qptrdiff(offsetof(
      qt_meta_stringdata_KbBind_t, stringdata0) +  30    -  4  * sizeof(QByteArrayData
      ))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 8 ,   qptrdiff(offsetof(
      qt_meta_stringdata_KbBind_t, stringdata0) +  38    -  5  * sizeof(QByteArrayData
      ))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 3 ,   qptrdiff(offsetof(
      qt_meta_stringdata_KbBind_t, stringdata0) +  47    -  6  * sizeof(QByteArrayData
      ))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 4 ,   qptrdiff(offsetof(
      qt_meta_stringdata_KbBind_t, stringdata0) +  51    -  7  * sizeof(QByteArrayData
      ))   )

    },
    "KbBind\0didLoad\0\0layoutChanged\0updated\0"
    "keyEvent\0key\0down"
}
```

Definition at line 30 of file moc_kbbind.cpp.

13.127 src/ckb/release/moc_kbbindwidget.cpp File Reference

```
#include "../kbbindwidget.h"
#include <QtCore/qbytearray.h>
#include <QtCore/qmetatype.h>
```

Include dependency graph for moc_kbbindwidget.cpp:



Data Structures

- struct qt_meta_stringdata_KbBindWidget_t

Macros

- #define QT_MOC_LITERAL(idx, ofs, len)

Variables

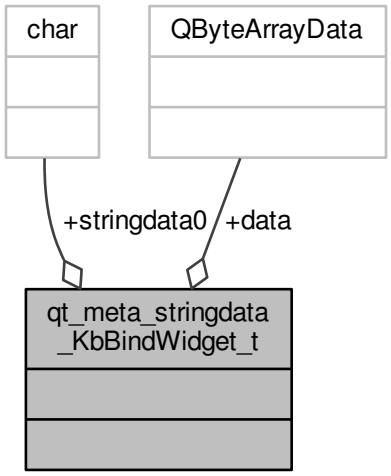
- static const qt_meta_stringdata_KbBindWidget_t qt_meta_stringdata_KbBindWidget
- static const uint qt_meta_data_KbBindWidget []

13.127.1 Data Structure Documentation

13.127.1.1 struct qt_meta_stringdata_KbBindWidget_t

Definition at line 21 of file moc_kbbindwidget.cpp.

Collaboration diagram for qt_meta_stringdata_KbBindWidget_t:



Data Fields

QByteArrayData	data[9]	
char	stringdata0[120]	

13.127.2 Macro Definition Documentation

13.127.2.1 #define QT_MOC_LITERAL(*idx, ofs, len*)

Value:

```
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET(len, \
    qptrdiff(offsetof(qt_meta_stringdata_KbBindWidget_t, stringdata0) +
      ofs \
        - idx * sizeof(QByteArrayData)) \
    )
```

Definition at line 25 of file moc_kbbindwidget.cpp.

13.127.3 Variable Documentation

13.127.3.1 const uint qt_meta_data_KbBindWidget[] [static]

Definition at line 49 of file moc_kbbindwidget.cpp.

13.127.3.2 const qt_meta_stringdata_KbBindWidget_t qt_meta_stringdata_KbBindWidget [static]

Initial value:

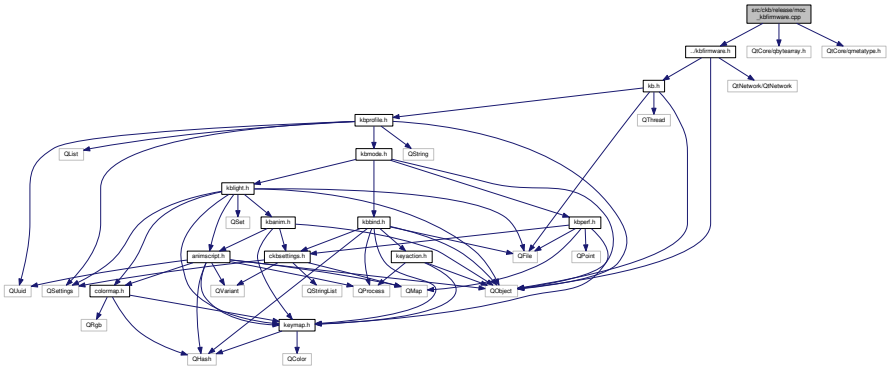
```
= {
    {
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 12 ,   qptrdiff(offsetof(
      qt_meta_stringdata_KbBindWidget_t, stringdata0) +  0    -  0  * sizeof(
      QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 10 ,   qptrdiff(offsetof(
      qt_meta_stringdata_KbBindWidget_t, stringdata0) +  13    -  1  * sizeof(
      QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 0 ,   qptrdiff(offsetof(
      qt_meta_stringdata_KbBindWidget_t, stringdata0) +  24    -  2  * sizeof(
      QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 9 ,   qptrdiff(offsetof(
      qt_meta_stringdata_KbBindWidget_t, stringdata0) +  25    -  3  * sizeof(
      QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 12 ,   qptrdiff(offsetof(
      qt_meta_stringdata_KbBindWidget_t, stringdata0) +  35    -  4  * sizeof(
      QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 9 ,   qptrdiff(offsetof(
      qt_meta_stringdata_KbBindWidget_t, stringdata0) +  48    -  5  * sizeof(
      QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 16 ,   qptrdiff(offsetof(
      qt_meta_stringdata_KbBindWidget_t, stringdata0) +  58    -  6  * sizeof(
      QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 22 ,   qptrdiff(offsetof(
      qt_meta_stringdata_KbBindWidget_t, stringdata0) +  75    -  7  * sizeof(
      QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 21 ,   qptrdiff(offsetof(
      qt_meta_stringdata_KbBindWidget_t, stringdata0) +  98    -  8  * sizeof(
      QByteArrayData))   )

    },
    "KbBindWidget\0updateBind\0\0newLayout\0"
    "newSelection\0selection\0updateSelDisplay\0"
    "on_resetButton_clicked\0on_copyButton_clicked"
}
```

Definition at line 30 of file moc_kbbindwidget.cpp.

13.128 src/ckb/release/moc_kbfirmware.cpp File Reference

```
#include "../kbfirmware.h"
#include <QtCore/qbytearray.h>
#include <QtCore/qmetatype.h>
```

Include dependency graph for moc_kbfirmware.cpp:



Data Structures

- struct qt_meta_stringdata_KbFirmware_t

Macros

- #define QT_MOC_LITERAL(idx, ofs, len)

Variables

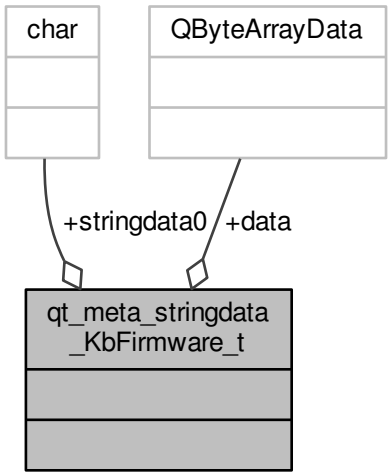
- static const qt_meta_stringdata_KbFirmware_t qt_meta_stringdata_KbFirmware
- static const uint qt_meta_data_KbFirmware [ ]

13.128.1 Data Structure Documentation

13.128.1.1 struct qt_meta_stringdata_KbFirmware_t

Definition at line 21 of file moc_kbfirmware.cpp.

Collaboration diagram for qt_meta_stringdata_KbFirmware_t:



Data Fields

QByteArrayData	data[7]	
char	stringdata0[77]	

13.128.2 Macro Definition Documentation

13.128.2.1 #define QT_MOC_LITERAL(*idx, ofs, len*)

Value:

```
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET(len, \
    qptrdiff(offsetof(qt_meta_stringdata_KbFirmware_t, stringdata0) + ofs \
        - idx * sizeof(QByteArrayData)) \
    )
```

Definition at line 25 of file moc_kbfirmware.cpp.

13.128.3 Variable Documentation

13.128.3.1 const uint qt_meta_data_KbFirmware[] [static]

Initial value:

```
= {

       7,
       0,
       0,    0,
       3,   14,
       0,    0,
       0,    0,
       0,    0,
       0,
```

```
       1,

       1,    0,   29,    2, 0x06 ,

       3,    1,   30,    2, 0x08 ,
       6,    0,   33,    2, 0x08 ,

    QMetaType::Void,

    QMetaType::Void, 0x80000000 | 4,    5,
    QMetaType::Void,

       0
}
```

Definition at line 46 of file moc_kbfirmware.cpp.

13.128.3.2 const qt_meta_stringdata_KbFirmware_t qt_meta_stringdata_KbFirmware [static]

Initial value:

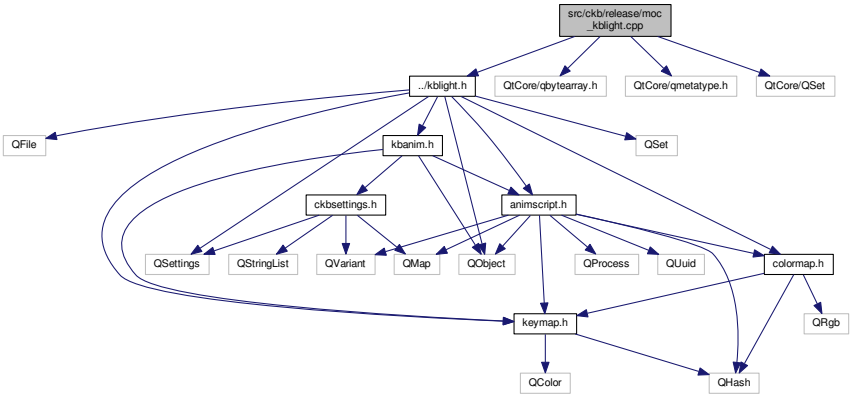
```
= {
    {
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 10 ,   qptrdiff(offsetof(
      qt_meta_stringdata_KbFirmware_t, stringdata0) +  0    -  0  * sizeof(
      QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 10 ,   qptrdiff(offsetof(
      qt_meta_stringdata_KbFirmware_t, stringdata0) +  11    -  1  * sizeof(
      QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 0 ,   qptrdiff(offsetof(
      qt_meta_stringdata_KbFirmware_t, stringdata0) +  22    -  2  * sizeof(
      QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 15 ,   qptrdiff(offsetof(
      qt_meta_stringdata_KbFirmware_t, stringdata0) +  23    -  3  * sizeof(
      QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 14 ,   qptrdiff(offsetof(
      qt_meta_stringdata_KbFirmware_t, stringdata0) +  39    -  4  * sizeof(
      QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 5 ,   qptrdiff(offsetof(
      qt_meta_stringdata_KbFirmware_t, stringdata0) +  54    -  5  * sizeof(
      QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 16 ,   qptrdiff(offsetof(
      qt_meta_stringdata_KbFirmware_t, stringdata0) +  60    -  6  * sizeof(
      QByteArrayData))   )

    },
    "KbFirmware\0downloaded\0\0processDownload\0"
    "QNetworkReply*\0reply\0downloadFinished"
}
```

Definition at line 30 of file moc_kbfirmware.cpp.

13.129 src/ckb/release/moc_kblight.cpp File Reference

```
#include "../kblight.h"
#include <QtCore/qbytearray.h>
#include <QtCore/qmetatype.h>
#include <QtCore/QSet>
```

Include dependency graph for moc_kblight.cpp:



Data Structures

- struct qt_meta_stringdata_KbLight_t

Macros

- #define QT_MOC_LITERAL(idx, ofs, len)

Variables

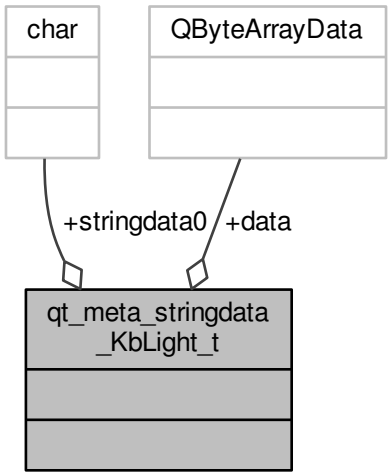
- static const qt_meta_stringdata_KbLight_t qt_meta_stringdata_KbLight
- static const uint qt_meta_data_KbLight [ ]

13.129.1 Data Structure Documentation

13.129.1.1 struct qt_meta_stringdata_KbLight_t

Definition at line 22 of file moc_kblight.cpp.

Collaboration diagram for qt_meta_stringdata_KbLight_t:



Data Fields

QByteArrayData	data[9]	
char	stringdata0[92]	

13.129.2 Macro Definition Documentation

13.129.2.1 #define QT_MOC_LITERAL(*idx, ofs, len*)

Value:

```
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET(len, \
    qptrdiff(offsetof(qt_meta_stringdata_KbLight_t, stringdata0) + ofs \
        - idx * sizeof(QByteArrayData)) \
    )
```

Definition at line 26 of file moc_kblight.cpp.

13.129.3 Variable Documentation

13.129.3.1 const uint qt_meta_data_KbLight[ ] [static]

Initial value:

```
= {

       7,
       0,
       0,    0,
       3,   14,
       0,    0,
       0,    0,
       0,    0,
       0,
```

```
       3,


       1,    0,   29,    2, 0x06 ,
       3,    0,   30,    2, 0x06 ,
       4,    2,   31,    2, 0x06 ,


    QMetaType::Void,
    QMetaType::Void,
    QMetaType::Void, 0x80000000 | 5, 0x80000000 | 7,    6,    8,

       0
}
```

Definition at line 50 of file moc_kblight.cpp.

13.129.3.2 const qt_meta_stringdata_KbLight_t qt_meta_stringdata_KbLight [static]

Initial value:

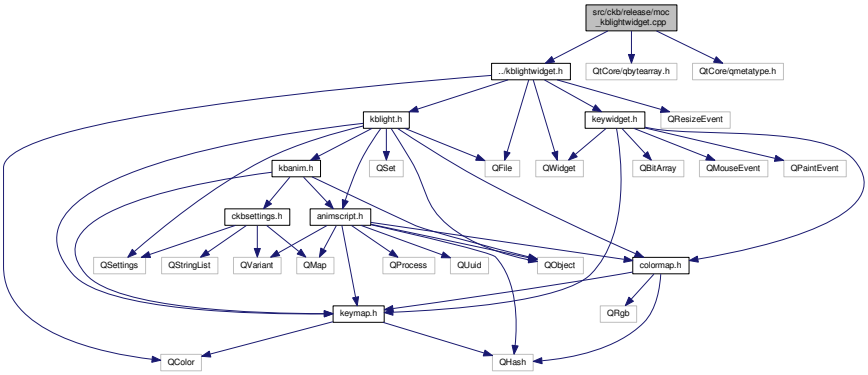
```
= {
    {
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 7 ,   qptrdiff(offsetof(
      qt_meta_stringdata_KbLight_t, stringdata0) +  0    -  0  * sizeof(
      QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 7 ,   qptrdiff(offsetof(
      qt_meta_stringdata_KbLight_t, stringdata0) +  8    -  1  * sizeof(
      QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 0 ,   qptrdiff(offsetof(
      qt_meta_stringdata_KbLight_t, stringdata0) +  16    -  2  * sizeof(
      QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 7 ,   qptrdiff(offsetof(
      qt_meta_stringdata_KbLight_t, stringdata0) +  17    -  3  * sizeof(
      QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 14 ,   qptrdiff(offsetof(
      qt_meta_stringdata_KbLight_t, stringdata0) +  25    -  4  * sizeof(
      QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 8 ,   qptrdiff(offsetof(
      qt_meta_stringdata_KbLight_t, stringdata0) +  40    -  5  * sizeof(
      QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 14 ,   qptrdiff(offsetof(
      qt_meta_stringdata_KbLight_t, stringdata0) +  49    -  6  * sizeof(
      QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 13 ,   qptrdiff(offsetof(
      qt_meta_stringdata_KbLight_t, stringdata0) +  64    -  7  * sizeof(
      QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 13 ,   qptrdiff(offsetof(
      qt_meta_stringdata_KbLight_t, stringdata0) +  78    -  8  * sizeof(
      QByteArrayData))   )

    },
    "KbLight\0didLoad\0\0updated\0frameDisplayed\0"
    "ColorMap\0animatedColors\0QSet<QString>\0"
    "indicatorList"
}
```

Definition at line 31 of file moc_kblight.cpp.

13.130 src/ckb/release/moc_kblightwidget.cpp File Reference

```
#include "../kblightwidget.h"
#include <QtCore/qbytearray.h>
#include <QtCore/qmetatype.h>
```

Include dependency graph for moc_kblightwidget.cpp:



Data Structures

- struct qt_meta_stringdata_KbLightWidget_t

Macros

- #define QT_MOC_LITERAL(idx, ofs, len)

Variables

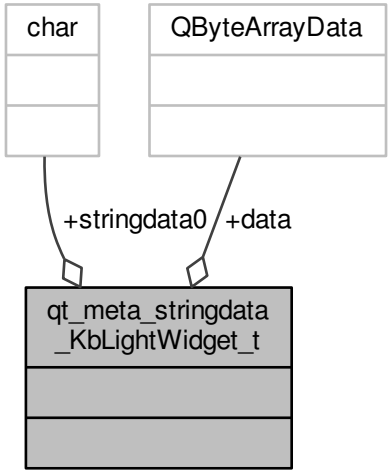
- static const qt_meta_stringdata_KbLightWidget_t qt_meta_stringdata_KbLightWidget
- static const uint qt_meta_data_KbLightWidget [ ]

13.130.1 Data Structure Documentation

13.130.1.1 struct qt_meta_stringdata_KbLightWidget_t

Definition at line 21 of file moc_kblightwidget.cpp.

Collaboration diagram for qt_meta_stringdata_KbLightWidget_t:



Data Fields

QByteArrayData	data[19]	
char	stringdata0[240]	

13.130.2 Macro Definition Documentation

13.130.2.1 #define QT_MOC_LITERAL(*idx, ofs, len*)

Value:

```
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET(len, \
    qptrdiff(offsetof(qt_meta_stringdata_KbLightWidget_t, stringdata0) +
      ofs \
        - idx * sizeof(QByteArrayData)) \
    )
```

Definition at line 25 of file moc_kblightwidget.cpp.

13.130.3 Variable Documentation

13.130.3.1 const uint qt_meta_data_KbLightWidget[] [static]

Definition at line 63 of file moc_kblightwidget.cpp.

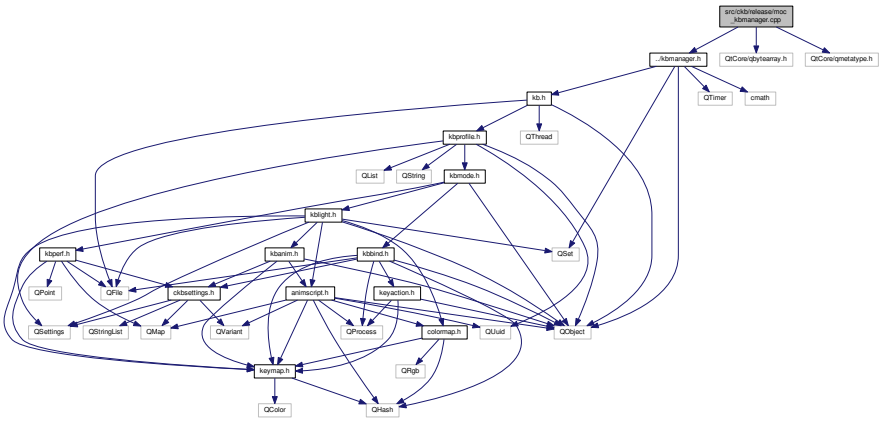
13.130.3.2 const qt_meta_stringdata_KbLightWidget_t qt_meta_stringdata_KbLightWidget [static]

Definition at line 30 of file moc_kblightwidget.cpp.

13.131 src/ckb/release/moc_kbmanager.cpp File Reference

```
#include "../kbmanager.h"
#include <QtCore/qbytearray.h>
#include <QtCore/qmetatype.h>
```

Include dependency graph for moc_kbmanager.cpp:



Data Structures

- struct qt_meta_stringdata_KbManager_t

Macros

- #define QT_MOC_LITERAL(idx, ofs, len)

Variables

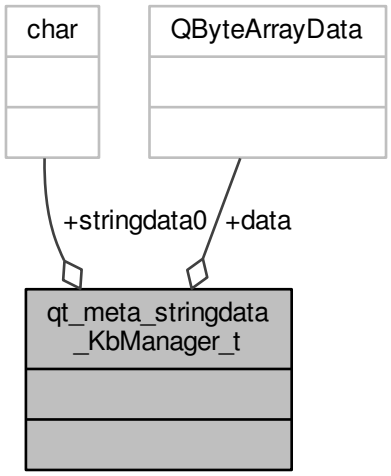
- static const qt_meta_stringdata_KbManager_t qt_meta_stringdata_KbManager
- static const uint qt_meta_data_KbManager [ ]

13.131.1 Data Structure Documentation

13.131.1.1 struct qt_meta_stringdata_KbManager_t

Definition at line 21 of file moc_kbmanager.cpp.

Collaboration diagram for qt_meta_stringdata_KbManager_t:



Data Fields

QByteArrayData	data[8]	
char	stringdata0[78]	

13.131.2 Macro Definition Documentation

13.131.2.1 #define QT_MOC_LITERAL(*idx, ofs, len*)

Value:

```
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET(len, \
    qptrdiff(offsetof(qt_meta_stringdata_KbManager_t, stringdata0) + ofs \
        - idx * sizeof(QByteArrayData)) \
    )
```

Definition at line 25 of file moc_kbmanager.cpp.

13.131.3 Variable Documentation

13.131.3.1 const uint qt_meta_data_KbManager[] [static]

Definition at line 48 of file moc_kbmanager.cpp.

13.131.3.2 const qt_meta_stringdata_KbManager_t qt_meta_stringdata_KbManager [static]

Initial value:

```
= {
    {
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 9 ,   qptrdiff(offsetof(
      qt_meta_stringdata_KbManager_t, stringdata0) +  0    -  0  * sizeof(
```

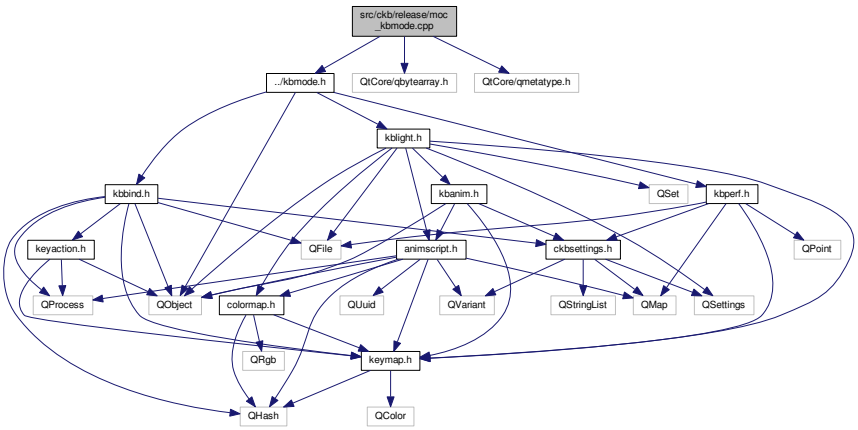
```
      QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 11 ,   qptrdiff(offsetof(
      qt_meta_stringdata_KbManager_t, stringdata0) +  10    -  1  * sizeof(
      QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 0 ,   qptrdiff(offsetof(
      qt_meta_stringdata_KbManager_t, stringdata0) +  22    -  2  * sizeof(
      QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 3 ,   qptrdiff(offsetof(
      qt_meta_stringdata_KbManager_t, stringdata0) +  23    -  3  * sizeof(
      QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 6 ,   qptrdiff(offsetof(
      qt_meta_stringdata_KbManager_t, stringdata0) +  27    -  4  * sizeof(
      QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 14 ,   qptrdiff(offsetof(
      qt_meta_stringdata_KbManager_t, stringdata0) +  34    -  5  * sizeof(
      QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 14 ,   qptrdiff(offsetof(
      qt_meta_stringdata_KbManager_t, stringdata0) +  49    -  6  * sizeof(
      QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 13 ,   qptrdiff(offsetof(
      qt_meta_stringdata_KbManager_t, stringdata0) +  64    -  7  * sizeof(
      QByteArrayData))   )

    },
    "KbManager\0kbConnected\0\0Kb*\0device\0"
    "kbDisconnected\0versionUpdated\0"
    "scanKeyboards"
}
```

Definition at line 30 of file moc_kbmanager.cpp.

13.132 src/ckb/release/moc_kbmode.cpp File Reference

```
#include "../kbmode.h"
#include <QtCore/qbytearray.h>
#include <QtCore/qmetatype.h>
```

Include dependency graph for moc_kbmode.cpp:



Data Structures

- struct qt_meta_stringdata_KbMode_t

Macros

- #define QT_MOC_LITERAL(idx, ofs, len)

Variables

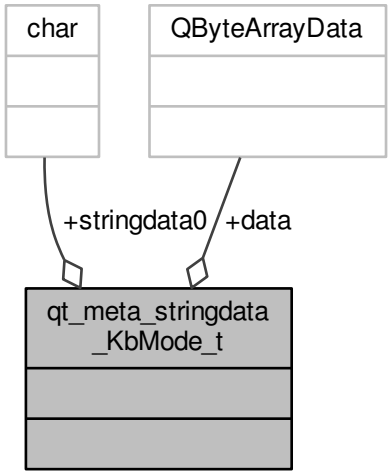
- static const qt_meta_stringdata_KbMode_t qt_meta_stringdata_KbMode
- static const uint qt_meta_data_KbMode [ ]

13.132.1 Data Structure Documentation

13.132.1.1 struct qt_meta_stringdata_KbMode_t

Definition at line 21 of file moc_kbmode.cpp.

Collaboration diagram for qt_meta_stringdata_KbMode_t:



Data Fields

QByteArrayData	data[4]	
char	stringdata0[25]	

13.132.2 Macro Definition Documentation

13.132.2.1 #define QT_MOC_LITERAL(*idx, ofs, len*)

Value:

```
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET(len, \
    qptrdiff(offsetof(qt_meta_stringdata_KbMode_t, stringdata0) + ofs \
        - idx * sizeof(QByteArrayData)) \
    )
```

Definition at line 25 of file moc_kbmode.cpp.

13.132.3 Variable Documentation

13.132.3.1 const uint qt_meta_data_KbMode[] [static]

Initial value:

```
= {

       7,
       0,
       0,    0,
       2,   14,
       0,    0,
       0,    0,
       0,    0,
       0,
       1,


       1,    0,   24,    2, 0x06 ,


       3,    0,   25,    2, 0x08 ,


    QMetaType::Void,


    QMetaType::Void,

       0
}
```

Definition at line 42 of file moc_kbmode.cpp.

13.132.3.2 const qt_meta_stringdata_KbMode_t qt_meta_stringdata_KbMode [static]

Initial value:

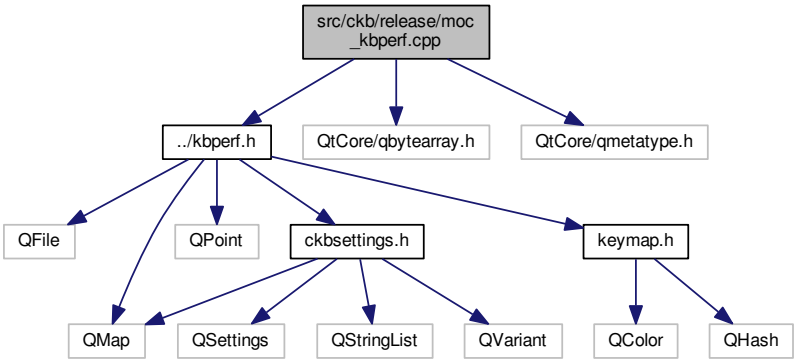
```
= {
    {
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 6 ,   qptrdiff(offsetof(
      qt_meta_stringdata_KbMode_t, stringdata0) +  0    -  0  * sizeof(QByteArrayData)
      )   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 7 ,   qptrdiff(offsetof(
      qt_meta_stringdata_KbMode_t, stringdata0) +  7    -  1  * sizeof(QByteArrayData)
      )   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 0 ,   qptrdiff(offsetof(
      qt_meta_stringdata_KbMode_t, stringdata0) +  15    -  2  * sizeof(QByteArrayData
      ))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 8 ,   qptrdiff(offsetof(
      qt_meta_stringdata_KbMode_t, stringdata0) +  16    -  3  * sizeof(QByteArrayData
      ))   )

    },
    "KbMode\0updated\0\0doUpdate"
}
```

Definition at line 30 of file moc_kbmode.cpp.

13.133 src/ckb/release/moc_kbperf.cpp File Reference

```
#include "../kbperf.h"
#include <QtCore/qbytearray.h>
#include <QtCore/qmetatype.h>
```

Include dependency graph for moc_kbperf.cpp:



Data Structures

- struct qt_meta_stringdata_KbPerf_t

Macros

- #define QT_MOC_LITERAL(idx, ofs, len)

Variables

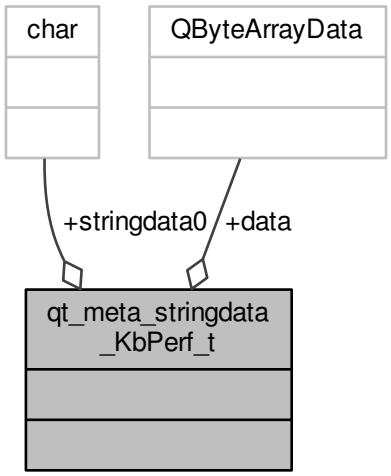
- static const qt_meta_stringdata_KbPerf_t qt_meta_stringdata_KbPerf
- static const uint qt_meta_data_KbPerf [ ]

13.133.1 Data Structure Documentation

13.133.1.1 struct qt_meta_stringdata_KbPerf_t

Definition at line 21 of file moc_kbperf.cpp.

Collaboration diagram for qt_meta_stringdata_KbPerf_t:



Data Fields

QByteArrayData	data[4]	
char	stringdata0[32]	

13.133.2 Macro Definition Documentation

13.133.2.1 #define QT_MOC_LITERAL(*idx, ofs, len*)

Value:

```
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET(len, \
    qptrdiff(offsetof(qt_meta_stringdata_KbPerf_t, stringdata0) + ofs \
        - idx * sizeof(QByteArrayData)) \
    )
```

Definition at line 25 of file moc_kbperf.cpp.

13.133.3 Variable Documentation

13.133.3.1 const uint qt_meta_data_KbPerf[] [static]

Initial value:

```
= {

       7,
       0,
       0,    0,
       2,   14,
       0,    0,
       0,    0,
       0,    0,
       0,
```

```
        2,

        1,    0,   24,    2, 0x06 ,
        3,    0,   25,    2, 0x06 ,

    QMetaType::Void,
    QMetaType::Void,

        0
}
```

Definition at line 42 of file moc_kbperf.cpp.

13.133.3.2 const qt_meta_stringdata_KbPerf_t qt_meta_stringdata_KbPerf [static]

Initial value:

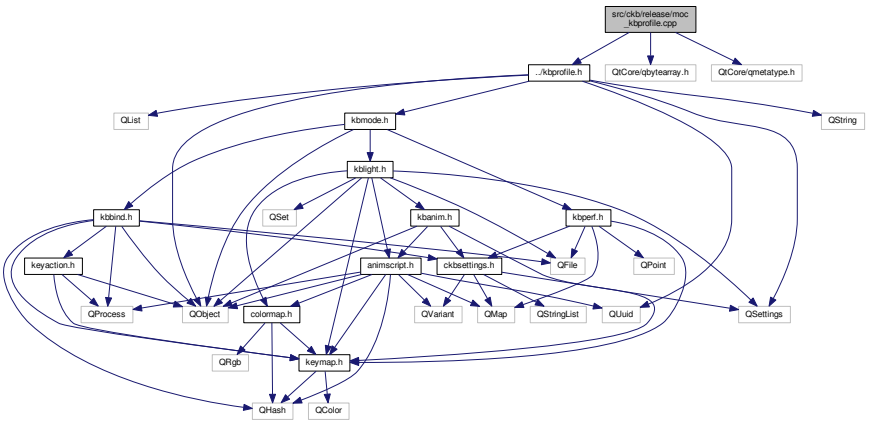
```
= {
    {
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 6 ,   qptrdiff(offsetof(
      qt_meta_stringdata_KbPerf_t, stringdata0) +  0    -  0  * sizeof(QByteArrayData)
      )   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 7 ,   qptrdiff(offsetof(
      qt_meta_stringdata_KbPerf_t, stringdata0) +  7    -  1  * sizeof(QByteArrayData)
      )   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 0 ,   qptrdiff(offsetof(
      qt_meta_stringdata_KbPerf_t, stringdata0) +  15    -  2  * sizeof(QByteArrayData
      ))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 15 ,   qptrdiff(offsetof(
      qt_meta_stringdata_KbPerf_t, stringdata0) +  16    -  3  * sizeof(QByteArrayData
      ))   )

    },
    "KbPerf\0didLoad\0\0settingsUpdated"
}
```

Definition at line 30 of file moc_kbperf.cpp.

13.134 src/ckb/release/moc_kbprofile.cpp File Reference

```
#include "../kbprofile.h"
#include <QtCore/qbytearray.h>
#include <QtCore/qmetatype.h>
```

Include dependency graph for moc_kbprofile.cpp:



Data Structures

- struct qt_meta_stringdata_KbProfile_t

Macros

- #define QT_MOC_LITERAL(idx, ofs, len)

Variables

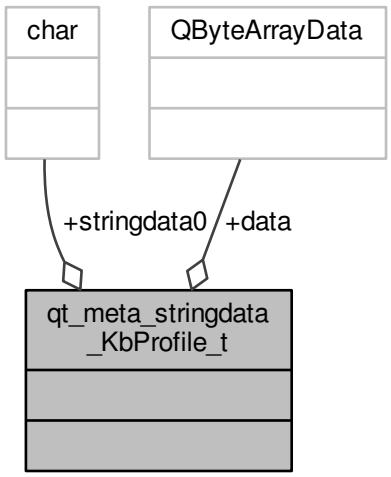
- static const qt_meta_stringdata_KbProfile_t qt_meta_stringdata_KbProfile
- static const uint qt_meta_data_KbProfile []

13.134.1 Data Structure Documentation

13.134.1.1 struct qt_meta_stringdata_KbProfile_t

Definition at line 21 of file moc_kbprofile.cpp.

Collaboration diagram for qt_meta_stringdata_KbProfile_t:



Data Fields

QByteArrayData	data[1]	
char	stringdata0[10]	

13.134.2 Macro Definition Documentation

13.134.2.1 #define QT_MOC_LITERAL(*idx, ofs, len*)

Value:

```
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET(len, \
    qptrdiff(offsetof(qt_meta_stringdata_KbProfile_t, stringdata0) + ofs \
        - idx * sizeof(QByteArrayData)) \
    )
```

Definition at line 25 of file moc_kbprofile.cpp.

13.134.3 Variable Documentation

13.134.3.1 const uint qt_meta_data_KbProfile[] [static]

Initial value:

```
= {

       7,
       0,
       0,    0,
       0,    0,
       0,    0,
       0,    0,
       0,    0,
       0,
       0,

       0
}
```

Definition at line 39 of file moc_kbprofile.cpp.

13.134.3.2 const qt_meta_stringdata_KbProfile_t qt_meta_stringdata_KbProfile [static]

Initial value:

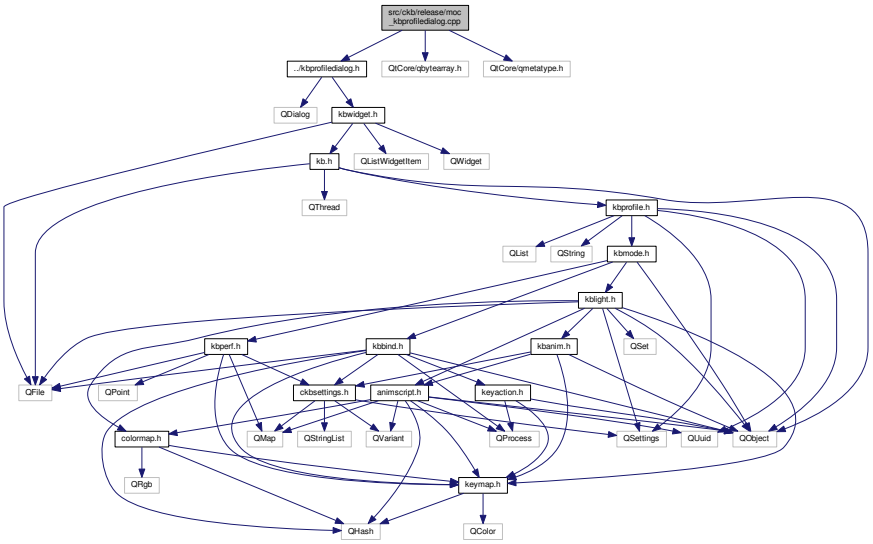
```
= {
    {
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 9 ,   qptrdiff(offsetof(
      qt_meta_stringdata_KbProfile_t, stringdata0) +  0    -  0  * sizeof(
      QByteArrayData))   )

    },
    "KbProfile"
}
```

Definition at line 30 of file moc_kbprofile.cpp.

13.135 src/ckb/release/moc_kbprofiledialog.cpp File Reference

```
#include "../kbprofiledialog.h"
#include <QtCore/qbytearray.h>
#include <QtCore/qmetatype.h>
```

Include dependency graph for moc_kbprofiledialog.cpp:



Data Structures

- struct qt_meta_stringdata_KbProfileDialog_t

Macros

- #define QT_MOC_LITERAL(idx, ofs, len)

Variables

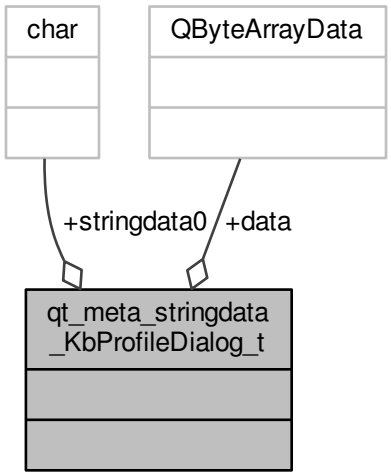
- static const qt_meta_stringdata_KbProfileDialog_t qt_meta_stringdata_KbProfileDialog
- static const uint qt_meta_data_KbProfileDialog []

13.135.1 Data Structure Documentation

13.135.1.1 struct qt_meta_stringdata_KbProfileDialog_t

Definition at line 21 of file moc_kbprofiledialog.cpp.

Collaboration diagram for qt_meta_stringdata_KbProfileDialog_t:



Data Fields

QByteArrayData	data[12]	
char	stringdata0[212]	

13.135.2 Macro Definition Documentation

13.135.2.1 #define QT_MOC_LITERAL(*idx, ofs, len*)

Value:

```
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET(len, \
    qptrdiff(offsetof(qt_meta_stringdata_KbProfileDialog_t, stringdata0
      ) + ofs \
        - idx * sizeof(QByteArrayData)) \
    )
```

Definition at line 25 of file moc_kbprofiledialog.cpp.

13.135.3 Variable Documentation

13.135.3.1 const uint qt_meta_data_KbProfileDialog[ ] [static]

Initial value:

```
= {

       7,
       0,
       0,    0,
       5,   14,
       0,    0,
       0,    0,
       0,    0,
```

```
        0,
        0,


        1,    0,   39,    2, 0x08 ,
        3,    2,   40,    2, 0x08 ,
        7,    1,   45,    2, 0x08 ,
        9,    1,   48,    2, 0x08 ,
       10,    1,   51,    2, 0x08 ,


    QMetaType::Void,
    QMetaType::Void, 0x80000000 | 4, 0x80000000 | 4,    5,    6,
    QMetaType::Void, 0x80000000 | 4,    8,
    QMetaType::Void, 0x80000000 | 4,    8,
    QMetaType::Void, QMetaType::QPoint,   11,

        0
}
```

Definition at line 56 of file moc_kbprofiledialog.cpp.

13.135.3.2 const qt_meta_stringdata_KbProfileDialog_t qt_meta_stringdata_KbProfileDialog [static]

Initial value:

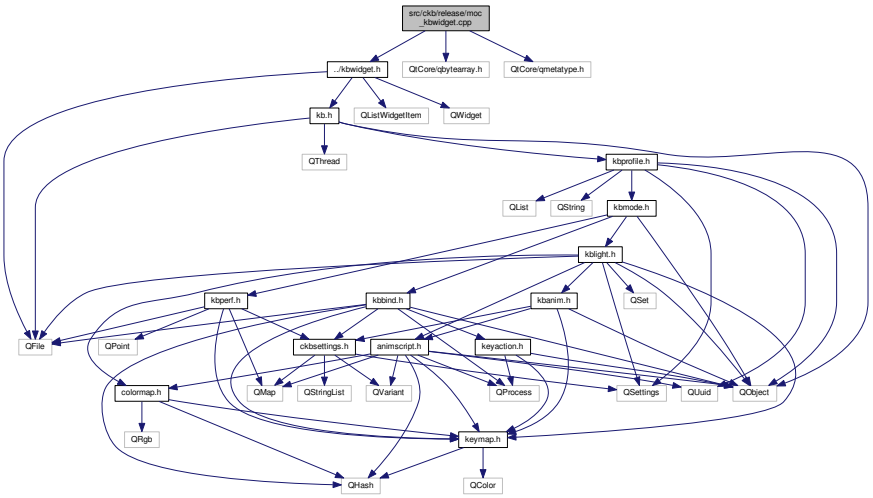
```
= {
    {
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 15 ,   qptrdiff(offsetof(
      qt_meta_stringdata_KbProfileDialog_t, stringdata0) +  0    -  0  *
      sizeof(QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 21 ,   qptrdiff(offsetof(
      qt_meta_stringdata_KbProfileDialog_t, stringdata0) +  16    -  1  *
      sizeof(QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 0 ,   qptrdiff(offsetof(
      qt_meta_stringdata_KbProfileDialog_t, stringdata0) +  38    -  2  *
      sizeof(QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 33 ,   qptrdiff(offsetof(
      qt_meta_stringdata_KbProfileDialog_t, stringdata0) +  39    -  3  *
      sizeof(QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 16 ,   qptrdiff(offsetof(
      qt_meta_stringdata_KbProfileDialog_t, stringdata0) +  73    -  4  *
      sizeof(QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 7 ,   qptrdiff(offsetof(
      qt_meta_stringdata_KbProfileDialog_t, stringdata0) +  90    -  5  *
      sizeof(QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 8 ,   qptrdiff(offsetof(
      qt_meta_stringdata_KbProfileDialog_t, stringdata0) +  98    -  6  *
      sizeof(QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 26 ,   qptrdiff(offsetof(
      qt_meta_stringdata_KbProfileDialog_t, stringdata0) +  107    -  7  *
      sizeof(QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 4 ,   qptrdiff(offsetof(
      qt_meta_stringdata_KbProfileDialog_t, stringdata0) +  134    -  8  *
      sizeof(QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 26 ,   qptrdiff(offsetof(
      qt_meta_stringdata_KbProfileDialog_t, stringdata0) +  139    -  9  *
      sizeof(QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 41 ,   qptrdiff(offsetof(
      qt_meta_stringdata_KbProfileDialog_t, stringdata0) +  166    -  10  *
      sizeof(QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 3 ,   qptrdiff(offsetof(
      qt_meta_stringdata_KbProfileDialog_t, stringdata0) +  208    -  11  *
      sizeof(QByteArrayData))   )

    },
    "KbProfileDialog\0profileList_reordered\0"
    "\0on_profileList_currentItemChanged\0"
    "QListWidgetItem*\0current\0previous\0"
    "on_profileList_itemClicked\0item\0"
    "on_profileList_itemChanged\0"
    "on_profileList_customContextMenuRequested\0"
    "pos"
}
```

Definition at line 30 of file moc_kbprofiledialog.cpp.

13.136 src/ckb/release/moc_kbwidget.cpp File Reference

```
#include "../kbwidget.h"
#include <QtCore/qbytearray.h>
#include <QtCore/qmetatype.h>
```

Include dependency graph for moc_kbwidget.cpp:



Data Structures

- struct qt_meta_stringdata_KbWidget_t

Macros

- #define QT_MOC_LITERAL(idx, ofs, len)

Variables

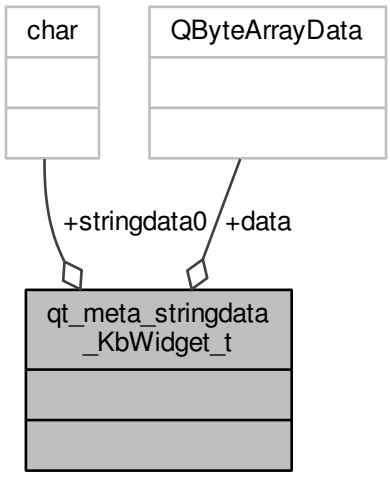
- static const qt_meta_stringdata_KbWidget_t qt_meta_stringdata_KbWidget
- static const uint qt_meta_data_KbWidget [ ]

13.136.1 Data Structure Documentation

13.136.1.1 struct qt_meta_stringdata_KbWidget_t

Definition at line 21 of file moc_kbwidget.cpp.

Collaboration diagram for qt_meta_stringdata_KbWidget_t:



Data Fields

QByteArrayData	data[29]	
char	stringdata0[442]	

13.136.2 Macro Definition Documentation

13.136.2.1 #define QT_MOC_LITERAL(*idx, ofs, len*)

Value:

```
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET(len, \
    qptrdiff(offsetof(qt_meta_stringdata_KbWidget_t, stringdata0) + ofs \
        - idx * sizeof(QByteArrayData)) \
    )
```

Definition at line 25 of file moc_kbwidget.cpp.

13.136.3 Variable Documentation

13.136.3.1 const uint qt_meta_data_KbWidget[] [static]

Definition at line 80 of file moc_kbwidget.cpp.

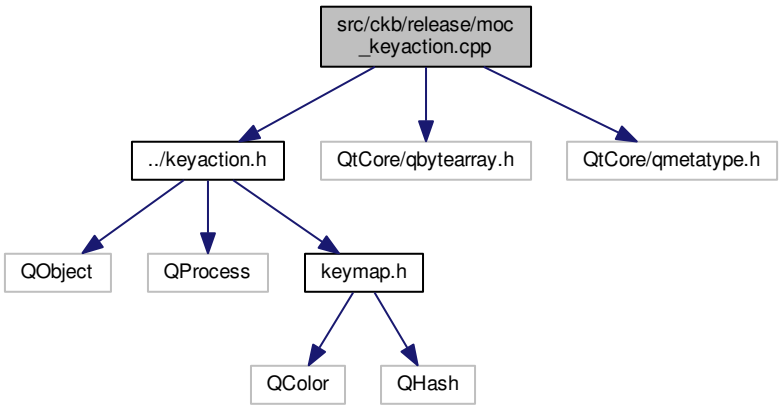
13.136.3.2 const qt_meta_stringdata_KbWidget_t qt_meta_stringdata_KbWidget [static]

Definition at line 30 of file moc_kbwidget.cpp.

13.137 src/ckb/release/moc_keyaction.cpp File Reference

```
#include "../keyaction.h"
#include <QtCore/qbytearray.h>
#include <QtCore/qmetatype.h>
```

Include dependency graph for moc_keyaction.cpp:



Data Structures

- struct qt_meta_stringdata_KeyAction_t

Macros

- #define QT_MOC_LITERAL(idx, ofs, len)

Variables

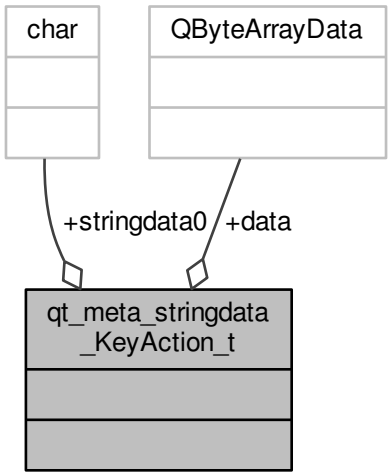
- static const qt_meta_stringdata_KeyAction_t qt_meta_stringdata_KeyAction
- static const uint qt_meta_data_KeyAction [ ]

13.137.1 Data Structure Documentation

13.137.1.1 struct qt_meta_stringdata_KeyAction_t

Definition at line 21 of file moc_keyaction.cpp.

Collaboration diagram for qt_meta_stringdata_KeyAction_t:



Data Fields

QByteArrayData	data[1]	
char	stringdata0[10]	

13.137.2 Macro Definition Documentation

13.137.2.1 #define QT_MOC_LITERAL(*idx, ofs, len*)

Value:

```
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET(len, \
    qptrdiff(offsetof(qt_meta_stringdata_KeyAction_t, stringdata0) + ofs \
        - idx * sizeof(QByteArrayData)) \
    )
```

Definition at line 25 of file moc_keyaction.cpp.

13.137.3 Variable Documentation

13.137.3.1 const uint qt_meta_data_KeyAction[ ] [static]

Initial value:

```
= {

       7,
       0,
       0,    0,
       0,    0,
       0,    0,
       0,    0,
       0,    0,
       0,
```

```
        0,

        0
}
```

Definition at line 39 of file moc_keyaction.cpp.

13.137.3.2 const qt_meta_stringdata_KeyAction_t qt_meta_stringdata_KeyAction [static]

Initial value:

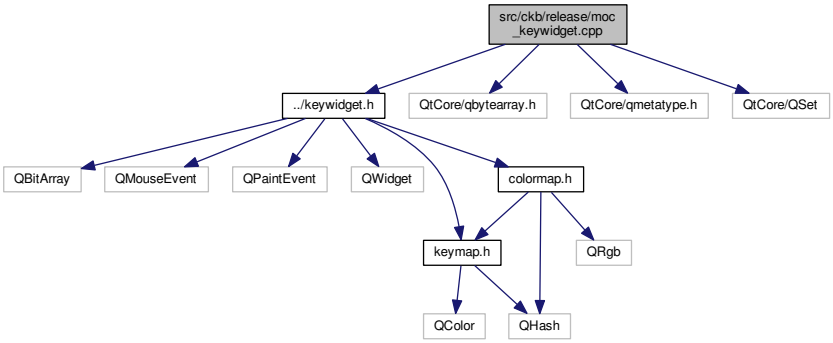
```
= {
    {
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 9 ,   qptrdiff(offsetof(
      qt_meta_stringdata_KeyAction_t, stringdata0) +  0    -  0  * sizeof(
      QByteArrayData))   )

    },
    "KeyAction"
}
```

Definition at line 30 of file moc_keyaction.cpp.

13.138 src/ckb/release/moc_keywidget.cpp File Reference

```
#include "../keywidget.h"
#include <QtCore/qbytearray.h>
#include <QtCore/qmetatype.h>
#include <QtCore/QSet>
```

Include dependency graph for moc_keywidget.cpp:



Data Structures

- struct qt_meta_stringdata_KeyWidget_t

Macros

- #define QT_MOC_LITERAL(idx, ofs, len)

Variables

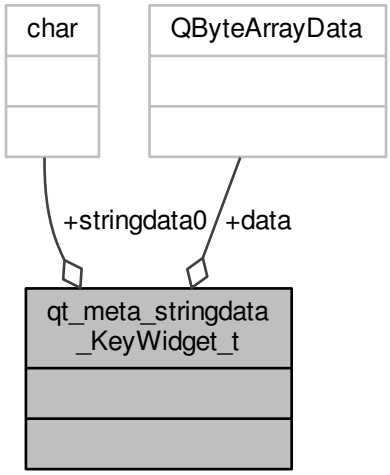
- static const qt_meta_stringdata_KeyWidget_t qt_meta_stringdata_KeyWidget
- static const uint qt_meta_data_KeyWidget [ ]

13.138.1 Data Structure Documentation

13.138.1.1 struct qt_meta_stringdata_KeyWidget_t

Definition at line 22 of file moc_keywidget.cpp.

Collaboration diagram for qt_meta_stringdata_KeyWidget_t:



Data Fields

QByteArrayData	data[10]	
char	stringdata0[118]	

13.138.2 Macro Definition Documentation

13.138.2.1 #define QT_MOC_LITERAL(*idx, ofs, len*)

Value:

```
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET(len, \
    qptrdiff(offsetof(qt_meta_stringdata_KeyWidget_t, stringdata0) + ofs \
        - idx * sizeof(QByteArrayData)) \
    )
```

Definition at line 26 of file moc_keywidget.cpp.

13.138.3 Variable Documentation

13.138.3.1 const uint qt_meta_data_KeyWidget[] [static]

Definition at line 52 of file moc_keywidget.cpp.

13.138.3.2 const qt_meta_stringdata_KeyWidget_t qt_meta_stringdata_KeyWidget [static]

Initial value:

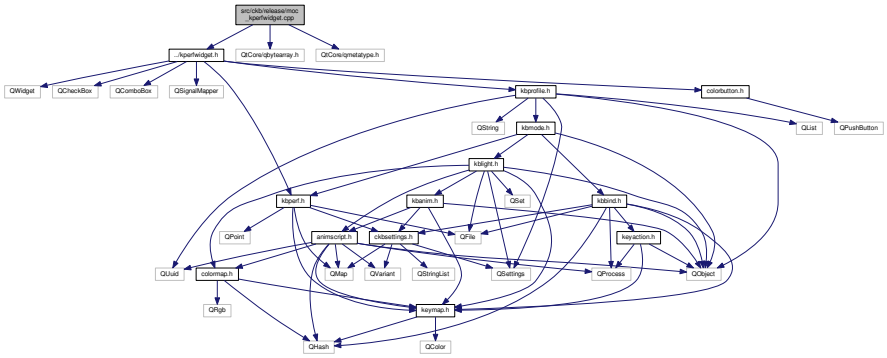
```
= {
    {
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 9 ,   qptrdiff(offsetof(
      qt_meta_stringdata_KeyWidget_t, stringdata0) +  0    -  0  * sizeof(
      QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 16 ,   qptrdiff(offsetof(
      qt_meta_stringdata_KeyWidget_t, stringdata0) +  10    -  1  * sizeof(
      QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 0 ,   qptrdiff(offsetof(
      qt_meta_stringdata_KeyWidget_t, stringdata0) +  27    -  2  * sizeof(
      QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 8 ,   qptrdiff(offsetof(
      qt_meta_stringdata_KeyWidget_t, stringdata0) +  28    -  3  * sizeof(
      QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 16 ,   qptrdiff(offsetof(
      qt_meta_stringdata_KeyWidget_t, stringdata0) +  37    -  4  * sizeof(
      QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 15 ,   qptrdiff(offsetof(
      qt_meta_stringdata_KeyWidget_t, stringdata0) +  54    -  5  * sizeof(
      QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 8 ,   qptrdiff(offsetof(
      qt_meta_stringdata_KeyWidget_t, stringdata0) +  70    -  6  * sizeof(
      QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 13 ,   qptrdiff(offsetof(
      qt_meta_stringdata_KeyWidget_t, stringdata0) +  79    -  7  * sizeof(
      QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 13 ,   qptrdiff(offsetof(
      qt_meta_stringdata_KeyWidget_t, stringdata0) +  93    -  8  * sizeof(
      QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 10 ,   qptrdiff(offsetof(
      qt_meta_stringdata_KeyWidget_t, stringdata0) +  107    -  9  * sizeof(
      QByteArrayData))   )

    },
    "KeyWidget\0selectionChanged\0\0selected\0"
    "sidelightToggled\0displayColorMap\0"
    "ColorMap\0newDisplayMap\0QSet<QString>\0"
    "indicators"
}
```

Definition at line 31 of file moc_keywidget.cpp.

13.139 src/ckb/release/moc_kperfwidget.cpp File Reference

```
#include "../kperfwidget.h"
#include <QtCore/qbytearray.h>
#include <QtCore/qmetatype.h>
```

Include dependency graph for moc_kperfwidget.cpp:



Data Structures

- struct qt_meta_stringdata_KPerfWidget_t

Macros

- #define QT_MOC_LITERAL(idx, ofs, len)

Variables

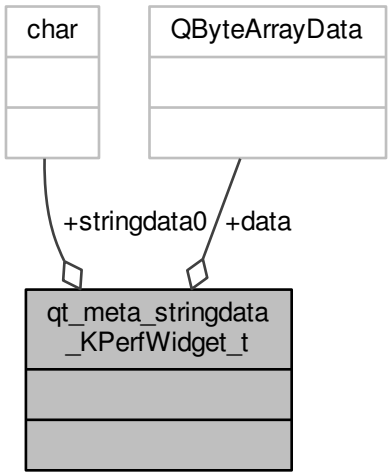
- static const qt_meta_stringdata_KPerfWidget_t qt_meta_stringdata_KPerfWidget
- static const uint qt_meta_data_KPerfWidget [ ]

13.139.1 Data Structure Documentation

13.139.1.1 struct qt_meta_stringdata_KPerfWidget_t

Definition at line 21 of file moc_kperfwidget.cpp.

Collaboration diagram for qt_meta_stringdata_KPerfWidget_t:



Data Fields

QByteArrayData	data[7]	
char	stringdata0[85]	

13.139.2 Macro Definition Documentation

13.139.2.1 #define QT_MOC_LITERAL(*idx, ofs, len*)

Value:

```
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET(len, \
    qptrdiff(offsetof(qt_meta_stringdata_KPerfWidget_t, stringdata0) + ofs
      \
        - idx * sizeof(QByteArrayData)) \
    )
```

Definition at line 25 of file moc_kperfwidget.cpp.

13.139.3 Variable Documentation

13.139.3.1 const uint qt_meta_data_KPerfWidget[] [static]

Initial value:

```
= {

       7,
       0,
       0,    0,
       3,   14,
       0,    0,
       0,    0,
       0,    0,
```

```
       0,
       0,


       1,    1,   29,    2, 0x08 ,
       4,    1,   32,    2, 0x08 ,
       6,    0,   35,    2, 0x08 ,


    QMetaType::Void, QMetaType::Int,    3,
    QMetaType::Void, QMetaType::Int,    5,
    QMetaType::Void,

       0
}
```

Definition at line 47 of file moc_kperfwidget.cpp.

13.139.3.2 const qt_meta_stringdata_KPerfWidget_t qt_meta_stringdata_KPerfWidget [static]

Initial value:

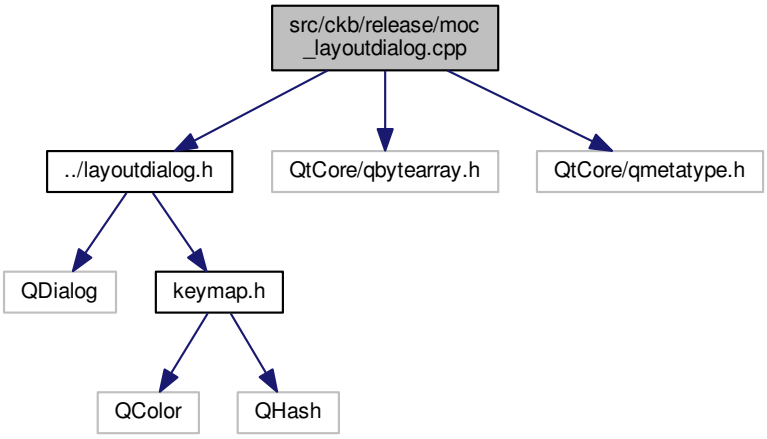
```
= {
    {
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 11 ,   qptrdiff(offsetof(
      qt_meta_stringdata_KPerfWidget_t, stringdata0) +  0    -  0  * sizeof(
      QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 9 ,   qptrdiff(offsetof(
      qt_meta_stringdata_KPerfWidget_t, stringdata0) +  12    -  1  * sizeof(
      QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 0 ,   qptrdiff(offsetof(
      qt_meta_stringdata_KPerfWidget_t, stringdata0) +  22    -  2  * sizeof(
      QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 5 ,   qptrdiff(offsetof(
      qt_meta_stringdata_KPerfWidget_t, stringdata0) +  23    -  3  * sizeof(
      QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 28 ,   qptrdiff(offsetof(
      qt_meta_stringdata_KPerfWidget_t, stringdata0) +  29    -  4  * sizeof(
      QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 4 ,   qptrdiff(offsetof(
      qt_meta_stringdata_KPerfWidget_t, stringdata0) +  58    -  5  * sizeof(
      QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 21 ,   qptrdiff(offsetof(
      qt_meta_stringdata_KPerfWidget_t, stringdata0) +  63    -  6  * sizeof(
      QByteArrayData))   )

    },
    "KPerfWidget\0uiUpdated\0\0index\0"
    "on_intensityBox_valueChanged\0arg1\0"
    "on_copyButton_clicked"
}
```

Definition at line 30 of file moc_kperfwidget.cpp.

13.140 src/ckb/release/moc_layoutdialog.cpp File Reference

```
#include "../layoutdialog.h"
#include <QtCore/qbytearray.h>
#include <QtCore/qmetatype.h>
```

Include dependency graph for moc_layoutdialog.cpp:



Data Structures

- struct qt_meta_stringdata_LayoutDialog_t

Macros

- #define QT_MOC_LITERAL(idx, ofs, len)

Variables

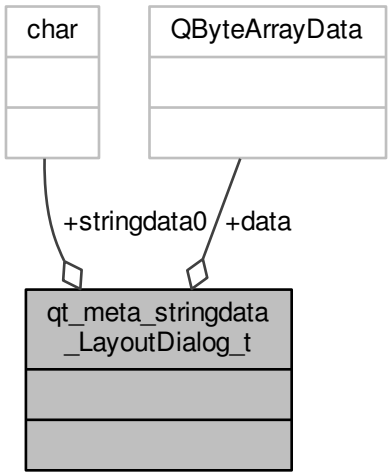
- static const qt_meta_stringdata_LayoutDialog_t qt_meta_stringdata_LayoutDialog
- static const uint qt_meta_data_LayoutDialog [ ]

13.140.1 Data Structure Documentation

13.140.1.1 struct qt_meta_stringdata_LayoutDialog_t

Definition at line 21 of file moc_layoutdialog.cpp.

Collaboration diagram for qt_meta_stringdata_LayoutDialog_t:



Data Fields

QByteArrayData	data[1]	
char	stringdata0[13]	

13.140.2 Macro Definition Documentation

13.140.2.1 #define QT_MOC_LITERAL(*idx, ofs, len*)

Value:

```
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET(len, \
    qptrdiff(offsetof(qt_meta_stringdata_LayoutDialog_t, stringdata0) +
      ofs \
        - idx * sizeof(QByteArrayData)) \
    )
```

Definition at line 25 of file moc_layoutdialog.cpp.

13.140.3 Variable Documentation

13.140.3.1 const uint qt_meta_data_LayoutDialog[] [static]

Initial value:

```
= {

       7,
       0,
       0,    0,
       0,    0,
       0,    0,
       0,    0,
       0,    0,
```

```
        0,
        0,

        0
}
```

Definition at line 39 of file moc_layoutdialog.cpp.

13.140.3.2 const qt_meta_stringdata_LayoutDialog_t qt_meta_stringdata_LayoutDialog [static]

Initial value:

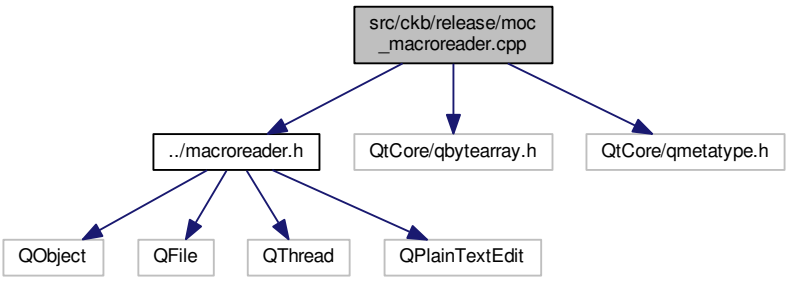
```
= {
    {
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 12 ,   qptrdiff(offsetof(
      qt_meta_stringdata_LayoutDialog_t, stringdata0) +  0    -  0  * sizeof(
      QByteArrayData))   )

    },
    "LayoutDialog"
}
```

Definition at line 30 of file moc_layoutdialog.cpp.

13.141 src/ckb/release/moc_macroreader.cpp File Reference

```
#include "../macroreader.h"
#include <QtCore/qbytearray.h>
#include <QtCore/qmetatype.h>
```

Include dependency graph for moc_macroreader.cpp:



Data Structures

- struct qt_meta_stringdata_MacroReaderThread_t
- struct qt_meta_stringdata_MacroReader_t

Macros

- #define QT_MOC_LITERAL(idx, ofs, len)
- #define QT_MOC_LITERAL(idx, ofs, len)

Variables

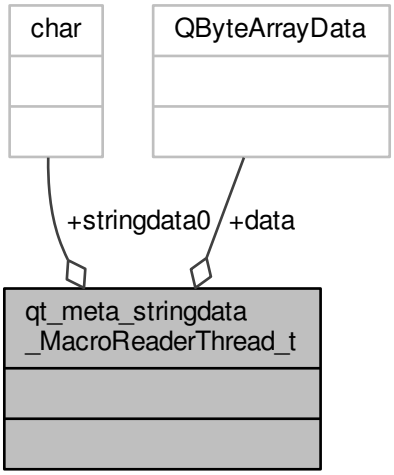
- static const qt_meta_stringdata_MacroReaderThread_t qt_meta_stringdata_MacroReaderThread
- static const uint qt_meta_data_MacroReaderThread []
- static const qt_meta_stringdata_MacroReader_t qt_meta_stringdata_MacroReader
- static const uint qt_meta_data_MacroReader []

13.141.1 Data Structure Documentation

13.141.1.1 struct qt_meta_stringdata_MacroReaderThread_t

Definition at line 21 of file moc_macroreader.cpp.

Collaboration diagram for qt_meta_stringdata_MacroReaderThread_t:



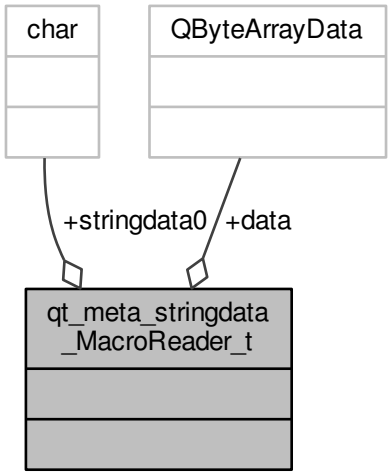
Data Fields

QByteArrayData	data[4]	
char	stringdata0[34]	

13.141.1.2 struct qt_meta_stringdata_MacroReader_t

Definition at line 111 of file moc_macroreader.cpp.

Collaboration diagram for qt_meta_stringdata_MacroReader_t:



Data Fields

QByteArrayData	data[1]	
char	stringdata0[12]	

13.141.2 Macro Definition Documentation

13.141.2.1 #define QT_MOC_LITERAL(*idx, ofs, len*)

Value:

```
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET(len, \
    qptrdiff(offsetof(qt_meta_stringdata_MacroReaderThread_t,
      stringdata0) + ofs \
        - idx * sizeof(QByteArrayData)) \
    )
```

Definition at line 115 of file moc_macroreader.cpp.

13.141.2.2 #define QT_MOC_LITERAL(*idx, ofs, len*)

Value:

```
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET(len, \
    qptrdiff(offsetof(qt_meta_stringdata_MacroReader_t, stringdata0) + ofs
      \
        - idx * sizeof(QByteArrayData)) \
    )
```

Definition at line 115 of file moc_macroreader.cpp.

13.141.3 Variable Documentation

13.141.3.1 const uint qt_meta_data_MacroReader[] [static]

Initial value:

```
= {

       7,
       0,
       0,    0,
       0,    0,
       0,    0,
       0,    0,
       0,    0,
       0,
       0,

       0
}
```

Definition at line 129 of file moc_macroreader.cpp.

13.141.3.2 const uint qt_meta_data_MacroReaderThread[] [static]

Initial value:

```
= {

       7,
       0,
       0,    0,
       1,   14,
       0,    0,
       0,    0,
       0,    0,
       0,
       0,


       1,    1,   19,    2, 0x08 ,


    QMetaType::Void, QMetaType::QString,    3,

       0
}
```

Definition at line 42 of file moc_macroreader.cpp.

13.141.3.3 const qt_meta_stringdata_MacroReader_t qt_meta_stringdata_MacroReader [static]

Initial value:

```
= {
    {
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 11 ,   qptrdiff(offsetof(
      qt_meta_stringdata_MacroReader_t, stringdata0) +  0    -  0  * sizeof(
      QByteArrayData))   )

    },
    "MacroReader"
}
```

Definition at line 120 of file moc_macroreader.cpp.

13.141.3.4 const qt_meta_stringdata_MacroReaderThread_t qt_meta_stringdata_MacroReaderThread [static]

Initial value:

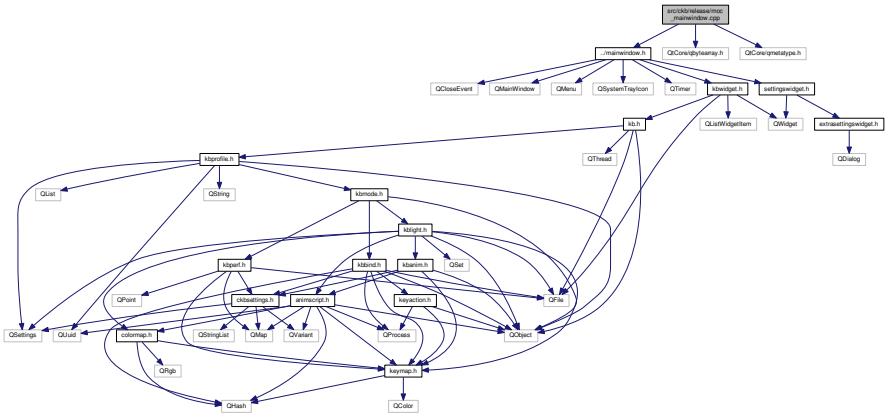
```
= {
    {
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 17 ,   qptrdiff(offsetof(
      qt_meta_stringdata_MacroReaderThread_t, stringdata0) +  0    -  0  *
      sizeof(QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 9 ,   qptrdiff(offsetof(
      qt_meta_stringdata_MacroReaderThread_t, stringdata0) +  18    -  1  *
       sizeof(QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 0 ,   qptrdiff(offsetof(
      qt_meta_stringdata_MacroReaderThread_t, stringdata0) +  28    -  2  *
       sizeof(QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 4 ,   qptrdiff(offsetof(
      qt_meta_stringdata_MacroReaderThread_t, stringdata0) +  29    -  3  *
       sizeof(QByteArrayData))   )

    },
    "MacroReaderThread\0readMacro\0\0line"
}
```

Definition at line 30 of file moc_macroreader.cpp.

13.142 src/ckb/release/moc_mainwindow.cpp File Reference

```
#include "../mainwindow.h"
#include <QtCore/qbytearray.h>
#include <QtCore/qmetatype.h>
```

Include dependency graph for moc_mainwindow.cpp:



Data Structures

- struct qt_meta_stringdata_MainWindow_t

Macros

- #define QT_MOC_LITERAL(idx, ofs, len)

Variables

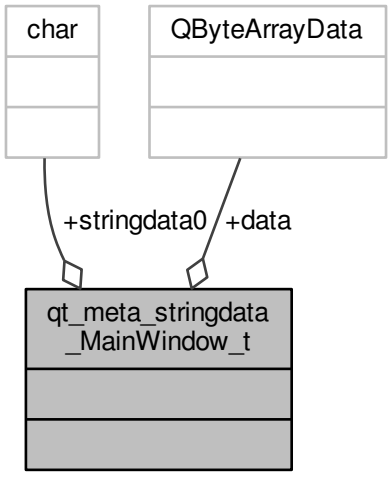
- static const qt_meta_stringdata_MainWindow_t qt_meta_stringdata_MainWindow
- static const uint qt_meta_data_MainWindow []

13.142.1 Data Structure Documentation

13.142.1.1 struct qt_meta_stringdata_MainWindow_t

Definition at line 21 of file moc_mainwindow.cpp.

Collaboration diagram for qt_meta_stringdata_MainWindow_t:



Data Fields

QByteArrayData	data[22]	
char	stringdata0[253]	

13.142.2 Macro Definition Documentation

13.142.2.1 #define QT_MOC_LITERAL(*idx, ofs, len*)

Value:

```
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET(len, \
    qptrdiff(offsetof(qt_meta_stringdata_MainWindow_t, stringdata0) + ofs \
        - idx * sizeof(QByteArrayData)) \
    )
```

Definition at line 25 of file moc_mainwindow.cpp.

13.142.3 Variable Documentation

13.142.3.1 const uint qt_meta_data_MainWindow[] [static]

Definition at line 66 of file moc_mainwindow.cpp.

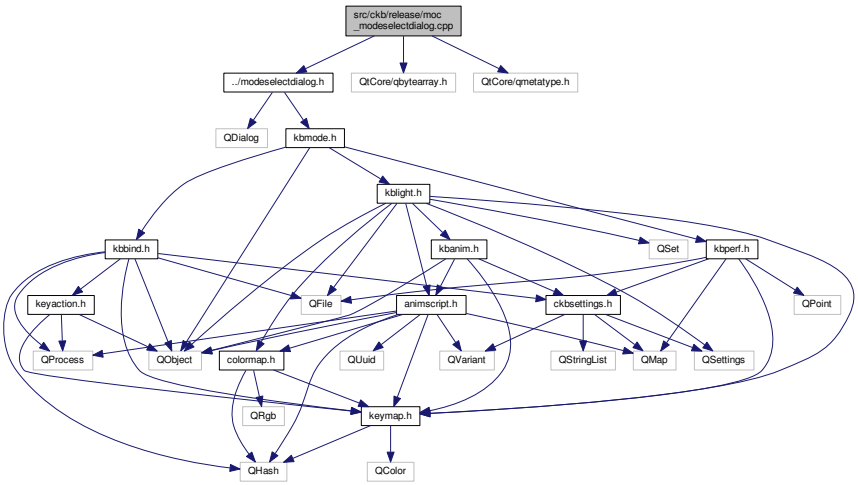
13.142.3.2 const qt_meta_stringdata_MainWindow_t qt_meta_stringdata_MainWindow [static]

Definition at line 30 of file moc_mainwindow.cpp.

13.143 src/ckb/release/moc_modeselectdialog.cpp File Reference

```
#include "../modeselectdialog.h"
#include <QtCore/qbytearray.h>
#include <QtCore/qmetatype.h>
```

Include dependency graph for moc_modeselectdialog.cpp:



Data Structures

- struct qt_meta_stringdata_ModeSelectDialog_t

Macros

- #define QT_MOC_LITERAL(idx, ofs, len)

Variables

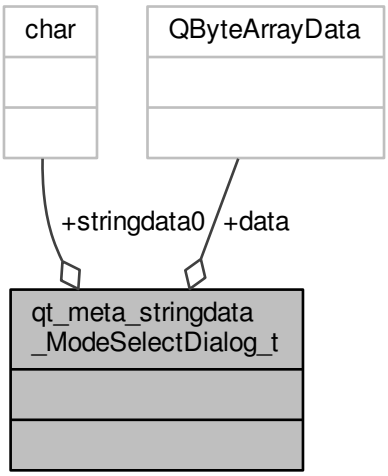
- static const qt_meta_stringdata_ModeSelectDialog_t qt_meta_stringdata_ModeSelectDialog
- static const uint qt_meta_data_ModeSelectDialog []

13.143.1 Data Structure Documentation

13.143.1.1 struct qt_meta_stringdata_ModeSelectDialog_t

Definition at line 21 of file moc_modeselectdialog.cpp.

Collaboration diagram for qt_meta_stringdata_ModeSelectDialog_t:



Data Fields

QByteArrayData	data[4]	
char	stringdata0[67]	

13.143.2 Macro Definition Documentation

13.143.2.1 #define QT_MOC_LITERAL(*idx, ofs, len*)

Value:

```
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET(len, \
    qptrdiff(offsetof(qt_meta_stringdata_ModeSelectDialog_t,
      stringdata0) + ofs \
        - idx * sizeof(QByteArrayData)) \
    )
```

Definition at line 25 of file moc_modeselectdialog.cpp.

13.143.3 Variable Documentation

13.143.3.1 const uint qt_meta_data_ModeSelectDialog[] [static]

Initial value:

```
= {

       7,
       0,
       0,    0,
       2,   14,
       0,    0,
       0,    0,
       0,    0,
```

```
        0,
        0,


        1,    0,   24,    2, 0x08 ,
        3,    0,   25,    2, 0x08 ,


    QMetaType::Void,
    QMetaType::Void,

        0
}
```

Definition at line 43 of file moc_modeselectdialog.cpp.

13.143.3.2 const qt_meta_stringdata_ModeSelectDialog_t qt_meta_stringdata_ModeSelectDialog [static]

Initial value:

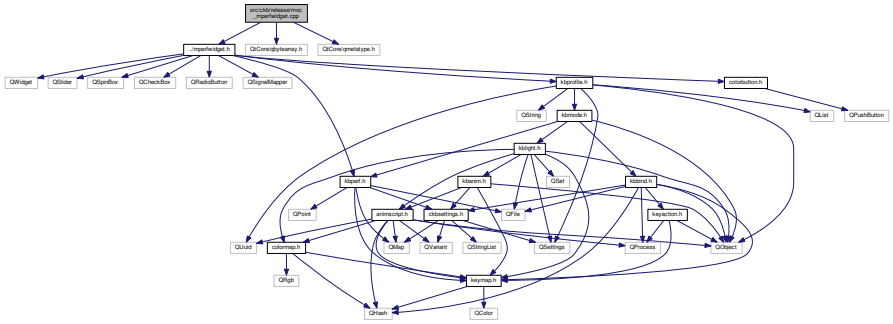
```
= {
    {
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 16 ,   qptrdiff(offsetof(
      qt_meta_stringdata_ModeSelectDialog_t, stringdata0) +  0    -  0  *
      sizeof(QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 23 ,   qptrdiff(offsetof(
      qt_meta_stringdata_ModeSelectDialog_t, stringdata0) +  17    -  1  *
      sizeof(QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 0 ,   qptrdiff(offsetof(
      qt_meta_stringdata_ModeSelectDialog_t, stringdata0) +  41    -  2  *
      sizeof(QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 24 ,   qptrdiff(offsetof(
      qt_meta_stringdata_ModeSelectDialog_t, stringdata0) +  42    -  3  *
      sizeof(QByteArrayData))   )

    },
    "ModeSelectDialog\0on_selAllButton_clicked\0"
    "\0on_selNoneButton_clicked"
}
```

Definition at line 30 of file moc_modeselectdialog.cpp.

13.144 src/ckb/release/moc_mperfwidget.cpp File Reference

```
#include "../mperfwidget.h"
#include <QtCore/qbytearray.h>
#include <QtCore/qmetatype.h>
```

Include dependency graph for moc_mperfwidget.cpp:



Data Structures

- struct qt_meta_stringdata_MPerfWidget_t

Macros

- #define QT_MOC_LITERAL(idx, ofs, len)

Variables

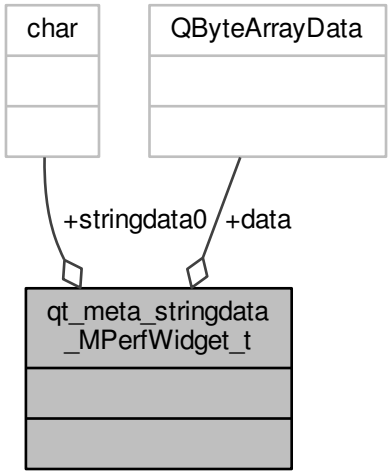
- static const qt_meta_stringdata_MPerfWidget_t qt_meta_stringdata_MPerfWidget
- static const uint qt_meta_data_MPerfWidget []

13.144.1 Data Structure Documentation

13.144.1.1 struct qt_meta_stringdata_MPerfWidget_t

Definition at line 21 of file moc_mperfwidget.cpp.

Collaboration diagram for qt_meta_stringdata_MPerfWidget_t:



Data Fields

QByteArrayData	data[18]	
char	stringdata0[249]	

13.144.2 Macro Definition Documentation

13.144.2.1 #define QT_MOC_LITERAL(*idx, ofs, len*)

Value:

```
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET(len, \
    qptrdiff(offsetof(qt_meta_stringdata_MPerfWidget_t, stringdata0) + ofs
      \
        - idx * sizeof(QByteArrayData)) \
    )
```

Definition at line 25 of file moc_mperfwidget.cpp.

13.144.3 Variable Documentation

13.144.3.1 const uint qt_meta_data_MPerfWidget[] [static]

Definition at line 62 of file moc_mperfwidget.cpp.

13.144.3.2 const qt_meta_stringdata_MPerfWidget_t qt_meta_stringdata_MPerfWidget [static]

Initial value:

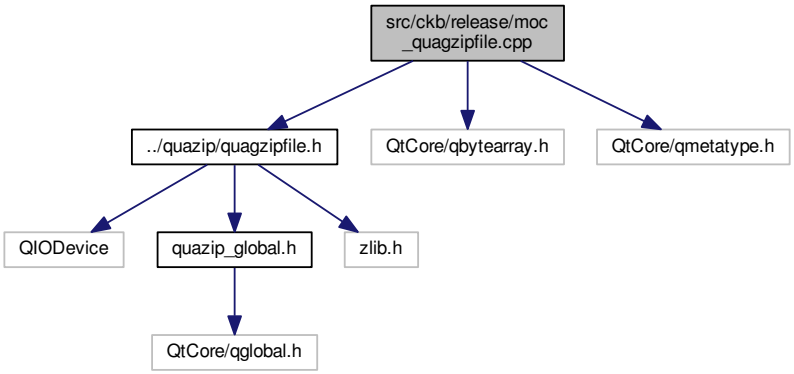
```
= {
    {
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 11 ,   qptrdiff(offsetof(
      qt_meta_stringdata_MPerfWidget_t, stringdata0) +  0    -  0  * sizeof(
      QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 12 ,   qptrdiff(offsetof(
      qt_meta_stringdata_MPerfWidget_t, stringdata0) +  12    -  1  * sizeof(
      QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 0 ,   qptrdiff(offsetof(
      qt_meta_stringdata_MPerfWidget_t, stringdata0) +  25    -  2  * sizeof(
      QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 5 ,   qptrdiff(offsetof(
      qt_meta_stringdata_MPerfWidget_t, stringdata0) +  26    -  3  * sizeof(
      QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 12 ,   qptrdiff(offsetof(
      qt_meta_stringdata_MPerfWidget_t, stringdata0) +  32    -  4  * sizeof(
      QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 12 ,   qptrdiff(offsetof(
      qt_meta_stringdata_MPerfWidget_t, stringdata0) +  45    -  5  * sizeof(
      QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 12 ,   qptrdiff(offsetof(
      qt_meta_stringdata_MPerfWidget_t, stringdata0) +  58    -  6  * sizeof(
      QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 11 ,   qptrdiff(offsetof(
      qt_meta_stringdata_MPerfWidget_t, stringdata0) +  71    -  7  * sizeof(
      QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 11 ,   qptrdiff(offsetof(
      qt_meta_stringdata_MPerfWidget_t, stringdata0) +  83    -  8  * sizeof(
      QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 13 ,   qptrdiff(offsetof(
      qt_meta_stringdata_MPerfWidget_t, stringdata0) +  95    -  9  * sizeof(
      QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 16 ,   qptrdiff(offsetof(
      qt_meta_stringdata_MPerfWidget_t, stringdata0) +  109    -  10  * sizeof(
      QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 7 ,   qptrdiff(offsetof(
      qt_meta_stringdata_MPerfWidget_t, stringdata0) +  126    -  11  * sizeof(
      QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 19 ,   qptrdiff(offsetof(
      qt_meta_stringdata_MPerfWidget_t, stringdata0) +  134    -  12  * sizeof(
      QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 19 ,   qptrdiff(offsetof(
      qt_meta_stringdata_MPerfWidget_t, stringdata0) +  154    -  13  * sizeof(
      QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 23 ,   qptrdiff(offsetof(
      qt_meta_stringdata_MPerfWidget_t, stringdata0) +  174    -  14  * sizeof(
      QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 21 ,   qptrdiff(offsetof(
      qt_meta_stringdata_MPerfWidget_t, stringdata0) +  198    -  15  * sizeof(
      QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 23 ,   qptrdiff(offsetof(
      qt_meta_stringdata_MPerfWidget_t, stringdata0) +  220    -  16  * sizeof(
      QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 4 ,   qptrdiff(offsetof(
      qt_meta_stringdata_MPerfWidget_t, stringdata0) +  244    -  17  * sizeof(
      QByteArrayData))   )

    },
    "MPerfWidget\0colorClicked\0\0index\0"
    "colorChanged\0sliderXMoved\0sliderYMoved\0"
    "boxXChanged\0boxYChanged\0enableChanged\0"
    "on_xyBox_clicked\0checked\0on_indicBox_clicked\0"
    "on_aSnapBox_clicked\0on_lHeightBox_activated\0"
    "on_copyButton_clicked\0on_spinBox_valueChanged\0"
    "arg1"
}
```

Definition at line 30 of file moc_mperfwidget.cpp.

13.145 src/ckb/release/moc_quagzipfile.cpp File Reference

```
#include "../quazip/quagzipfile.h"
#include <QtCore/qbytearray.h>
#include <QtCore/qmetatype.h>
```

Include dependency graph for moc_quagzipfile.cpp:



Data Structures

- struct qt_meta_stringdata_QuaGzipFile_t

Macros

- #define QT_MOC_LITERAL(idx, ofs, len)

Variables

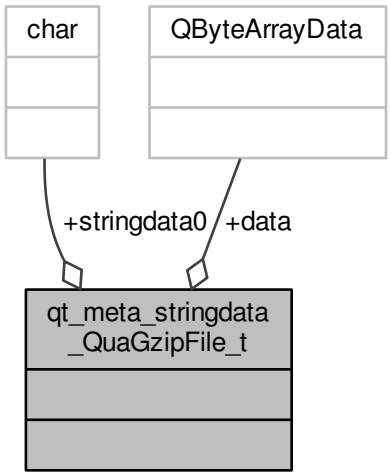
- static const qt_meta_stringdata_QuaGzipFile_t qt_meta_stringdata_QuaGzipFile
- static const uint qt_meta_data_QuaGzipFile []

13.145.1 Data Structure Documentation

13.145.1.1 struct qt_meta_stringdata_QuaGzipFile_t

Definition at line 21 of file moc_quagzipfile.cpp.

Collaboration diagram for qt_meta_stringdata_QuaGzipFile_t:



Data Fields

QByteArrayData	data[1]	
char	stringdata0[12]	

13.145.2 Macro Definition Documentation

13.145.2.1 #define QT_MOC_LITERAL(*idx, ofs, len*)

Value:

```
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET(len, \
    qptrdiff(offsetof(qt_meta_stringdata_QuaGzipFile_t, stringdata0) + ofs
      \
        - idx * sizeof(QByteArrayData)) \
    )
```

Definition at line 25 of file moc_quagzipfile.cpp.

13.145.3 Variable Documentation

13.145.3.1 const uint qt_meta_data_QuaGzipFile[] [static]

Initial value:

```
= {

       7,
       0,
       0,    0,
       0,    0,
       0,    0,
       0,    0,
       0,    0,
```

```
        0,
        0,

        0
}
```

Definition at line 39 of file moc_quagzipfile.cpp.

13.145.3.2 const qt_meta_stringdata_QuaGzipFile_t qt_meta_stringdata_QuaGzipFile [static]

Initial value:

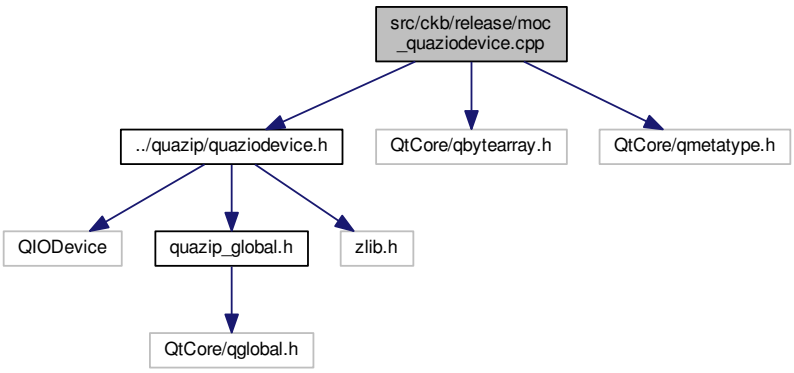
```
= {
    {
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 11 ,   qptrdiff(offsetof(
      qt_meta_stringdata_QuaGzipFile_t, stringdata0) +  0    -  0  * sizeof(
      QByteArrayData))   )

    },
    "QuaGzipFile"
}
```

Definition at line 30 of file moc_quagzipfile.cpp.

13.146 src/ckb/release/moc_quaziodevice.cpp File Reference

```
#include "../quazip/quaziodevice.h"
#include <QtCore/qbytearray.h>
#include <QtCore/qmetatype.h>
```

Include dependency graph for moc_quaziodevice.cpp:



Data Structures

- struct qt_meta_stringdata_QuaZIODevice_t

Macros

- #define QT_MOC_LITERAL(idx, ofs, len)

Variables

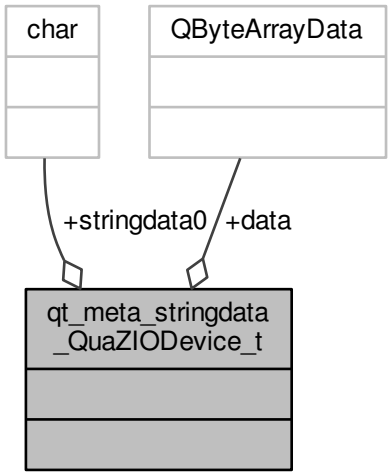
- static const qt_meta_stringdata_QuaZIODevice_t qt_meta_stringdata_QuaZIODevice
- static const uint qt_meta_data_QuaZIODevice []

13.146.1 Data Structure Documentation

13.146.1.1 struct qt_meta_stringdata_QuaZIODevice_t

Definition at line 21 of file moc_quaziodevice.cpp.

Collaboration diagram for qt_meta_stringdata_QuaZIODevice_t:



Data Fields

QByteArrayData	data[1]	
char	stringdata0[13]	

13.146.2 Macro Definition Documentation

13.146.2.1 #define QT_MOC_LITERAL(*idx, ofs, len*)

Value:

```
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET(len, \
    qptrdiff(offsetof(qt_meta_stringdata_QuaZIODevice_t, stringdata0) +
      ofs \
        - idx * sizeof(QByteArrayData)) \
    )
```

Definition at line 25 of file moc_quaziodevice.cpp.

13.146.3 Variable Documentation

13.146.3.1 const uint qt_meta_data_QuaZIODevice[] [static]

Initial value:

```
= {

       7,
       0,
       0,    0,
       0,    0,
       0,    0,
       0,    0,
       0,    0,
       0,
       0,

       0
}
```

Definition at line 39 of file moc_quaziodevice.cpp.

13.146.3.2 const qt_meta_stringdata_QuaZIODevice_t qt_meta_stringdata_QuaZIODevice [static]

Initial value:

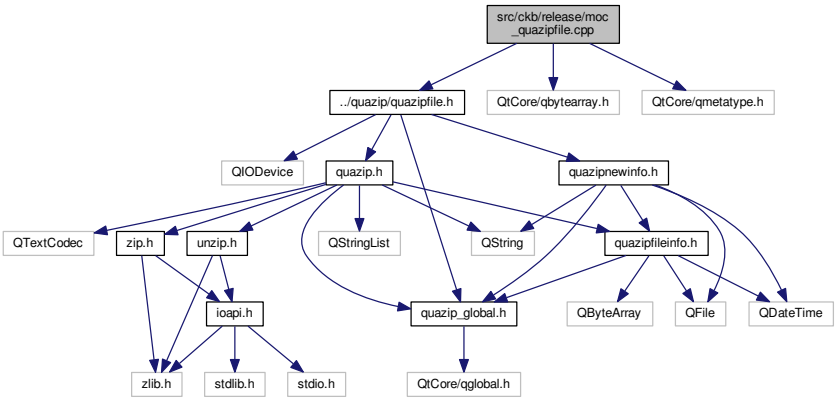
```
= {
    {
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 12 ,   qptrdiff(offsetof(
      qt_meta_stringdata_QuaZIODevice_t, stringdata0) +  0    -  0  * sizeof(
      QByteArrayData))   )

    },
    "QuaZIODevice"
}
```

Definition at line 30 of file moc_quaziodevice.cpp.

13.147 src/ckb/release/moc_quazipfile.cpp File Reference

```
#include "../quazip/quazipfile.h"
#include <QtCore/qbytearray.h>
#include <QtCore/qmetatype.h>
```

Include dependency graph for moc_quazipfile.cpp:



Data Structures

- struct qt_meta_stringdata_QuaZipFile_t

Macros

- #define QT_MOC_LITERAL(idx, ofs, len)

Variables

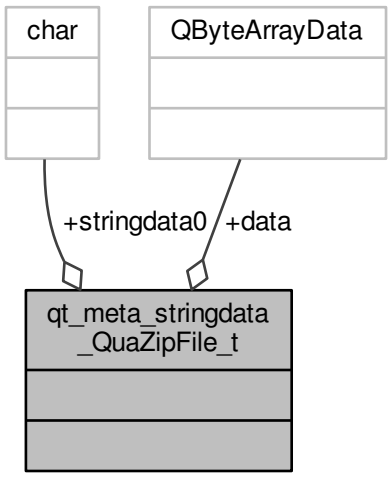
- static const qt_meta_stringdata_QuaZipFile_t qt_meta_stringdata_QuaZipFile
- static const uint qt_meta_data_QuaZipFile []

13.147.1 Data Structure Documentation

13.147.1.1 struct qt_meta_stringdata_QuaZipFile_t

Definition at line 21 of file moc_quazipfile.cpp.

Collaboration diagram for qt_meta_stringdata_QuaZipFile_t:



Data Fields

QByteArrayData	data[1]	
char	stringdata0[11]	

13.147.2 Macro Definition Documentation

13.147.2.1 #define QT_MOC_LITERAL(*idx, ofs, len*)

Value:

```
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET(len, \
    qptrdiff(offsetof(qt_meta_stringdata_QuaZipFile_t, stringdata0) + ofs \
        - idx * sizeof(QByteArrayData)) \
    )
```

Definition at line 25 of file moc_quazipfile.cpp.

13.147.3 Variable Documentation

13.147.3.1 const uint qt_meta_data_QuaZipFile[] [static]

Initial value:

```
= {

       7,
       0,
       0,    0,
       0,    0,
       0,    0,
       0,    0,
       0,    0,
       0,
       0,

       0
}
```

Definition at line 39 of file moc_quazipfile.cpp.

13.147.3.2 const qt_meta_stringdata_QuaZipFile_t qt_meta_stringdata_QuaZipFile [static]

Initial value:

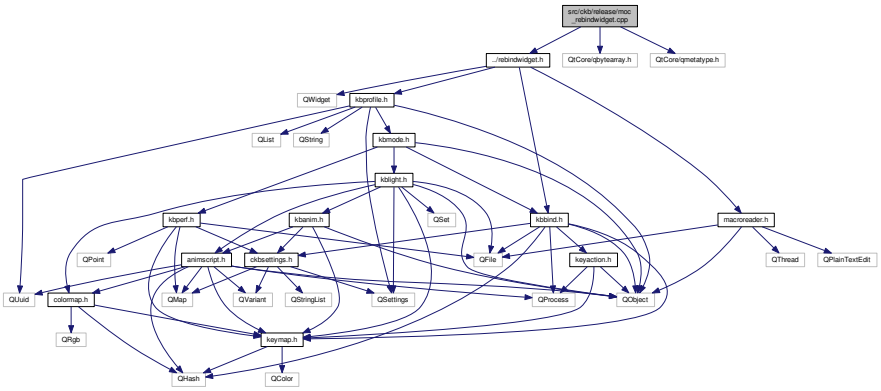
```
= {
    {
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 10 ,   qptrdiff(offsetof(
      qt_meta_stringdata_QuaZipFile_t, stringdata0) +  0    -  0  * sizeof(
      QByteArrayData))   )

    },
    "QuaZipFile"
}
```

Definition at line 30 of file moc_quazipfile.cpp.

13.148 src/ckb/release/moc_rebindwidget.cpp File Reference

```
#include "../rebindwidget.h"
#include <QtCore/qbytearray.h>
#include <QtCore/qmetatype.h>
```

Include dependency graph for moc_rebindwidget.cpp:



Data Structures

- struct qt_meta_stringdata_RebindWidget_t

Macros

- #define QT_MOC_LITERAL(idx, ofs, len)

Variables

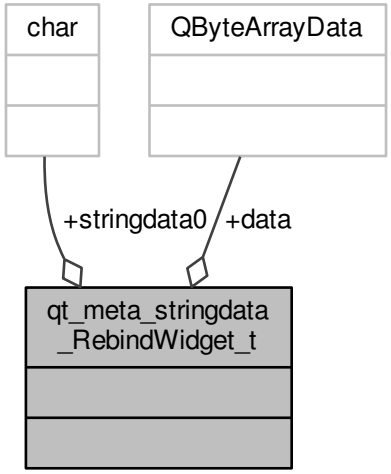
- static const qt_meta_stringdata_RebindWidget_t qt_meta_stringdata_RebindWidget
- static const uint qt_meta_data_RebindWidget [ ]

13.148.1 Data Structure Documentation

13.148.1.1 struct qt_meta_stringdata_RebindWidget_t

Definition at line 21 of file moc_rebindwidget.cpp.

Collaboration diagram for qt_meta_stringdata_RebindWidget_t:



Data Fields

QByteArrayData	data[51]	
char	stringdata0[1179]	

13.148.2 Macro Definition Documentation

13.148.2.1 #define QT_MOC_LITERAL(*idx, ofs, len*)

Value:

```
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET(len, \
    qptrdiff(offsetof(qt_meta_stringdata_RebindWidget_t, stringdata0) +
      ofs \
        - idx * sizeof(QByteArrayData)) \
    )
```

Definition at line 25 of file moc_rebindwidget.cpp.

13.148.3 Variable Documentation

13.148.3.1 const uint qt_meta_data_RebindWidget[ ] [static]

Definition at line 124 of file moc_rebindwidget.cpp.

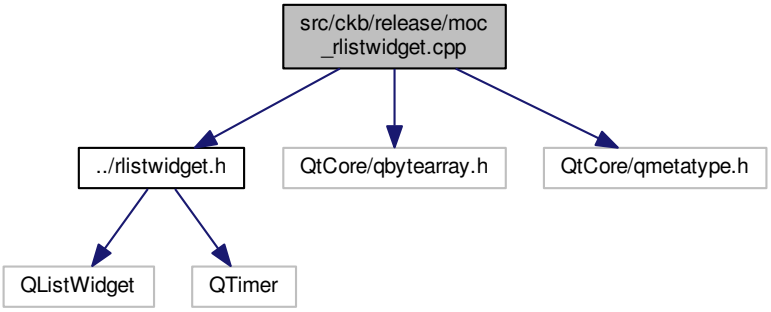
13.148.3.2 const qt_meta_stringdata_RebindWidget_t qt_meta_stringdata_RebindWidget [static]

Definition at line 30 of file moc_rebindwidget.cpp.

13.149 src/ckb/release/moc_rlistwidget.cpp File Reference

```
#include "../rlistwidget.h"
#include <QtCore/qbytearray.h>
#include <QtCore/qmetatype.h>
```

Include dependency graph for moc_rlistwidget.cpp:



Data Structures

- struct qt_meta_stringdata_RListWidget_t

Macros

- #define QT_MOC_LITERAL(idx, ofs, len)

Variables

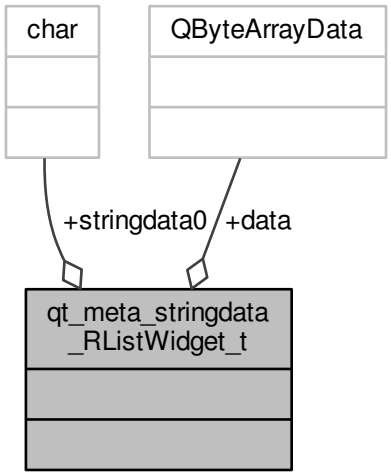
- static const qt_meta_stringdata_RListWidget_t qt_meta_stringdata_RListWidget
- static const uint qt_meta_data_RListWidget []

13.149.1 Data Structure Documentation

13.149.1.1 struct qt_meta_stringdata_RListWidget_t

Definition at line 21 of file moc_rlistwidget.cpp.

Collaboration diagram for qt_meta_stringdata_RListWidget_t:



Data Fields

QByteArrayData	data[8]	
char	stringdata0[71]	

13.149.2 Macro Definition Documentation

13.149.2.1 #define QT_MOC_LITERAL(*idx, ofs, len*)

Value:

```
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET(len, \
    qptrdiff(offsetof(qt_meta_stringdata_RListWidget_t, stringdata0) + ofs
      \
        - idx * sizeof(QByteArrayData)) \
    )
```

Definition at line 25 of file moc_rlistwidget.cpp.

13.149.3 Variable Documentation

13.149.3.1 const uint qt_meta_data_RListWidget[] [static]

Definition at line 47 of file moc_rlistwidget.cpp.

13.149.3.2 const qt_meta_stringdata_RListWidget_t qt_meta_stringdata_RListWidget [static]

Initial value:

```
= {
    {
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 11 ,   qptrdiff(offsetof(
```

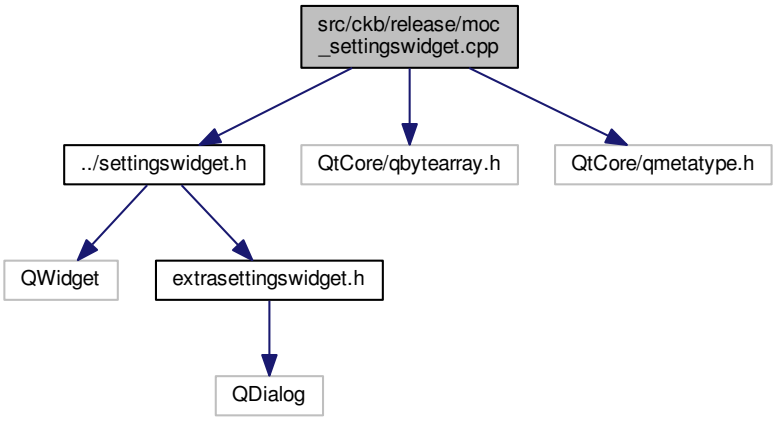
```
      qt_meta_stringdata_RListWidget_t, stringdata0) +  0    -  0  * sizeof(
      QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 12 ,   qptrdiff(offsetof(
      qt_meta_stringdata_RListWidget_t, stringdata0) +  12    -  1  * sizeof(
      QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 0 ,   qptrdiff(offsetof(
      qt_meta_stringdata_RListWidget_t, stringdata0) +  25    -  2  * sizeof(
      QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 9 ,   qptrdiff(offsetof(
      qt_meta_stringdata_RListWidget_t, stringdata0) +  26    -  3  * sizeof(
      QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 5 ,   qptrdiff(offsetof(
      qt_meta_stringdata_RListWidget_t, stringdata0) +  36    -  4  * sizeof(
      QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 16 ,   qptrdiff(offsetof(
      qt_meta_stringdata_RListWidget_t, stringdata0) +  42    -  5  * sizeof(
      QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 4 ,   qptrdiff(offsetof(
      qt_meta_stringdata_RListWidget_t, stringdata0) +  59    -  6  * sizeof(
      QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 6 ,   qptrdiff(offsetof(
      qt_meta_stringdata_RListWidget_t, stringdata0) +  64    -  7  * sizeof(
      QByteArrayData))   )

    },
    "RListWidget\0orderChanged\0\0timerTick\0"
    "enter\0QListWidgetItem*\0item\0change"
}
```

Definition at line 30 of file moc_rlistwidget.cpp.

13.150 src/ckb/release/moc_settingswidget.cpp File Reference

```
#include "../settingswidget.h"
#include <QtCore/qbytearray.h>
#include <QtCore/qmetatype.h>
```

Include dependency graph for moc_settingswidget.cpp:



Data Structures

- struct qt_meta_stringdata_SettingsWidget_t

Macros

- #define QT_MOC_LITERAL(idx, ofs, len)

Variables

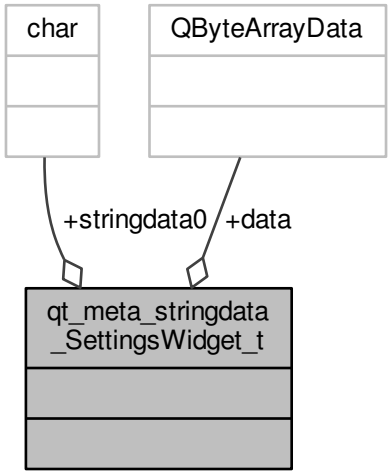
- static const qt_meta_stringdata_SettingsWidget_t qt_meta_stringdata_SettingsWidget
- static const uint qt_meta_data_SettingsWidget []

13.150.1 Data Structure Documentation

13.150.1.1 struct qt_meta_stringdata_SettingsWidget_t

Definition at line 21 of file moc_settingswidget.cpp.

Collaboration diagram for qt_meta_stringdata_SettingsWidget_t:



Data Fields

QByteArrayData	data[15]	
char	stringdata0[264]	

13.150.2 Macro Definition Documentation

13.150.2.1 #define QT_MOC_LITERAL(*idx, ofs, len*)

Value:

```
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET(len, \
    qptrdiff(offsetof(qt_meta_stringdata_SettingsWidget_t, stringdata0)
      + ofs \
        - idx * sizeof(QByteArrayData)) \
    )
```

Definition at line 25 of file moc_settingswidget.cpp.

13.150.3 Variable Documentation

13.150.3.1 const uint qt_meta_data_SettingsWidget[] [static]

Definition at line 59 of file moc_settingswidget.cpp.

13.150.3.2 const qt_meta_stringdata_SettingsWidget_t qt_meta_stringdata_SettingsWidget [static]

Initial value:

```
= {
    {
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 14 ,   qptrdiff(offsetof(
      qt_meta_stringdata_SettingsWidget_t, stringdata0) +  0    -  0  * sizeof
      (QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 21 ,   qptrdiff(offsetof(
      qt_meta_stringdata_SettingsWidget_t, stringdata0) +  15    -  1  *
      sizeof(QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 0 ,   qptrdiff(offsetof(
      qt_meta_stringdata_SettingsWidget_t, stringdata0) +  37    -  2  *
      sizeof(QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 20 ,   qptrdiff(offsetof(
      qt_meta_stringdata_SettingsWidget_t, stringdata0) +  38    -  3  *
      sizeof(QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 5 ,   qptrdiff(offsetof(
      qt_meta_stringdata_SettingsWidget_t, stringdata0) +  59    -  4  *
      sizeof(QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 21 ,   qptrdiff(offsetof(
      qt_meta_stringdata_SettingsWidget_t, stringdata0) +  65    -  5  *
      sizeof(QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 20 ,   qptrdiff(offsetof(
      qt_meta_stringdata_SettingsWidget_t, stringdata0) +  87    -  6  *
      sizeof(QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 19 ,   qptrdiff(offsetof(
      qt_meta_stringdata_SettingsWidget_t, stringdata0) +  108    -  7  *
      sizeof(QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 19 ,   qptrdiff(offsetof(
      qt_meta_stringdata_SettingsWidget_t, stringdata0) +  128    -  8  *
      sizeof(QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 20 ,   qptrdiff(offsetof(
      qt_meta_stringdata_SettingsWidget_t, stringdata0) +  148    -  9  *
      sizeof(QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 7 ,   qptrdiff(offsetof(
      qt_meta_stringdata_SettingsWidget_t, stringdata0) +  169    -  10  *
      sizeof(QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 23 ,   qptrdiff(offsetof(
      qt_meta_stringdata_SettingsWidget_t, stringdata0) +  177    -  11  *
      sizeof(QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 22 ,   qptrdiff(offsetof(
      qt_meta_stringdata_SettingsWidget_t, stringdata0) +  201    -  12  *
      sizeof(QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 16 ,   qptrdiff(offsetof(
      qt_meta_stringdata_SettingsWidget_t, stringdata0) +  224    -  13  *
      sizeof(QByteArrayData))   ),
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET( 22 ,   qptrdiff(offsetof(
      qt_meta_stringdata_SettingsWidget_t, stringdata0) +  241    -  14  *
      sizeof(QByteArrayData))   )

    },
    "SettingsWidget\0on_pushButton_clicked\0"
    "\0on_capsBox_activated\0index\0"
    "on_shiftBox_activated\0on_ctrlBox_activated\0"
    "on_altBox_activated\0on_winBox_activated\0"
    "on_autoFWBox_clicked\0checked\0"
    "on_loginItemBox_clicked\0on_layoutBox_activated\0"
    "showLayoutDialog\0on_extraButton_clicked"
}
```

Definition at line 30 of file moc_settingswidget.cpp.

13.151 src/ckb/release/qrc_binary.cpp File Reference

Macros

- #define QT_RCC_PREPEND_NAMESPACE(name) name
- #define QT_RCC_MANGLE_NAMESPACE(name) name

Functions

- bool qRegisterResourceData (int, const unsigned char *, const unsigned char *, const unsigned char *)
- bool qUnregisterResourceData (int, const unsigned char *, const unsigned char *, const unsigned char *)
- int qInitResources_binary ()
- int qCleanupResources_binary ()

Variables

- static const unsigned char qt_resource_data []
- static const unsigned char qt_resource_name []
- static const unsigned char qt_resource_struct []

13.151.1 Macro Definition Documentation

13.151.1.1 #define QT_RCC_MANGLE_NAMESPACE(*name*) name

Definition at line 147 of file qrc_binary.cpp.

13.151.1.2 #define QT_RCC_PREPEND_NAMESPACE(*name*) name

Definition at line 146 of file qrc_binary.cpp.

Referenced by qCleanupResources_binary(), and qInitResources_binary().

13.151.2 Function Documentation

13.151.2.1 int qCleanupResources_binary ( )

Definition at line 171 of file qrc_binary.cpp.

References QT_RCC_PREPEND_NAMESPACE, qt_resource_data, qt_resource_name, qt_resource_struct, and qUnregisterResourceData().

```
172 {
173     QT_RCC_PREPEND_NAMESPACE(qUnregisterResourceData)
174         (0x01, qt_resource_struct, qt_resource_name,
    qt_resource_data);
175     return 1;
176 }
```

Here is the call graph for this function:

13.151.2.2 int qInitResources_binary ( )

Definition at line 163 of file qrc_binary.cpp.

References qRegisterResourceData(), QT_RCC_PREPEND_NAMESPACE, qt_resource_data, qt_resource_name, and qt_resource_struct.

```
164 {
165     QT_RCC_PREPEND_NAMESPACE(qRegisterResourceData)
166         (0x01, qt_resource_struct, qt_resource_name,
      qt_resource_data);
167     return 1;
168 }
```

Here is the call graph for this function:

13.151.2.3 bool qRegisterResourceData (int , const unsigned char * , const unsigned char * , const unsigned char *)

Referenced by qInitResources_binary().

Here is the caller graph for this function:

13.151.2.4 bool qUnregisterResourceData (int , const unsigned char * , const unsigned char * , const unsigned char *)

Referenced by qCleanupResources_binary().

Here is the caller graph for this function:

13.151.3 Variable Documentation

13.151.3.1 const unsigned char qt_resource_data[] [static]

Definition at line 9 of file qrc_binary.cpp.

Referenced by qCleanupResources_binary(), and qInitResources_binary().

13.151.3.2 const unsigned char qt_resource_name[] [static]

Initial value:

```
= {

  0x0,0x3,
  0x0,0x0,0x68,0xfe,
  0x0,0x62,
  0x0,0x69,0x0,0x6e,

  0x0,0x10,
  0x9,0x4b,0x25,0x27,
  0x0,0x63,
  0x0,0x6b,0x0,0x62,0x0,0x2d,0x0,0x6e,0x0,0x65,0x0,0x78,0x0,0x74,0x0,0x2d,0x0,0x6b,0x0,0x65,0x0,0x79,0x0,
       0x2e,0x0,0x67,0x0,0x70,0x0,0x67,

}
```

Definition at line 114 of file qrc_binary.cpp.

Referenced by qCleanupResources_binary(), and qInitResources_binary().

13.151.3.3 const unsigned char qt_resource_struct[] [static]

Initial value:

```
= {

  0x0,0x0,0x0,0x0,0x0,0x2,0x0,0x0,0x0,0x1,0x0,0x0,0x0,0x1,

  0x0,0x0,0x0,0x0,0x0,0x2,0x0,0x0,0x0,0x1,0x0,0x0,0x0,0x2,

  0x0,0x0,0x0,0xc,0x0,0x0,0x0,0x0,0x0,0x1,0x0,0x0,0x0,0x0,

}
```

Definition at line 128 of file qrc_binary.cpp.

Referenced by qCleanupResources_binary(), and qInitResources_binary().

13.152 src/ckb/release/qrc_image.cpp File Reference

Macros

- #define QT_RCC_PREPEND_NAMESPACE(name) name
- #define QT_RCC_MANGLE_NAMESPACE(name) name

Functions

- bool qRegisterResourceData (int, const unsigned char *, const unsigned char *, const unsigned char *)
- bool qUnregisterResourceData (int, const unsigned char *, const unsigned char *, const unsigned char *)
- int qInitResources_image ()
- int qCleanupResources_image ()

Variables

- static const unsigned char qt_resource_data []
- static const unsigned char qt_resource_name []
- static const unsigned char qt_resource_struct []

13.152.1 Macro Definition Documentation

13.152.1.1 #define QT_RCC_MANGLE_NAMESPACE(*name*) name

Definition at line 14160 of file qrc_image.cpp.

13.152.1.2 #define QT_RCC_PREPEND_NAMESPACE(*name*) name

Definition at line 14159 of file qrc_image.cpp.

Referenced by qCleanupResources_image(), and qInitResources_image().

13.152.2 Function Documentation

13.152.2.1 int qCleanupResources_image ( )

Definition at line 14184 of file qrc_image.cpp.

References QT_RCC_PREPEND_NAMESPACE, qt_resource_data, qt_resource_name, qt_resource_struct, and qUnregisterResourceData().

```
14185 {
14186     QT_RCC_PREPEND_NAMESPACE(qUnregisterResourceData)
14187         (0x01, qt_resource_struct, qt_resource_name,
      qt_resource_data);
14188     return 1;
14189 }
```

Here is the call graph for this function:

13.152.2.2 int qInitResources_image ( )

Definition at line 14176 of file qrc_image.cpp.

References qRegisterResourceData(), QT_RCC_PREPEND_NAMESPACE, qt_resource_data, qt_resource_-name, and qt_resource_struct.

```
14177 {
14178     QT_RCC_PREPEND_NAMESPACE(qRegisterResourceData)
14179         (0x01, qt_resource_struct, qt_resource_name,
      qt_resource_data);
14180     return 1;
14181 }
```

Here is the call graph for this function:

13.152.2.3 bool qRegisterResourceData (int , const unsigned char * , const unsigned char * , const unsigned char *)

Referenced by qInitResources_image().

Here is the caller graph for this function:

13.152.2.4 bool qUnregisterResourceData (int , const unsigned char * , const unsigned char * , const unsigned char *)

Referenced by qCleanupResources_image().

Here is the caller graph for this function:

13.152.3 Variable Documentation

13.152.3.1 const unsigned char qt_resource_data[] [static]

Definition at line 9 of file qrc_image.cpp.

Referenced by qCleanupResources_image(), and qInitResources_image().

13.152.3.2 const unsigned char qt_resource_name[] [static]

Definition at line 14007 of file qrc_image.cpp.

Referenced by qCleanupResources_image(), and qInitResources_image().

13.152.3.3 const unsigned char qt_resource_struct[] [static]

Definition at line 14109 of file qrc_image.cpp.

Referenced by qCleanupResources_image(), and qInitResources_image().

13.153 src/ckb/release/qrc_text.cpp File Reference

Macros

- #define QT_RCC_PREPEND_NAMESPACE(name) name
- #define QT_RCC_MANGLE_NAMESPACE(name) name

Functions

- bool qRegisterResourceData (int, const unsigned char *, const unsigned char *, const unsigned char *)
- bool qUnregisterResourceData (int, const unsigned char *, const unsigned char *, const unsigned char *)
- int qInitResources_text ()
- int qCleanupResources_text ()

Variables

- static const unsigned char qt_resource_data []
- static const unsigned char qt_resource_name []
- static const unsigned char qt_resource_struct []

13.153.1 Macro Definition Documentation

13.153.1.1 #define QT_RCC_MANGLE_NAMESPACE(*name*) name

Definition at line 350 of file qrc_text.cpp.

13.153.1.2 #define QT_RCC_PREPEND_NAMESPACE(*name*) name

Definition at line 349 of file qrc_text.cpp.

Referenced by qCleanupResources_text(), and qInitResources_text().

13.153.2 Function Documentation

13.153.2.1 int qCleanupResources_text ( )

Definition at line 374 of file qrc_text.cpp.

References QT_RCC_PREPEND_NAMESPACE, qt_resource_data, qt_resource_name, qt_resource_struct, and qUnregisterResourceData().

```
375 {
376     QT_RCC_PREPEND_NAMESPACE(qUnregisterResourceData)
377         (0x01, qt_resource_struct, qt_resource_name,
      qt_resource_data);
378     return 1;
379 }
```

Here is the call graph for this function:

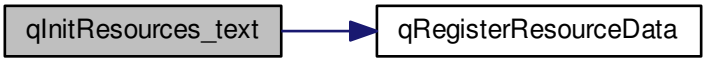
13.153.2.2 int qInitResources_text ( )

Definition at line 366 of file qrc_text.cpp.

References qRegisterResourceData(), QT_RCC_PREPEND_NAMESPACE, qt_resource_data, qt_resource_name, and qt_resource_struct.

```
367 {
368     QT_RCC_PREPEND_NAMESPACE(qRegisterResourceData)
369         (0x01, qt_resource_struct, qt_resource_name,
      qt_resource_data);
370     return 1;
371 }
```

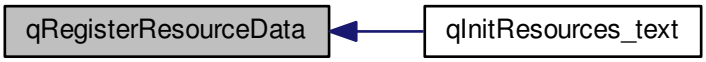
Here is the call graph for this function:



13.153.2.3 bool qRegisterResourceData (int , const unsigned char * , const unsigned char * , const unsigned char *)

Referenced by qInitResources_text().

Here is the caller graph for this function:



13.153.2.4 bool qUnregisterResourceData (int , const unsigned char * , const unsigned char * , const unsigned char *)

Referenced by qCleanupResources_text().

Here is the caller graph for this function:

13.153.3 Variable Documentation

13.153.3.1 const unsigned char qt_resource_data[] [static]

Definition at line 9 of file qrc_text.cpp.

Referenced by qCleanupResources_text(), and qInitResources_text().

13.153.3.2 const unsigned char qt_resource_name[] [static]

Initial value:

```
= {

  0x0,0x3,
  0x0,0x0,0x7b,0xf4,
  0x0,0x74,
  0x0,0x78,0x0,0x74,

  0x0,0xb,
  0xa,0x1e,0x1,0xe0,
  0x0,0x63,
  0x0,0x6b,0x0,0x62,0x0,0x2e,0x0,0x64,0x0,0x65,0x0,0x73,0x0,0x6b,0x0,0x74,0x0,0x6f,0x0,0x70,

  0x0,0x10,
  0xe,0xac,0xaf,0x46,
  0x0,0x64,
  0x0,0x65,0x0,0x6d,0x0,0x6f,0x0,0x70,0x0,0x72,0x0,0x6f,0x0,0x66,0x0,0x69,0x0,0x6c,0x0,0x65,0x0,0x2e,0x0,
       0x63,0x0,0x6f,0x0,0x6e,0x0,0x66,

  0x0,0x11,
  0x0,0xd4,0x25,0xa4,
  0x0,0x63,
  0x0,0x6f,0x0,0x6d,0x0,0x2e,0x0,0x63,0x0,0x6b,0x0,0x62,0x0,0x2e,0x0,0x63,0x0,0x6b,0x0,0x62,0x0,0x2e,0x0,
       0x70,0x0,0x6c,0x0,0x69,0x0,0x73,0x0,0x74,


}
```

Definition at line 302 of file qrc_text.cpp.

Referenced by qCleanupResources_text(), and qInitResources_text().

13.153.3.3 const unsigned char qt_resource_struct[] [static]

Initial value:

```
= {

  0x0,0x0,0x0,0x0,0x0,0x2,0x0,0x0,0x0,0x1,0x0,0x0,0x0,0x1,
```

```
  0x0,0x0,0x0,0x0,0x0,0x2,0x0,0x0,0x0,0x3,0x0,0x0,0x0,0x2,
  0x0,0x0,0x0,0x4e,0x0,0x0,0x0,0x0,0x0,0x1,0x0,0x0,0xf,0x74,
  0x0,0x0,0x0,0xc,0x0,0x0,0x0,0x0,0x0,0x1,0x0,0x0,0x0,0x0,
  0x0,0x0,0x0,0x28,0x0,0x1,0x0,0x0,0x0,0x1,0x0,0x0,0x0,0xc6,
}
```

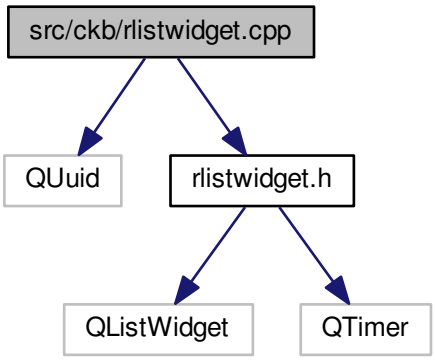
Definition at line 327 of file qrc_text.cpp.

Referenced by qCleanupResources_text(), and qInitResources_text().

13.154 src/ckb/rlistwidget.cpp File Reference

```
#include <QUuid>
#include "rlistwidget.h"
```

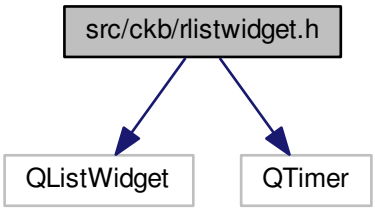
Include dependency graph for rlistwidget.cpp:



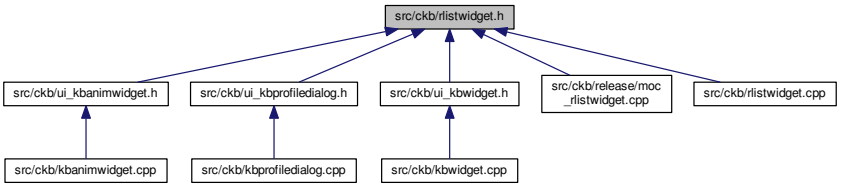
13.155 src/ckb/rlistwidget.h File Reference

```
#include <QListWidget>
#include <QTimer>
```

Include dependency graph for rlistwidget.h:



This graph shows which files directly or indirectly include this file:



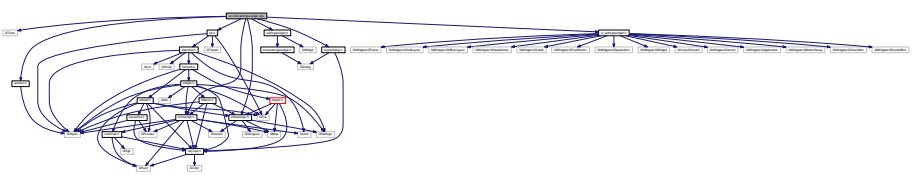
Data Structures

- class RListWidget

13.156 src/ckb/settingswidget.cpp File Reference

```
#include <QTimer>
#include "animscript.h"
#include "autorun.h"
#include "ckbsettings.h"
#include "kb.h"
#include "layoutdialog.h"
#include "settingswidget.h"
#include "ui_settingswidget.h"
```

Include dependency graph for settingswidget.cpp:



Functions

- static QString right (const QString &left)

Variables

- QString devpath
- static QStringList modKeys
- static QStringList modNames

13.156.1 Function Documentation

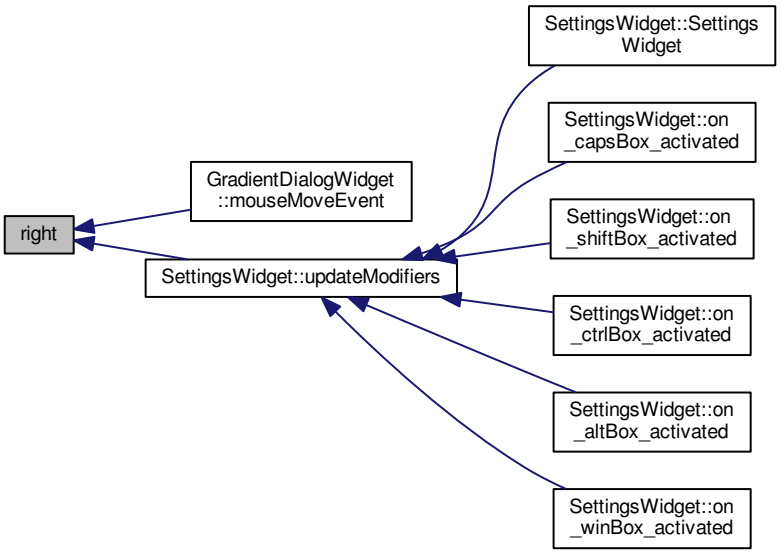
13.156.1.1 static QString right (const QString & *left*) [inline], [static]

Definition at line 118 of file settingswidget.cpp.

Referenced by GradientDialogWidget::mouseMoveEvent(), and SettingsWidget::updateModifiers().

```
118                                                 {
119     if(left[0] != 'l')
120         return left;
121     return "r" + left.mid(1);
122 }
```

Here is the caller graph for this function:



13.156.2 Variable Documentation

13.156.2.1 QString devpath

Definition at line 4 of file kbmanager.cpp.

Referenced by KbManager::scanKeyboards().

13.156.2.2 QStringList modKeys [static]

Definition at line 13 of file settingswidget.cpp.

Referenced by SettingsWidget::SettingsWidget(), and SettingsWidget::updateModifiers().

13.156.2.3 QStringList modNames [static]

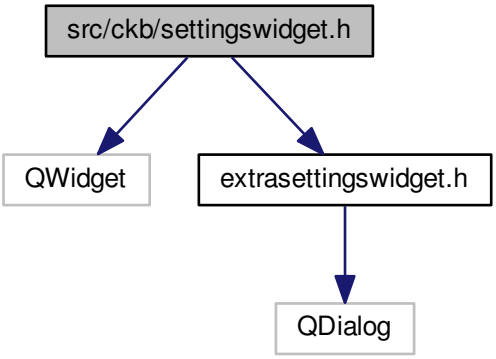
Definition at line 13 of file settingswidget.cpp.

Referenced by SettingsWidget::SettingsWidget().

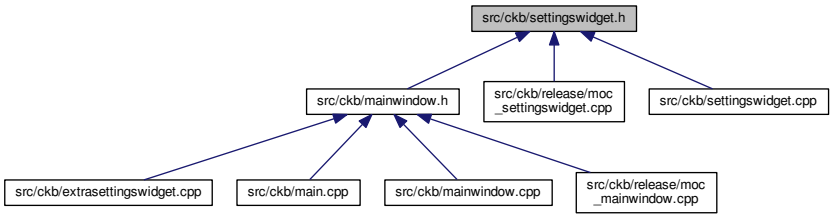
13.157 src/ckb/settingswidget.h File Reference

```
#include <QWidget>
#include "extrasettingswidget.h"
```

Include dependency graph for settingswidget.h:



This graph shows which files directly or indirectly include this file:



Data Structures

- class SettingsWidget

Namespaces

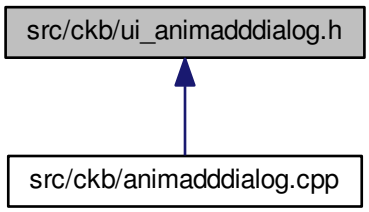
- Ui

13.158 src/ckb/ui_animadddialog.h File Reference

```
#include <QtCore/QVariant>
#include <QtWidgets/QAction>
#include <QtWidgets/QApplication>
#include <QtWidgets/QButtonGroup>
#include <QtWidgets/QCheckBox>
#include <QtWidgets/QComboBox>
#include <QtWidgets/QDialog>
#include <QtWidgets/QDialogButtonBox>
#include <QtWidgets/QGridLayout>
#include <QtWidgets/QHBoxLayout>
#include <QtWidgets/QHeaderView>
#include <QtWidgets/QLabel>
#include <QtWidgets/QSpacerItem>
```

Include dependency graph for ui_animadddialog.h:

This graph shows which files directly or indirectly include this file:



Data Structures

- class Ui_AnimAddDialog
- class Ui::AnimAddDialog

Namespaces

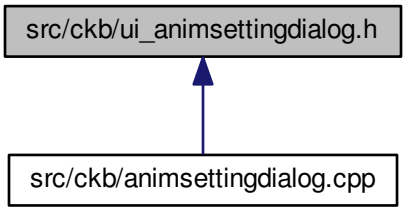
- Ui

13.159 src/ckb/ui_animsettingdialog.h File Reference

```
#include <QtCore/QVariant>
#include <QtWidgets/QAction>
#include <QtWidgets/QApplication>
#include <QtWidgets/QButtonGroup>
#include <QtWidgets/QCheckBox>
#include <QtWidgets/QDialog>
#include <QtWidgets/QDialogButtonBox>
#include <QtWidgets/QDoubleSpinBox>
#include <QtWidgets/QFrame>
#include <QtWidgets/QGridLayout>
#include <QtWidgets/QHeaderView>
#include <QtWidgets/QLabel>
#include <QtWidgets/QLineEdit>
#include <QtWidgets/QSpacerItem>
#include <QtWidgets/QTabWidget>
#include <QtWidgets/QWidget>
```

Include dependency graph for ui_animsettingdialog.h:

This graph shows which files directly or indirectly include this file:



Data Structures

- class Ui_AnimSettingDialog
- class Ui::AnimSettingDialog

Namespaces

- Ui

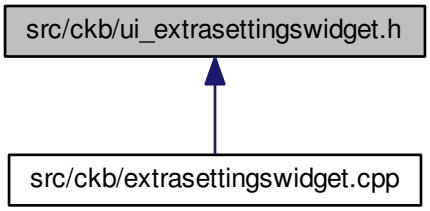
13.160 src/ckb/ui_extrasettingswidget.h File Reference

```
#include <QtCore/QVariant>
```

```
#include <QtWidgets/QAction>
#include <QtWidgets/QApplication>
#include <QtWidgets/QButtonGroup>
#include <QtWidgets/QCheckBox>
#include <QtWidgets/QDialog>
#include <QtWidgets/QDialogButtonBox>
#include <QtWidgets/QFrame>
#include <QtWidgets/QGridLayout>
#include <QtWidgets/QHBoxLayout>
#include <QtWidgets/QHeaderView>
#include <QtWidgets/QLabel>
#include <QtWidgets/QPushButton>
#include <QtWidgets/QSpacerItem>
#include <QtWidgets/QSpinBox>
#include <QtWidgets/QWidget>
```

Include dependency graph for ui_extrasettingswidget.h:

This graph shows which files directly or indirectly include this file:



Data Structures

- class Ui_ExtraSettingsWidget
- class Ui::ExtraSettingsWidget

Namespaces

- Ui

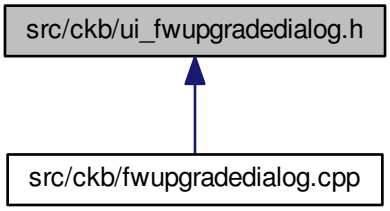
13.161 src/ckb/ui_fwupgradedialog.h File Reference

```
#include <QtCore/QVariant>
```

```
#include <QtWidgets/QAction>
#include <QtWidgets/QApplication>
#include <QtWidgets/QButtonGroup>
#include <QtWidgets/QDialog>
#include <QtWidgets/QGridLayout>
#include <QtWidgets/QHeaderView>
#include <QtWidgets/QLabel>
#include <QtWidgets/QProgressBar>
#include <QtWidgets/QPushButton>
#include <QtWidgets/QSpacerItem>
```

Include dependency graph for ui_fwupgradedialog.h:

This graph shows which files directly or indirectly include this file:



Data Structures

- class Ui_FwUpgradeDialog
- class Ui::FwUpgradeDialog

Namespaces

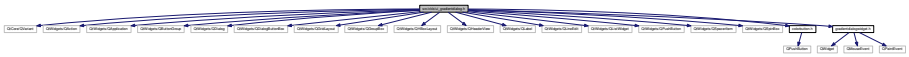
- Ui

13.162 src/ckb/ui_gradientdialog.h File Reference

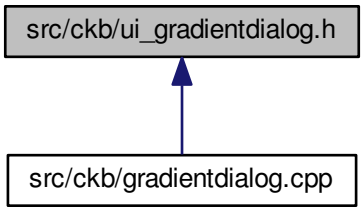
```
#include <QtCore/QVariant>
```

```
#include <QtWidgets/QAction>
#include <QtWidgets/QApplication>
#include <QtWidgets/QButtonGroup>
#include <QtWidgets/QDialog>
#include <QtWidgets/QDialogButtonBox>
#include <QtWidgets/QGridLayout>
#include <QtWidgets/QGroupBox>
#include <QtWidgets/QHBoxLayout>
#include <QtWidgets/QHeaderView>
#include <QtWidgets/QLabel>
#include <QtWidgets/QLineEdit>
#include <QtWidgets/QListWidget>
#include <QtWidgets/QPushButton>
#include <QtWidgets/QSpacerItem>
#include <QtWidgets/QSpinBox>
#include "colorbutton.h"
#include "gradientdialogwidget.h"
```

Include dependency graph for ui_gradientdialog.h:



This graph shows which files directly or indirectly include this file:



Data Structures

- class Ui_GradientDialog
- class Ui::GradientDialog

Namespaces

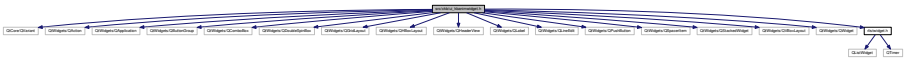
- Ui

13.163 src/ckb/ui_kbanimwidget.h File Reference

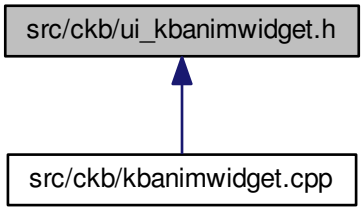
```
#include <QtCore/QVariant>
```

```
#include <QtWidgets/QAction>
#include <QtWidgets/QApplication>
#include <QtWidgets/QButtonGroup>
#include <QtWidgets/QComboBox>
#include <QtWidgets/QDoubleSpinBox>
#include <QtWidgets/QGridLayout>
#include <QtWidgets/QHBoxLayout>
#include <QtWidgets/QHeaderView>
#include <QtWidgets/QLabel>
#include <QtWidgets/QLineEdit>
#include <QtWidgets/QPushButton>
#include <QtWidgets/QSpacerItem>
#include <QtWidgets/QStackedWidget>
#include <QtWidgets/QVBoxLayout>
#include <QtWidgets/QWidget>
#include "rlistwidget.h"
```

Include dependency graph for ui_kbanimwidget.h:



This graph shows which files directly or indirectly include this file:



Data Structures

- class Ui_KbAnimWidget
- class Ui::KbAnimWidget

Namespaces

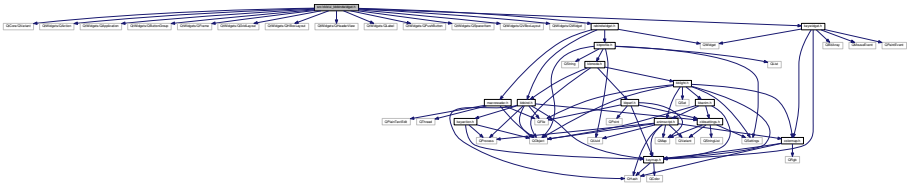
- Ui

13.164 src/ckb/ui_kbbindwidget.h File Reference

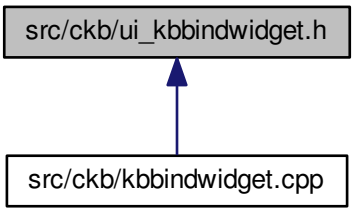
```
#include <QtCore/QVariant>
```

```
#include <QtWidgets/QAction>
#include <QtWidgets/QApplication>
#include <QtWidgets/QButtonGroup>
#include <QtWidgets/QFrame>
#include <QtWidgets/QGridLayout>
#include <QtWidgets/QHBoxLayout>
#include <QtWidgets/QHeaderView>
#include <QtWidgets/QLabel>
#include <QtWidgets/QPushButton>
#include <QtWidgets/QSpacerItem>
#include <QtWidgets/QVBoxLayout>
#include <QtWidgets/QWidget>
#include "keywidget.h"
#include "rebindwidget.h"
```

Include dependency graph for ui_kbbindwidget.h:



This graph shows which files directly or indirectly include this file:



Data Structures

- class Ui_KbBindWidget
- class Ui::KbBindWidget

Namespaces

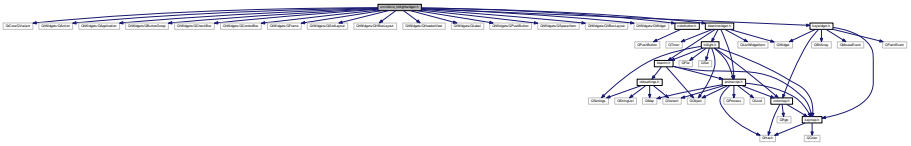
- Ui

13.165 src/ckb/ui_kblightwidget.h File Reference

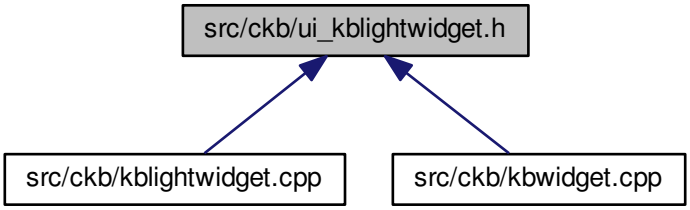
```
#include <QtCore/QVariant>
```

```
#include <QtWidgets/QAction>
#include <QtWidgets/QApplication>
#include <QtWidgets/QButtonGroup>
#include <QtWidgets/QCheckBox>
#include <QtWidgets/QComboBox>
#include <QtWidgets/QFrame>
#include <QtWidgets/QGridLayout>
#include <QtWidgets/QHBoxLayout>
#include <QtWidgets/QHeaderView>
#include <QtWidgets/QLabel>
#include <QtWidgets/QPushButton>
#include <QtWidgets/QSpacerItem>
#include <QtWidgets/QVBoxLayout>
#include <QtWidgets/QWidget>
#include "colorbutton.h"
#include "kbanimwidget.h"
#include "keywidget.h"
```

Include dependency graph for ui_kblightwidget.h:



This graph shows which files directly or indirectly include this file:



Data Structures

- class Ui_KbLightWidget
- class Ui::KbLightWidget

Namespaces

- Ui

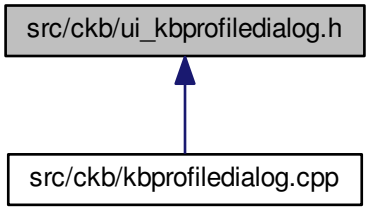
13.166 src/ckb/ui_kbprofiledialog.h File Reference

```
#include <QtCore/QVariant>
```

```
#include <QtWidgets/QAction>
#include <QtWidgets/QApplication>
#include <QtWidgets/QButtonGroup>
#include <QtWidgets/QDialog>
#include <QtWidgets/QDialogButtonBox>
#include <QtWidgets/QGridLayout>
#include <QtWidgets/QHeaderView>
#include <QtWidgets/QSpacerItem>
#include "rlistwidget.h"
```

Include dependency graph for ui_kbprofiledialog.h:

This graph shows which files directly or indirectly include this file:



Data Structures

- class Ui_KbProfileDialog
- class Ui::KbProfileDialog

Namespaces

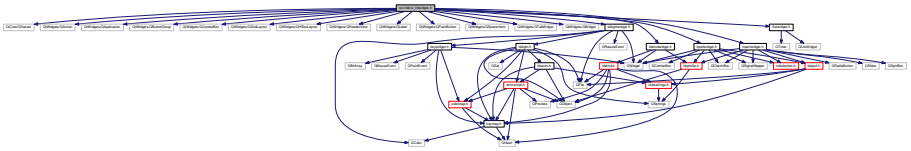
- Ui

13.167 src/ckb/ui_kbwidget.h File Reference

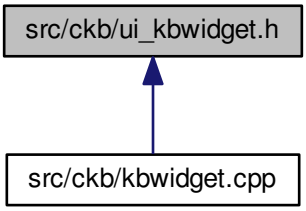
```
#include <QtCore/QVariant>
```

```
#include <QtWidgets/QAction>
#include <QtWidgets/QApplication>
#include <QtWidgets/QButtonGroup>
#include <QtWidgets/QComboBox>
#include <QtWidgets/QGridLayout>
#include <QtWidgets/QHBoxLayout>
#include <QtWidgets/QHeaderView>
#include <QtWidgets/QLabel>
#include <QtWidgets/QPushButton>
#include <QtWidgets/QSpacerItem>
#include <QtWidgets/QTabWidget>
#include <QtWidgets/QWidget>
#include <kblightwidget.h>
#include "kbbindwidget.h"
#include "kperfwidget.h"
#include "mperfwidget.h"
#include "rlistwidget.h"
```

Include dependency graph for ui_kbwidget.h:



This graph shows which files directly or indirectly include this file:



Data Structures

- class Ui_KbWidget
- class Ui::KbWidget

Namespaces

- Ui

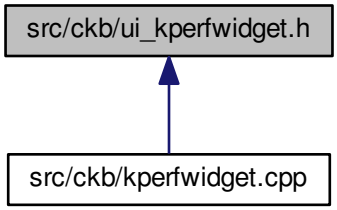
13.168 src/ckb/ui_kperfwidget.h File Reference

```
#include <QtCore/QVariant>
```

```
#include <QtWidgets/QAction>
#include <QtWidgets/QApplication>
#include <QtWidgets/QButtonGroup>
#include <QtWidgets/QCheckBox>
#include <QtWidgets/QComboBox>
#include <QtWidgets/QFrame>
#include <QtWidgets/QGridLayout>
#include <QtWidgets/QHBoxLayout>
#include <QtWidgets/QHeaderView>
#include <QtWidgets/QLabel>
#include <QtWidgets/QPushButton>
#include <QtWidgets/QSpacerItem>
#include <QtWidgets/QSpinBox>
#include <QtWidgets/QWidget>
#include "colorbutton.h"
```

Include dependency graph for ui_kperfwidget.h:

This graph shows which files directly or indirectly include this file:



Data Structures

- class Ui_KPerfWidget
- class Ui::KPerfWidget

Namespaces

- Ui

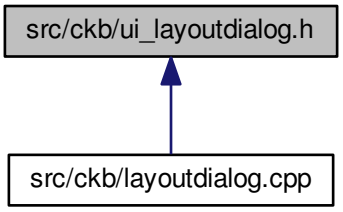
13.169 src/ckb/ui_layoutdialog.h File Reference

```
#include <QtCore/QVariant>
```

```
#include <QtWidgets/QAction>
#include <QtWidgets/QApplication>
#include <QtWidgets/QButtonGroup>
#include <QtWidgets/QComboBox>
#include <QtWidgets/QDialog>
#include <QtWidgets/QDialogButtonBox>
#include <QtWidgets/QGridLayout>
#include <QtWidgets/QHeaderView>
#include <QtWidgets/QLabel>
#include <QtWidgets/QSpacerItem>
```

Include dependency graph for ui_layoutdialog.h:

This graph shows which files directly or indirectly include this file:



Data Structures

- class Ui_LayoutDialog
- class Ui::LayoutDialog

Namespaces

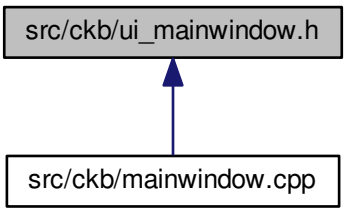
- Ui

13.170 src/ckb/ui_mainwindow.h File Reference

```
#include <QtCore/QVariant>
#include <QtWidgets/QAction>
#include <QtWidgets/QApplication>
#include <QtWidgets/QButtonGroup>
#include <QtWidgets/QGridLayout>
#include <QtWidgets/QHeaderView>
#include <QtWidgets/QMainWindow>
#include <QtWidgets/QTabWidget>
#include <QtWidgets/QWidget>
```

Include dependency graph for ui_mainwindow.h:

This graph shows which files directly or indirectly include this file:



Data Structures

- class Ui_MainWindow
- class Ui::MainWindow

Namespaces

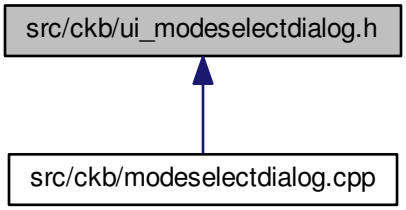
- Ui

13.171 src/ckb/ui_modeselectdialog.h File Reference

```
#include <QtCore/QVariant>
#include <QtWidgets/QAction>
#include <QtWidgets/QApplication>
#include <QtWidgets/QButtonGroup>
#include <QtWidgets/QDialog>
#include <QtWidgets/QDialogButtonBox>
#include <QtWidgets/QGridLayout>
#include <QtWidgets/QHeaderView>
#include <QtWidgets/QLabel>
#include <QtWidgets/QListWidget>
#include <QtWidgets/QPushButton>
```

Include dependency graph for ui_modeselectdialog.h:

This graph shows which files directly or indirectly include this file:



Data Structures

- class Ui_ModeSelectDialog
- class Ui::ModeSelectDialog

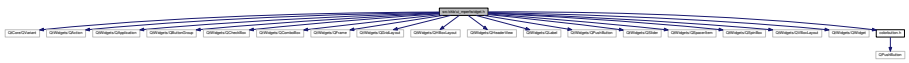
Namespaces

- Ui

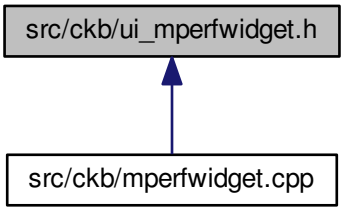
13.172 src/ckb/ui_mperfwidget.h File Reference

```
#include <QtCore/QVariant>
#include <QtWidgets/QAction>
#include <QtWidgets/QApplication>
#include <QtWidgets/QButtonGroup>
#include <QtWidgets/QCheckBox>
#include <QtWidgets/QComboBox>
#include <QtWidgets/QFrame>
#include <QtWidgets/QGridLayout>
#include <QtWidgets/QHBoxLayout>
#include <QtWidgets/QHeaderView>
#include <QtWidgets/QLabel>
#include <QtWidgets/QPushButton>
#include <QtWidgets/QSlider>
#include <QtWidgets/QSpacerItem>
#include <QtWidgets/QSpinBox>
#include <QtWidgets/QVBoxLayout>
#include <QtWidgets/QWidget>
#include "colorbutton.h"
```

Include dependency graph for ui_mperfwidget.h:



This graph shows which files directly or indirectly include this file:



Data Structures

- class Ui_MPerfWidget
- class Ui::MPerfWidget

Namespaces

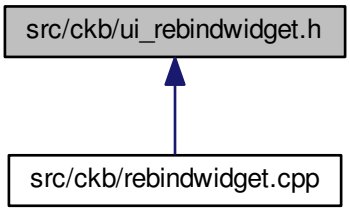
- Ui

13.173 src/ckb/ui_rebindwidget.h File Reference

```
#include <QtCore/QVariant>
#include <QtWidgets/QAction>
#include <QtWidgets/QApplication>
#include <QtWidgets/QButtonGroup>
#include <QtWidgets/QCheckBox>
#include <QtWidgets/QComboBox>
#include <QtWidgets/QGridLayout>
#include <QtWidgets/QGroupBox>
#include <QtWidgets/QHBoxLayout>
#include <QtWidgets/QHeaderView>
#include <QtWidgets/QLabel>
#include <QtWidgets/QLineEdit>
#include <QtWidgets/QPlainTextEdit>
#include <QtWidgets/QPushButton>
#include <QtWidgets/QRadioButton>
#include <QtWidgets/QSpacerItem>
#include <QtWidgets/QSpinBox>
#include <QtWidgets/QTabWidget>
#include <QtWidgets/QVBoxLayout>
#include <QtWidgets/QWidget>
```

Include dependency graph for ui_rebindwidget.h:

This graph shows which files directly or indirectly include this file:



Data Structures

- class Ui_RebindWidget
- class Ui::RebindWidget

Namespaces

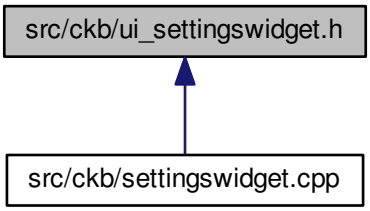
- Ui

13.174 src/ckb/ui_settingswidget.h File Reference

```
#include <QtCore/QVariant>
#include <QtWidgets/QAction>
#include <QtWidgets/QApplication>
#include <QtWidgets/QButtonGroup>
#include <QtWidgets/QCheckBox>
#include <QtWidgets/QComboBox>
#include <QtWidgets/QFrame>
#include <QtWidgets/QGridLayout>
#include <QtWidgets/QHBoxLayout>
#include <QtWidgets/QHeaderView>
#include <QtWidgets/QLabel>
#include <QtWidgets/QPushButton>
#include <QtWidgets/QSpacerItem>
#include <QtWidgets/QWidget>
```

Include dependency graph for ui_settingswidget.h:

This graph shows which files directly or indirectly include this file:



Data Structures

- class Ui_SettingsWidget
- class Ui::SettingsWidget

Namespaces

- Ui

Index

~AnimAddDialog
AnimAddDialog, 43
~AnimScript
AnimScript, 57
~AnimSettingDialog
AnimSettingDialog, 105
~CkbSettings
CkbSettings, 125
~CkbSettingsWriter
CkbSettingsWriter, 138
~ColorMap
ColorMap, 150
~ExtraSettingsWidget
ExtraSettingsWidget, 173
~FwUpgradeDialog
FwUpgradeDialog, 183
~GradientDialog
GradientDialog, 213
~KPerfWidget
KPerfWidget, 706
~Kb
Kb, 261
~KbAnimWidget
KbAnimWidget, 346
~KbBindWidget
KbBindWidget, 400
~KbFirmware
KbFirmware, 412
~KbLight
KbLight, 425
~KbLightWidget
KbLightWidget, 461
~KbMode
KbMode, 489
~KbProfileDialog
KbProfileDialog, 553
~KbWidget
KbWidget, 569
~KeyAction
KeyAction, 605
~LayoutDialog
LayoutDialog, 719
~MPerfWidget
MPerfWidget, 781
~MacroReader
MacroReader, 723
~MainWindow
MainWindow, 737
~ModeSelectDialog
ModeSelectDialog, 753
~QuaGzipFile
QuaGzipFile, 816
~QuaZIODevice
QuaZIODevice, 822
~QuaZip
QuaZip, 831
~QuaZipDir
QuaZipDir, 862
~QuaZipFile
QuaZipFile, 882
~QuaZipFilePrivate
QuaZipFilePrivate, 913
~RebindWidget
RebindWidget, 970
~SGroup
SGroup, 1032
~SettingsWidget
SettingsWidget, 1010
_LAYOUT_MAX
KeyMap, 633
_MODEL_MAX
KeyMap, 634
_FILE_OFFSET_BIT
ioapi.h, 1341
_LARGEFILE64_SOURCE
ioapi.h, 1342
_Z_OF
ioapi.h, 1344
unzip.c, 1392
unzip.h, 1446, 1447
zip.c, 1465
zip.h, 1518, 1519
__USE_FILE_OFFSET64
ioapi.h, 1341
__USE_LARGEFILE64
ioapi.h, 1341
_alpha
ColorButton, 148
GradientButton, 201
_angleSnap
KbPerf, 530
_anim
AnimSettingDialog, 116
_animList
KbLight, 448
_animMap
KbLight, 448
_backing

CkbSettingsWriter, 139
_bigIcons
ColorButton, 148
_bind
KbBind, 390
KbMode, 498
_bindMap
KeyWidget, 677
_checkUpdates
KbFirmware, 412
_color
ColorButton, 148
_colorBuffer
AnimScript, 81
_colorMap
KbLight, 449
KeyWidget, 677
_colors
AnimScript, 81
ColorMap, 157
GradientDialogWidget, 237
_count
ColorMap, 157
_curDpi
KbPerf, 505
_current
GradientDialogWidget, 237
_currentMode
Kb, 302
KbProfile, 550
_currentProfile
Kb, 302
_daemonVersion
KbManager, 483
_delay
Kb, 302
_devParent
KbBind, 390
_devices
KbManager, 483
_dimming
KbLight, 449
_displayColorMap
KeyWidget, 678
_dither
Kb, 302
_dpiIndicator
KbPerf, 530
_eventTimer
KbManager, 483
_fileForBoard
KbFirmware, 413
_frameRate
Kb, 302
_friendlyName
Key, 599
_globalRemap
KbBind, 390
_globalSettings
ckbsettings.cpp, 1242
_guiVersion
KbManager, 483
_guid
KbAnim, 331
_hwProfile
Kb, 302
_iOpacity
KbPerf, 530
_id
KbMode, 498
KbProfile, 550
_indicatorList
KbLight, 449
_indicatorMap
KbLight, 449
_indicators
KeyWidget, 678
_info
AnimScript, 81
_isActive
KbAnim, 331
_isActiveKp
KbAnim, 331
_kbManager
KbManager, 484
_keyMap
KbProfile, 550
_keyNames
ColorMap, 157
_keys
AnimAddDialog, 46
AnimScript, 81
KbAnim, 331
KeyMap, 656
_lastFrameAnimMap
KbLight, 449
_lastFrameDimming
KbLight, 449
_lastFrameIndicatorMap
KbLight, 449
_latestForBoard
KbFirmware, 414
_layout
Kb, 303
_liftHeight
KbPerf, 531
_light
AnimAddDialog, 46
KbMode, 498
_map
AnimScript, 81
KbAnim, 331
KbBind, 390
KbLight, 449
_mapCount
ColorMap, 157

_mode
KbAnim, 331
_modeList
ModeSelectDialog, 755
_model
Kb, 303
_modes
KbProfile, 550
_monochrome
KeyWidget, 678
_mouseAccel
Kb, 303
_name
KbAnim, 331
KbMode, 498
KbProfile, 550
_needsMapRefresh
KbLight, 449
_needsSave
Kb, 303
KbAnim, 331
KbBind, 390
KbLight, 449
KbMode, 498
KbPerf, 531
KbProfile, 551
_needsUpdate
KbBind, 391
KbPerf, 531
_opacity
KbAnim, 332
_paramValues
AnimScript, 81
_parameters
KbAnim, 332
_patch
keymap.cpp, 1291
_path
AnimScript, 81
_perf
KbMode, 498
_presetValues
AnimScript, 82
_presets
AnimScript, 81
_previewAnim
KbLight, 450
_profiles
Kb, 303
_qColorMap
KbLight, 450
_removals
CkbSettingsWriter, 139
_rgbMode
KeyWidget, 678
_scanTimer
KbManager, 484
_script
KbAnim, 332
_scriptGuid
KbAnim, 332
_scriptName
KbAnim, 332
_scrollSpeed
Kb, 303
_setLabel
ColorButton, 148
_settings
SGroup, 1032
_shareDimming
kblight.cpp, 1272
_start
KbLight, 450
_stops
GradientButton, 201
GradientDialogWidget, 237
_storageName
Key, 599
_tempParameters
KbAnim, 332
_updates
CkbSettingsWriter, 139
_value
KeyAction, 627
_winLock
KbBind, 391
_xyLink
MPerfWidget, 791

ADD
KeyWidget, 660
ALWAYS_OFF
KPerfWidget, 704
ALWAYS_ON
KPerfWidget, 704
aCopyLabel
Ui::KbAnimWidget, 339
Ui_KbAnimWidget, 1081
ALLOC
unzip.c, 1390
zip.c, 1462
aMissingLabel
Ui::KbAnimWidget, 339
Ui_KbAnimWidget, 1081
aMissingPage
Ui::KbAnimWidget, 339
Ui_KbAnimWidget, 1081
ANIM_KRSTOP
keyaction.cpp, 1283
ANIM_ONCE
keyaction.cpp, 1283
ANSI_ENTER_H
keymap.cpp, 1286
ANSI_ENTER_W
keymap.cpp, 1287
ANSI_ENTER_X
keymap.cpp, 1287

ANSI_ENTER_Y
keymap.cpp, 1287
ANSI_LSHIFT_W
keymap.cpp, 1287
ANSI_LSHIFT_X
keymap.cpp, 1287
aNameLabel
Ui::KbAnimWidget, 339
Ui_KbAnimWidget, 1081
APPEND_STATUS_CREATE
zip.h, 1516
aSnapBox
Ui::MPerfWidget, 767
Ui_MPerfWidget, 1156
AUTO_CHECK_TIME
kbfirmware.cpp, 1268
aVerLabel
Ui::KbAnimWidget, 340
Ui_KbAnimWidget, 1082
action
KbBind, 365
actionAbout
Ui::MainWindow, 731
Ui_MainWindow, 1140
actionButton
Ui::FwUpgradeDialog, 193
Ui_FwUpgradeDialog, 1066
actionExit
Ui::MainWindow, 731
Ui_MainWindow, 1140
actionTab
Ui::RebindWidget, 950
Ui_RebindWidget, 1180
activeDevices
kb.cpp, 1259
activeLights
kblight.cpp, 1272
Add
KbAnim, 309
add_data_in_datablock
zip.c, 1465
addAnim
KbAnimWidget, 346
KbLight, 425
addCurrentFileToDirectoryMap
QuaZipPrivate, 927
addDevice
MainWindow, 737
addNewModeItem
KbWidget, 569
addNewProfileItem
KbProfileDialog, 553
addPreset
GradientDialog, 213
adjustDisplay
KeyAction, 605
advance
AnimScript, 58
alloc
ColorMap, 151
allocate_new_datablock
zip.c, 1466
allowAlpha
ColorButton, 142, 143
altBox
Ui::SettingsWidget, 1025
Ui_SettingsWidget, 1202
altLabel
Ui::SettingsWidget, 1025
Ui_SettingsWidget, 1202
angleDialChanged
AnimSettingDialog, 105
angleDialMapper
AnimSettingDialog, 116
angleFlip
animsettingdialog.cpp, 1218
angleSnap
KbPerf, 505, 506
angleSpinnerChanged
AnimSettingDialog, 106
angleSpinnerMapper
AnimSettingDialog, 116
angleSpinners
AnimSettingDialog, 116
AnimScript::Param, 56, 1216
animAction
KeyAction, 606
AnimAddDialog, 41
~AnimAddDialog, 43
_keys, 46
_light, 46
AnimAddDialog, 42
AnimAddDialog, 42
chosenPreset, 43
chosenScript, 44
on_animBox_activated, 44
on_presetBox_activated, 45
on_previewBox_clicked, 46
scripts, 46
showPreview, 46
ui, 47
animBox
Ui::AnimAddDialog, 50
Ui::RebindWidget, 950
Ui_AnimAddDialog, 1036
Ui_RebindWidget, 1180
animButton
Ui::KbLightWidget, 456
Ui::RebindWidget, 950
Ui_KbLightWidget, 1096
Ui_RebindWidget, 1181
animChanged
KbAnimWidget, 347
animCountLabel
Ui::ExtraSettingsWidget, 165
Ui_ExtraSettingsWidget, 1057

animInfo
KeyAction, 607
animKeypress
KbLight, 426
animKrBox
Ui::RebindWidget, 951
Ui_RebindWidget, 1181
AnimList
KbLight, 424
animList
KbLight, 427
Ui::KbAnimWidget, 339
Ui_KbAnimWidget, 1081
animName
Ui::AnimSettingDialog, 91
Ui_AnimSettingDialog, 1044
animOnceBox
Ui::RebindWidget, 951
Ui_RebindWidget, 1181
animPage
Ui::KbAnimWidget, 339
Ui_KbAnimWidget, 1082
animPathLabel
Ui::ExtraSettingsWidget, 165
Ui_ExtraSettingsWidget, 1058
animScanButton
Ui::ExtraSettingsWidget, 165
Ui_ExtraSettingsWidget, 1058
AnimScript, 52
~AnimScript, 57
_colorBuffer, 81
_colors, 81
_info, 81
_keys, 81
_map, 81
_paramValues, 81
_path, 81
_presetValues, 82
_presets, 81
advance, 58
AnimScript, 57
AnimScript, 57
author, 58
begin, 59
colors, 60
copy, 60
copyright, 61
count, 61
description, 62
durationMsec, 82
end, 62
firstFrame, 82
frame, 63
guid, 64
hasFrame, 64
hasKeypress, 65
hasParam, 65
inFrame, 82
init, 66
initialized, 82
KP_NAME, 82
KP_NONE, 82
KP_POSITION, 82
keypress, 67
lastFrame, 82
license, 68
list, 68
load, 69
minX, 83
minY, 83
name, 72
param, 72
paramIterator, 74
parameters, 73
path, 74
preset, 74
PresetValue, 57
presets, 75
printParams, 75
process, 83
readAnyFrame, 83
readFrame, 83
readProcess, 76
repeatMsec, 83
retrigger, 77
scan, 78
scripts, 83
setDuration, 78
stop, 79
stopped, 83
version, 80
year, 80
AnimSettingDialog, 97
~AnimSettingDialog, 105
_anim, 116
angleDialChanged, 105
angleDialMapper, 116
angleSpinnerChanged, 106
angleSpinnerMapper, 116
angleSpinners, 116
AnimSettingDialog, 100
AnimSettingDialog, 100
hLine, 107
hasRepeat, 117
kpStopCheck, 117
lastDuration, 117
name, 108
newDuration, 108
on_delayBox_valueChanged, 109
on_kpDelayBox_valueChanged, 110
on_kpReleaseBox_stateChanged, 110
on_kpRepeatBox_valueChanged, 111
on_repeatBox_valueChanged, 111
settingWidgets, 117
stopCheck, 117
ui, 117

updateMapper, 117
updateParam, 112
updateStops, 115
animTab
Ui::RebindWidget, 951
Ui_RebindWidget, 1181
animWidget
Ui::KbLightWidget, 456
Ui_KbLightWidget, 1097
animation
KeyWidget, 678
animations
KbAnimWidget, 360
animscript.cpp
ONE_DAY, 1215
urlParam, 1215
animsettingdialog.cpp
angleFlip, 1218
appShare
main.cpp, 1317
mainwindow.cpp, 1322
append
KbProfile, 538
appendProfile
Kb, 264
applyButton
Ui::RebindWidget, 951
Ui_RebindWidget, 1181
applyChanges
RebindWidget, 970
applyIndicators
KbPerf, 506
atEnd
QuaZipFile, 882
author
AnimScript, 58
autoClose
QuaZipPrivate, 931
autoFWBox
Ui::SettingsWidget, 1025
Ui_SettingsWidget, 1202
AutoRun, 118
available, 118
disable, 119
enable, 119
isEnabled, 120
once, 121
autoSave
Kb, 264
autorun.cpp
settingPath, 1220
available
AutoRun, 118

BOTH
KPerfWidget, 704
BUFREADCOMMENT
unzip.c, 1391
zip.c, 1463
BUILD.md, 1213
backing
CkbSettings, 136
base
KbLight, 427
baseDpiIdx
KbPerf, 507, 508
begin
AnimScript, 59
beginGroup
CkbSettings, 125
bgButton
Ui::KbLightWidget, 456
Ui_KbLightWidget, 1097
bigIcons
ColorButton, 143, 144
bind
KbBindWidget, 408
KbMode, 489
KbPerf, 508
RebindWidget, 998
bindAction
KbBind, 366
BindMap
KeyWidget, 659
bindMap
KeyWidget, 660
bindTab
Ui::KbWidget, 593
Ui_KbWidget, 1109
bindWidget
Ui::KbWidget, 593
Ui_KbWidget, 1109
blend
KbAnim, 312
blendAdd
kbanim.cpp, 1261
blendBox
Ui::KbAnimWidget, 340
Ui_KbAnimWidget, 1082
blendDivide
kbanim.cpp, 1261
blendFunc
kbanim.cpp, 1261
blendMultiply
kbanim.cpp, 1261
blendNormal
kbanim.cpp, 1261
blendSubtract
kbanim.cpp, 1262
blob
FwUpgradeDialog, 188
boxXChanged
MPerfWidget, 781
boxXMapper
MPerfWidget, 791
boxYChanged
MPerfWidget, 782

boxYMapper
MPerfWidget, 791
brightnessBox
Ui::ExtraSettingsWidget, 165
Ui::KbLightWidget, 456
Ui_ExtraSettingsWidget, 1058
Ui_KbLightWidget, 1097
btnClearMacro
Ui::RebindWidget, 951
Ui_RebindWidget, 1181
btnStartMacro
Ui::RebindWidget, 951
Ui_RebindWidget, 1181
btnStopMacro
Ui::RebindWidget, 951
Ui_RebindWidget, 1181
builtIn
GradientDialog::Preset, 794
buttonBox
Ui::AnimAddDialog, 51
Ui::AnimSettingDialog, 91
Ui::ExtraSettingsWidget, 165
Ui::GradientDialog, 206
Ui::KbProfileDialog, 564
Ui::LayoutDialog, 716
Ui::ModeSelectDialog, 750
Ui_AnimAddDialog, 1036
Ui_AnimSettingDialog, 1044
Ui_ExtraSettingsWidget, 1058
Ui_GradientDialog, 1073
Ui_KbProfileDialog, 1102
Ui_LayoutDialog, 1136
Ui_ModeSelectDialog, 1144
buttonMapper1
MPerfWidget, 791
buttonMapper2
MPerfWidget, 791
byPosition
KeyMap, 635
bytesAvailable
QuaZipFile, 883

CAPS
KbPerf, 503
CENTRALHEADERMAGIC
zip.c, 1463
CKB_CONTAINER
ckb-anim.h, 1226
CKB_COPYRIGHT
ckb-anim.h, 1226
CKB_DESCRIPTION
ckb-anim.h, 1226
CKB_DOWN
ckb-anim.h, 1226
CKB_GRAD_MAX
ckb-anim.h, 1226
CKB_GUID
ckb-anim.h, 1227
CKB_KEYCLEAR
ckb-anim.h, 1227
CKB_KEYNAME_MAX
ckb-anim.h, 1227
CKB_KP_NAME
ckb-anim.h, 1227
CKB_KP_NONE
ckb-anim.h, 1227
CKB_KP_POSITION
ckb-anim.h, 1227
CKB_KPMODE
ckb-anim.h, 1227
CKB_LICENSE
ckb-anim.h, 1227
CKB_LIVEPARAMS
ckb-anim.h, 1227
CKB_MAX_WORD
ckb-anim.h, 1227
CKB_NAME
ckb-anim.h, 1227
CKB_PARAM
ckb-anim.h, 1228
CKB_PARAM_AGRADIENT
ckb-anim.h, 1228
CKB_PARAM_ANGLE
ckb-anim.h, 1228
CKB_PARAM_ARGB
ckb-anim.h, 1228
CKB_PARAM_BOOL
ckb-anim.h, 1228
CKB_PARAM_DOUBLE
ckb-anim.h, 1228
CKB_PARAM_GRADIENT
ckb-anim.h, 1228
CKB_PARAM_LABEL
ckb-anim.h, 1228
CKB_PARAM_LONG
ckb-anim.h, 1228
CKB_PARAM_RGB
ckb-anim.h, 1228
CKB_PARAM_STRING
ckb-anim.h, 1229
CKB_PARSE_AGRADIENT
ckb-anim.h, 1229
CKB_PARSE_ANGLE
ckb-anim.h, 1229
CKB_PARSE_ARGB
ckb-anim.h, 1229
CKB_PARSE_BOOL
ckb-anim.h, 1229
CKB_PARSE_DOUBLE
ckb-anim.h, 1229
CKB_PARSE_GRADIENT
ckb-anim.h, 1229
CKB_PARSE_LONG
ckb-anim.h, 1229
CKB_PARSE_RGB
ckb-anim.h, 1229
CKB_PARSE_STRING

ckb-anim.h, 1229
CKB_PREEMPT
ckb-anim.h, 1230
CKB_PRESET_END
ckb-anim.h, 1230
CKB_PRESET_PARAM
ckb-anim.h, 1230
CKB_PRESET_START
ckb-anim.h, 1230
CKB_REAL_ANGLE
ckb-anim.h, 1230
CKB_REPEAT
ckb-anim.h, 1230
CKB_TIME_ABSOLUTE
ckb-anim.h, 1230
CKB_TIME_DURATION
ckb-anim.h, 1230
CKB_TIMEMODE
ckb-anim.h, 1230
CKB_UP
ckb-anim.h, 1230
CKB_VERSION
ckb-anim.h, 1230
CRC32
crypt.h, 1335
cacheWritesInProgress
ckbsettings.cpp, 1241
ckbsettingswriter.cpp, 1245
calculate
QuaAdler32, 805
QuaChecksum32, 808
QuaCrc32, 811
call_zopen64
qioapi.cpp, 1349
call_zseek64
qioapi.cpp, 1349
call_ztell64
qioapi.cpp, 1349
cancelButton
Ui::FwUpgradeDialog, 193
Ui::RebindWidget, 951
Ui_FwUpgradeDialog, 1067
Ui_RebindWidget, 1181
capsBox
Ui::KPerfWidget, 691
Ui::SettingsWidget, 1025
Ui_KPerfWidget, 1124
Ui_SettingsWidget, 1202
capsColorOff
Ui::KPerfWidget, 691
Ui_KPerfWidget, 1124
capsColorOn
Ui::KPerfWidget, 691
Ui_KPerfWidget, 1124
capsLabel
Ui::SettingsWidget, 1026
Ui_SettingsWidget, 1202
CaseSensitivity
QuaZip, 830
caseSensitivity
QuaZipDir, 863
QuaZipFilePrivate, 914
catchUp
KbAnim, 313
cd
QuaZipDir, 863
cdUp
QuaZipDir, 865
centralWidget
Ui::MainWindow, 731
Ui_MainWindow, 1140
change
RListWidget, 1003
changeAnim
KbLightWidget, 462
changeAnimKeys
KbLightWidget, 462
changeColor
KbLightWidget, 463
checkFwUpdates
MainWindow, 738
checkUpdates
KbFirmware, 415
checksum
QuaAdler32, 806
QuaCrc32, 812
childGroups
CkbSettings, 125
childKeys
CkbSettings, 126
chosenPreset
AnimAddDialog, 43
chosenScript
AnimAddDialog, 44
ckb-anim.h
CKB_CONTAINER, 1226
CKB_COPYRIGHT, 1226
CKB_DESCRIPTION, 1226
CKB_DOWN, 1226
CKB_GRAD_MAX, 1226
CKB_GUID, 1227
CKB_KEYCLEAR, 1227
CKB_KEYNAME_MAX, 1227
CKB_KP_NAME, 1227
CKB_KP_NONE, 1227
CKB_KP_POSITION, 1227
CKB_KPMODE, 1227
CKB_LICENSE, 1227
CKB_LIVEPARAMS, 1227
CKB_MAX_WORD, 1227
CKB_NAME, 1227
CKB_PARAM, 1228
CKB_PARAM_AGRADIENT, 1228
CKB_PARAM_ANGLE, 1228
CKB_PARAM_ARGB, 1228
CKB_PARAM_BOOL, 1228

CKB_PARAM_DOUBLE, 1228
CKB_PARAM_GRADIENT, 1228
CKB_PARAM_LABEL, 1228
CKB_PARAM_LONG, 1228
CKB_PARAM_RGB, 1228
CKB_PARAM_STRING, 1229
CKB_PARSE_AGRADIENT, 1229
CKB_PARSE_ANGLE, 1229
CKB_PARSE_ARGB, 1229
CKB_PARSE_BOOL, 1229
CKB_PARSE_DOUBLE, 1229
CKB_PARSE_GRADIENT, 1229
CKB_PARSE_LONG, 1229
CKB_PARSE_RGB, 1229
CKB_PARSE_STRING, 1229
CKB_PREEMPT, 1230
CKB_PRESET_END, 1230
CKB_PRESET_PARAM, 1230
CKB_PRESET_START, 1230
CKB_REAL_ANGLE, 1230
CKB_REPEAT, 1230
CKB_TIME_ABSOLUTE, 1230
CKB_TIME_DURATION, 1230
CKB_TIMEMODE, 1230
CKB_UP, 1230
CKB_VERSION, 1230
ckb_alpha_blend, 1231
ckb_frame, 1231
ckb_getline, 1231
ckb_grad_color, 1232
ckb_info, 1233
ckb_init, 1233
ckb_keypress, 1233
ckb_parameter, 1233
ckb_read_params, 1234
ckb_scan_grad, 1235
ckb_start, 1235
ckb_time, 1236
FALSE, 1230
M_PI, 1231
main, 1236
printurl, 1238
TRUE, 1231
urldecode, 1239
ckb_gradient, 1225
ckb_key, 1223
ckb_runctx, 1224
ckb_alpha_blend
ckb-anim.h, 1231
ckb_frame
ckb-anim.h, 1231
ckb_getline
ckb-anim.h, 1231
ckb_grad_color
ckb-anim.h, 1232
ckb_info
ckb-anim.h, 1233
ckb_init
ckb-anim.h, 1233
ckb_keypress
ckb-anim.h, 1233
ckb_parameter
ckb-anim.h, 1233
ckb_read_params
ckb-anim.h, 1234
ckb_scan_grad
ckb-anim.h, 1235
ckb_start
ckb-anim.h, 1235
ckb_time
ckb-anim.h, 1236
ckbDaemonVersion
KbManager, 475
ckbDaemonVersionF
KbManager, 475
ckbGuiVersion
KbManager, 476
ckbGuiVersionF
KbManager, 476
CkbSettings, 122
~CkbSettings, 125
backing, 136
beginGroup, 125
childGroups, 125
childKeys, 126
CkbSettings, 124, 125
CkbSettings, 124, 125
cleanUp, 127
contains, 128
containsGroup, 128
endGroup, 129
get, 129
groups, 136
isBusy, 130
pwd, 131, 132
remove, 133
removeCache, 137
set, 133
setValue, 134
value, 135
writeCache, 137
CkbSettingsWriter, 137
~CkbSettingsWriter, 138
_backing, 139
_removals, 139
_updates, 139
CkbSettingsWriter, 138
CkbSettingsWriter, 138
run, 139
ckbVersion
KbFirmware::FW, 180
ckbsettings.cpp
_globalSettings, 1242
cacheWritesInProgress, 1241
globalCache, 1242
globalSettings, 1241

globalThread, 1242
lockMutex, 1240
lockMutexCache, 1240
lockMutexStatic, 1241
lockMutexStatic2, 1241
settingsCacheMutex, 1242
settingsMutex, 1242
ckbsettingswriter.cpp
cacheWritesInProgress, 1245
lockMutex, 1244
lockMutexCache, 1244
lockMutexStatic, 1244
lockMutexStatic2, 1244
settingsCacheMutex, 1245
settingsMutex, 1245
cleanBlob
FwUpgradeDialog, 183
cleanUp
CkbSettings, 127
cleanup
MainWindow, 739
clear
ColorMap, 151
clearAnimation
KeyWidget, 661
clearDirectoryMap
QuaZipPrivate, 928
clearSelection
KbAnimWidget, 348
KeyWidget, 661
close
KbLight, 428
QuaGzipFile, 816
QuaZIODevice, 822
QuaZip, 832
QuaZipFile, 883
closeAction
MainWindow, 746
closeEvent
FwUpgradeDialog, 183
MainWindow, 739
cmd
Kb, 303
cmdpath
Kb, 303
color
ColorButton, 144, 145
KbLight, 429, 430
colorAt
GradientDialogWidget, 225
ColorButton, 139
_alpha, 148
_bigIcons, 148
_color, 148
_setLabel, 148
allowAlpha, 142, 143
bigIcons, 143, 144
color, 144, 145
ColorButton, 142
colorChanged, 146
ColorButton, 142
pickColor, 146
setLabel, 146, 147
updateImage, 147
colorChanged
ColorButton, 146
GradientDialog, 214
MPerfWidget, 783
colorClicked
MPerfWidget, 784
colorForName
ColorMap, 152, 153
colorLink
MPerfWidget, 792
ColorMap, 149
∼ColorMap, 150
_colors, 157
_count, 157
_keyNames, 157
_mapCount, 157
alloc, 151
clear, 151
colorForName, 152, 153
ColorMap, 150
ColorMap, 150
colors, 153, 154
count, 154
deAlloc, 154
init, 155
keyNames, 156
operator=, 156
operator==, 157
colorMap
KbLight, 430
KeyWidget, 661, 662
colormap.cpp
qs_strcmp, 1247
colormap.h
QColorMap, 1249
QColorMapIterator, 1249
QMutableColorMapIterator, 1249
colors
AnimScript, 60
ColorMap, 153, 154
comboBox
Ui::LayoutDialog, 716
Ui_LayoutDialog, 1136
CommandLineBackground
main.cpp, 1316
CommandLineClose
main.cpp, 1316
CommandLineError
main.cpp, 1316
CommandLineHelpRequested
main.cpp, 1316
CommandLineOK

main.cpp, 1316
CommandLineVersionRequested
main.cpp, 1316
CommandLineParseResults
main.cpp, 1316
comment
QuaZipFileInfo, 902
QuaZipFileInfo64, 909
QuaZipNewInfo, 923
QuaZipPrivate, 931
commentCodec
QuaZipPrivate, 931
commitParams
KbAnim, 315
compressDir
JlCompress, 239
compressFile
JlCompress, 240, 241
compressFiles
JlCompress, 242
compressSubDir
JlCompress, 244
compressedSize
QuaZipFileInfo, 902
QuaZipFileInfo64, 909
configLabel
mainwindow.cpp, 1322
Constants
QuaZip, 830
contains
CkbSettings, 128
KeyMap, 636
containsGroup
CkbSettings, 128
convertCaseSensitivity
QuaZip, 833
convertMacroBox
RebindWidget, 972
copy
AnimScript, 60
copyButton
Ui::KbBindWidget, 396
Ui::KPerfWidget, 691
Ui::MPerfWidget, 767
Ui_KbBindWidget, 1089
Ui_KPerfWidget, 1124
Ui_MPerfWidget, 1156
copyData
JlCompress.cpp, 1346
copyright
AnimScript, 61
count
AnimScript, 61
ColorMap, 154
KeyMap, 636
QuaZipDir, 865
crc
QuaZipFileInfo, 902
QuaZipFileInfo64, 909
QuaZipFilePrivate, 914
crypt.h
CRC32, 1335
decrypt_byte, 1336
init_keys, 1336
update_keys, 1337
zdecode, 1336
zencode, 1336
csDefault
QuaZip, 830
csInsensitive
QuaZip, 830
csSensitive
QuaZip, 830
csize
QuaZipFile, 884
ctrlBox
Ui::SettingsWidget, 1026
Ui_SettingsWidget, 1203
ctrlLabel
Ui::SettingsWidget, 1026
Ui_SettingsWidget, 1203
curLabel
Ui::FwUpgradeDialog, 193
Ui_FwUpgradeDialog, 1067
curfile64_info, 1461
current
GradientDialogWidget, 226
KbAnimWidget, 360
currentBind
Kb, 264
currentChanged
GradientDialog, 214
GradientDialogWidget, 226
currentData
RListWidget, 1006
currentLight
Kb, 265
currentMode
Kb, 266
KbProfile, 538, 539
KbWidget, 586
currentPerf
Kb, 266
currentPreset
GradientDialog, 221
currentProfile
Kb, 267
currentSelection
KbBindWidget, 408
KbLightWidget, 471

d
QuaGzipFile, 819
QuaZIODevice, 826
QuaZipDir, 876
DE
KeyMap, 633

DK
KeyMap, 633
DAEMON.md, 1213
DATA_ROLE
RListWidget, 1006
DEF_MEM_LEVEL
zip.h, 1517
DESCRIPTORHEADERMAGIC
zip.c, 1463
DIALOG_HEIGHT_MAX
fwupgradedialog.cpp, 1252
DIALOG_HEIGHT_MIN
fwupgradedialog.cpp, 1253
DIALOG_WIDTH
fwupgradedialog.cpp, 1253
DPI_BASE
mperfwidget.cpp, 1332
DPI_COUNT
KbPerf, 531
MPerfWidget, 792
DPI_CUST_IDX
rebindwidget.cpp, 1523
DPI_CUSTOM
KeyAction, 627
DPI_CYCLE_DOWN
KeyAction, 628
DPI_CYCLE_UP
KeyAction, 628
DPI_DOWN
KeyAction, 628
DPI_MAX
KbPerf, 531
MPerfWidget, 792
DPI_MIN
KbPerf, 531
MPerfWidget, 792
DPI_OFFSET
rebindwidget.cpp, 1523
DPI_SNIPER
KeyAction, 628
DPI_UP
KeyAction, 628
daemonDialogText
mainwindow.cpp, 1322
dataDescriptorWritingEnabled
QuaZipPrivate, 931
dataForBoard
KbFirmware, 415
dateTime
QuaZipFileInfo, 902
QuaZipFileInfo64, 909
QuaZipNewInfo, 923
deAlloc
ColorMap, 154
decrypt_byte
crypt.h, 1336
defaultAction
KbBind, 366, 367
KeyAction, 607
defaultFileNameCodec
QuaZipPrivate, 932
delayBox
Ui::AnimSettingDialog, 91
Ui::ExtraSettingsWidget, 166
Ui_AnimSettingDialog, 1045
Ui_ExtraSettingsWidget, 1058
delayButtons
Ui::RebindWidget, 951
Ui_RebindWidget, 1182
deleteButton
Ui::KbAnimWidget, 340
Ui_KbAnimWidget, 1082
deleteHw
Kb, 267
deletePrevious
Kb, 268
description
AnimScript, 62
devLabel
Ui::FwUpgradeDialog, 193
Ui_FwUpgradeDialog, 1067
devParent
KbBind, 367
devTab
Ui::KbWidget, 593
Ui_KbWidget, 1109
devUpdate
KbWidget, 570
device
KbProfileDialog, 559
KbWidget, 586
devices
KbManager, 477
devicesLabel
Ui::SettingsWidget, 1026
Ui_SettingsWidget, 1203
devpath
Kb, 303
kbmanager.cpp, 1275
mainwindow.cpp, 1322
settingswidget.cpp, 1632
didLoad
KbBind, 368
KbLight, 431
KbPerf, 509
didUpdateSelection
KbAnimWidget, 348
dimming
KbLight, 431, 432
dirName
QuaZipDir, 866
directoryCaseInsensitive
QuaZipPrivate, 932
directoryCaseSensitive
QuaZipPrivate, 932
disable

AutoRun, 119
disableAppNap
media_mac.m, 1327
diskNumberStart
QuaZipFileInfo, 902
QuaZipFileInfo64, 909
displayColorMap
KeyWidget, 662
dither
Kb, 268, 269
ditherBox
Ui::ExtraSettingsWidget, 166
Ui_ExtraSettingsWidget, 1058
Divide
KbAnim, 310
doUpdate
KbMode, 490
downloadFinished
KbFirmware, 416
downloaded
KbFirmware, 416
dpi
KbPerf, 509, 510
dpiAction
KeyAction, 608
dpiBaseIdx
KbPerf, 531
dpiBox
Ui::RebindWidget, 952
Ui_RebindWidget, 1182
dpiButton
Ui::RebindWidget, 952
Ui_RebindWidget, 1182
dpiClr
KbPerf, 531
dpiColor
KbPerf, 511, 512
dpiCurX
KbPerf, 531
dpiCurY
KbPerf, 532
dpiCustLabel
Ui::RebindWidget, 952
Ui_RebindWidget, 1182
dpiCustXBox
Ui::RebindWidget, 952
Ui_RebindWidget, 1182
dpiCustYBox
Ui::RebindWidget, 952
Ui_RebindWidget, 1182
dpiCycleDown
KbPerf, 512
dpiCycleUp
KbPerf, 513
dpiDown
KbPerf, 513
dpiEnabled
KbPerf, 514
dpiExp
mperfwidget.cpp, 1330
dpiIndicator
KbPerf, 515
dpiInfo
KeyAction, 609
dpiLog
mperfwidget.cpp, 1331
dpiOn
KbPerf, 532
dpiRound
mperfwidget.cpp, 1331
dpiUp
KbPerf, 515
dpiWarning
Ui::RebindWidget, 952
Ui_RebindWidget, 1182
dpiX
KbPerf, 532
dpiY
KbPerf, 532
dragged
RListWidget, 1006
drawInfo
KeyWidget, 662
driverName
KeyAction, 609
duplicateAnim
KbAnimWidget, 349
KbLight, 432
durationMsec
AnimScript, 82

ES
KeyMap, 633
EU
KeyMap, 633
EU_DVORAK
KeyMap, 633
eBox1
Ui::MPerfWidget, 767
Ui_MPerfWidget, 1157
eBox2
Ui::MPerfWidget, 768
Ui_MPerfWidget, 1157
eBox3
Ui::MPerfWidget, 768
Ui_MPerfWidget, 1157
eBox4
Ui::MPerfWidget, 768
Ui_MPerfWidget, 1157
eBox5
Ui::MPerfWidget, 768
Ui_MPerfWidget, 1157
ENDHEADERMAGIC
zip.c, 1463
ENUM_C
media.h, 1324
ENUM_END_C

media.h, 1324
EXTERN_C
media.h, 1324
effectiveParams
KbAnim, 315
enable
AutoRun, 119
enableChanged
MPerfWidget, 784
enableMapper
MPerfWidget, 792
end
AnimScript, 62
endGroup
CkbSettings, 129
enter
RListWidget, 1003
entryInfoList
QuaZipDir, 866, 867
entryInfoList64
QuaZipDir, 867, 868
entryList
QuaZipDir, 868, 869
evLoop
FwUpgradeDialog, 188
eventTimer
KbManager, 477
exec
FwUpgradeDialog, 184
exists
QuaZipDir, 869, 870
exitSuccess
FwUpgradeDialog, 188
externalAttr
QuaZipFileInfo, 902
QuaZipFileInfo64, 909
QuaZipNewInfo, 923
extra
QuaZipFileInfo, 902
QuaZipFileInfo64, 910
SettingsWidget, 1017
extraButton
Ui::SettingsWidget, 1026
Ui_SettingsWidget, 1203
extraGlobal
QuaZipNewInfo, 924
extraLocal
QuaZipNewInfo, 924
ExtraSettingsWidget, 170
~ExtraSettingsWidget, 173
ExtraSettingsWidget, 172
ExtraSettingsWidget, 172
on_animScanButton_clicked, 173
on_brightnessBox_clicked, 174
on_delayBox_clicked, 175
on_ditherBox_clicked, 175
on_fpsBox_valueChanged, 175
on_mAccelBox_clicked, 176
on_sAccelBox_clicked, 176
on_sSpeedBox_valueChanged, 176
on_trayBox_clicked, 177
pollUpdates, 177
ui, 178
extractDir
JlCompress, 247
extractFile
JlCompress, 249, 251
extractFiles
JlCompress, 252
extrasettingswidget.cpp
lastSharedDimming, 1249

FR
KeyMap, 633
FALSE
ckb-anim.h, 1230
FW
KbFirmware::FW, 179
features
Kb, 304
file_in_zip64_read_info_s, 1388
fileName
KbFirmware::FW, 180
QuaZipFilePrivate, 914
fileNameCodec
QuaZipPrivate, 932
filePath
QuaZipDir, 871
fill_qiodevice64_filefunc
qioapi.cpp, 1350
fill_qiodevice_filefunc
qioapi.cpp, 1351
fill_zlib_filefunc64_32_def_from_filefunc32
ioapi.h, 1344
qioapi.cpp, 1352
fillRect
GradientDialogWidget, 226
filter
QuaZipDir, 871
find
Kb, 269
KbProfile, 539
findAnim
KbLight, 433
findAnimIdx
KbLight, 434
firmware
Kb, 304
firstFrame
AnimScript, 82
flags
QuaZipFileInfo, 902
QuaZipFileInfo64, 910
flush
QuaGzipFile, 817
QuaZIODevice, 823
fnBox

Ui::RebindWidget, 952
Ui_RebindWidget, 1182
fnButton
Ui::RebindWidget, 952
Ui_RebindWidget, 1183
fnKeys
RebindWidget, 999
fps
KbManager, 477
fpsBox
Ui::ExtraSettingsWidget, 166
Ui_ExtraSettingsWidget, 1058
fpsWarnLabel
Ui::ExtraSettingsWidget, 166
Ui_ExtraSettingsWidget, 1058
frame
AnimScript, 63
frameDisplayed
KbLight, 434
frameRate
Kb, 270
frameUpdate
Kb, 271
KbLight, 434
free_datablock
zip.c, 1467
free_linkedlist
zip.c, 1467
friendlyActionName
KbBind, 368
friendlyName
KbBind, 369
Key, 598
KeyAction, 610
KeyMap, 637
fromName
KeyMap, 638
fromStorage
KeyMap, 639
fromString
GradientButton, 197
functions
kbanim.cpp, 1262
fwLabel
Ui::KbWidget, 593
Ui_KbWidget, 1109
fwTable
KbFirmware, 420
fwTableHash
KbFirmware, 420
fwUpdButton
Ui::KbWidget, 593
Ui_KbWidget, 1109
fwUpdLabel
Ui::KbWidget, 594
Ui_KbWidget, 1109
fwUpdLayout
Ui::KbWidget, 594
Ui_KbWidget, 1110
fwUpdPath
Kb, 304
fwUpdate
Kb, 272
fwUpdateFinished
FwUpgradeDialog, 185
Kb, 273
fwUpdateProgress
FwUpgradeDialog, 186
Kb, 273
FwUpgradeDialog, 180
~FwUpgradeDialog, 183
blob, 188
cleanBlob, 183
closeEvent, 183
evLoop, 188
exec, 184
exitSuccess, 188
fwUpdateFinished, 185
fwUpdateProgress, 186
FwUpgradeDialog, 182
FwUpgradeDialog, 182
kb, 188
on_actionButton_clicked, 186
on_cancelButton_clicked, 187
removeDev, 187
saveBlob, 187
savePath, 188
ui, 189
fwVersion
KbFirmware::FW, 180
fwupgradedialog.cpp
DIALOG_HEIGHT_MAX, 1252
DIALOG_HEIGHT_MIN, 1253
DIALOG_WIDTH, 1253
ids, 1253
verifyFw, 1252

GB
KeyMap, 633
GB_DVORAK
KeyMap, 633
GLAIVE
KeyMap, 634
GLAIVE_HEIGHT
keymap.cpp, 1287
GLAIVE_WIDTH
keymap.cpp, 1287
GUID
KbProfileDialog, 560
KbWidget, 587
get
CkbSettings, 129
getActualFileName
QuaZipFile, 885
getCaseSensitivity
QuaZipFile, 886
getComment

QuaZip, 834
getCommentCodec
QuaZip, 834
getCurrentFileInfo
QuaZip, 835, 836
getCurrentFileName
QuaZip, 837
getDefaultFileNameCodec
QuaZipPrivate, 929
getEntriesCount
QuaZip, 838
getFileInfo
QuaZipFile, 886, 887
getFileInfoList
QuaZip, 839
QuaZipPrivate, 929
getFileInfoList64
QuaZip, 839
getFileList
JlCompress, 253
getFileName
QuaGzipFile, 817
QuaZipFile, 888
getFileNameCodec
QuaZip, 840
getFileNameList
QuaZip, 840
getGradient
GradientDialog, 215
getIndicator
KbPerf, 516
getIoDevice
QuaZIODevice, 824
QuaZip, 841
getKeyMap
Kb, 274
getLayout
KeyMap, 639, 640
getMacroNumber
Kb, 274
KbBind, 369
getMacroPath
Kb, 274
KbBind, 370
getMap
keymap.cpp, 1292
getMode
QuaZip, 841
getModel
KeyMap, 641, 642
getMuteState
media.h, 1325
media_linux.cpp, 1326
media_mac.m, 1328
getNTFSTime
quazipfileinfo.cpp, 1377
getNTFSaTime
QuaZipFileInfo64, 905
getNTFScTime
QuaZipFileInfo64, 906
getNTFSmTime
QuaZipFileInfo64, 906
getPermissions
QuaZipFileInfo, 901
QuaZipFileInfo64, 907
getUnzFile
QuaZip, 842
getZip
QuaZipFile, 888
getZipError
QuaZip, 843
QuaZipFile, 888
getZipFile
QuaZip, 844
getZipName
QuaZip, 844
QuaZipFile, 889
GlaiveKeys
keymap.cpp, 1300
glaiveOverlay
keywidget.cpp, 1308
globalCache
ckbsettings.cpp, 1242
globalRemap
KbBind, 371
globalRemapTime
KbBind, 391
globalSettings
ckbsettings.cpp, 1241
globalThread
ckbsettings.cpp, 1242
goToFirstFile
QuaZip, 845
goToFirstUnmappedFile
QuaZipPrivate, 930
goToNextFile
QuaZip, 846
gradient
GradientDialog::Preset, 794
GradientButton, 195
_alpha, 201
_stops, 201
fromString, 197
GradientButton, 197
gradientChanged, 198
GradientButton, 197
pickGradient, 198
toString, 199
updateImage, 200
gradientChanged
GradientButton, 198
GradientDialog, 209
~GradientDialog, 213
addPreset, 213
colorChanged, 214
currentChanged, 214

currentPreset, 221
getGradient, 215
GradientDialog, 212
GradientDialog, 212
makeIcon, 216
on_presetDelete_clicked, 217
on_presetList_currentItemChanged, 217
on_presetName_textEdited, 218
on_presetSave_clicked, 218
on_stopOpacity_valueChanged, 219
on_stopPos_valueChanged, 219
presets, 221
setPreset, 219
ui, 222
updatePresets, 220
GradientDialog::Preset, 793
builtIn, 794
gradient, 794
name, 794
Preset, 794
GradientDialogWidget, 222
_colors, 237
_current, 237
_stops, 237
colorAt, 225
current, 226
currentChanged, 226
fillRect, 226
GradientDialogWidget, 225
GradientDialogWidget, 225
makeStops, 227
mouseMoveEvent, 228
mousePressEvent, 229
mouseReleaseEvent, 229
moveCurrent, 230
paintEvent, 231
selected, 237
selectedOffset, 237
selectedPos, 237
selectedStop, 232
selectionColors, 233
setCurrentColor, 233
setStops, 234
stopCount, 235
stopRect, 235
stops, 236
gradientdialog.cpp
prefsPath, 1256
gridLayout
Ui::AnimAddDialog, 51
Ui::AnimSettingDialog, 92
Ui::ExtraSettingsWidget, 166
Ui::FwUpgradeDialog, 194
Ui::GradientDialog, 206
Ui::KbAnimWidget, 340
Ui::KbBindWidget, 396
Ui::KbLightWidget, 456
Ui::KbProfileDialog, 564
Ui::KbWidget, 594
Ui::KPerfWidget, 691
Ui::LayoutDialog, 716
Ui::MainWindow, 732
Ui::ModeSelectDialog, 751
Ui::MPerfWidget, 768
Ui::RebindWidget, 953
Ui::SettingsWidget, 1026
Ui_AnimAddDialog, 1036
Ui_AnimSettingDialog, 1045
Ui_ExtraSettingsWidget, 1058
Ui_FwUpgradeDialog, 1067
Ui_GradientDialog, 1073
Ui_KbAnimWidget, 1082
Ui_KbBindWidget, 1089
Ui_KbLightWidget, 1097
Ui_KbProfileDialog, 1102
Ui_KbWidget, 1110
Ui_KPerfWidget, 1124
Ui_LayoutDialog, 1136
Ui_MainWindow, 1140
Ui_ModeSelectDialog, 1144
Ui_MPerfWidget, 1157
Ui_RebindWidget, 1183
Ui_SettingsWidget, 1203
gridLayout_10
Ui::RebindWidget, 953
Ui_RebindWidget, 1183
gridLayout_2
Ui::AnimSettingDialog, 92
Ui::FwUpgradeDialog, 194
Ui::GradientDialog, 206
Ui::KbAnimWidget, 340
Ui::KbBindWidget, 396
Ui::KbLightWidget, 456
Ui::KbWidget, 594
Ui::RebindWidget, 953
Ui_AnimSettingDialog, 1045
Ui_FwUpgradeDialog, 1067
Ui_GradientDialog, 1073
Ui_KbAnimWidget, 1082
Ui_KbBindWidget, 1089
Ui_KbLightWidget, 1097
Ui_KbWidget, 1110
Ui_RebindWidget, 1183
gridLayout_3
Ui::GradientDialog, 207
Ui::KbAnimWidget, 340
Ui::KbWidget, 594
Ui::RebindWidget, 953
Ui::SettingsWidget, 1026
Ui_GradientDialog, 1073
Ui_KbAnimWidget, 1082
Ui_KbWidget, 1110
Ui_RebindWidget, 1183
Ui_SettingsWidget, 1203
gridLayout_4
Ui::KbAnimWidget, 340

Ui::KbLightWidget, 456
Ui::KbWidget, 594
Ui::RebindWidget, 953
Ui_KbAnimWidget, 1082
Ui_KbLightWidget, 1097
Ui_KbWidget, 1110
Ui_RebindWidget, 1183
gridLayout_5
Ui::KbWidget, 594
Ui::RebindWidget, 953
Ui::SettingsWidget, 1026
Ui_KbWidget, 1110
Ui_RebindWidget, 1183
Ui_SettingsWidget, 1203
gridLayout_7
Ui::KbWidget, 594
Ui_KbWidget, 1110
gridLayout_8
Ui::RebindWidget, 953
Ui_RebindWidget, 1183
groupBox_2
Ui::GradientDialog, 207
Ui::RebindWidget, 953
Ui_GradientDialog, 1074
Ui_RebindWidget, 1183
groupBox_3
Ui::RebindWidget, 953
Ui_RebindWidget, 1184
groupBox_4
Ui::RebindWidget, 954
Ui_RebindWidget, 1184
groups
CkbSettings, 136
guid
AnimScript, 64
KbAnim, 316
UsbId, 1212
guidString
UsbId, 1209

HARPOON
KeyMap, 634
HIGH
KbPerf, 502
HW_IMAX
KbPerf, 503
HARPOON_HEIGHT
keymap.cpp, 1287
HARPOON_WIDTH
keymap.cpp, 1287
hLine
AnimSettingDialog, 107
HW_I_COUNT
KbPerf, 532
KPerfWidget, 711
HWMODE_MAX
Kb, 304
handleNotificationChannel
KbBind, 371
harpOverlay
keywidget.cpp, 1308
HarpoonKeys
keymap.cpp, 1300
hasCurrentFile
QuaZip, 847
hasCurrentFile_f
QuaZipPrivate, 932
hasDownloaded
KbFirmware, 417
hasFrame
AnimScript, 64
hasGPG
KbFirmware, 420
hasKeypress
AnimScript, 65
hasLed
Key, 599
hasParam
AnimScript, 65
hasParameter
KbAnim, 316
hasRepeat
AnimSettingDialog, 117
hasScan
Key, 599
hasShownNewFW
KbWidget, 587
hash
KbFirmware::FW, 180
height
KbPerf, 502
Key, 599
KeyMap, 642
helpStatus
RebindWidget, 973
horizontalLayout
Ui::AnimAddDialog, 51
Ui::ExtraSettingsWidget, 166
Ui::GradientDialog, 207
Ui::KbAnimWidget, 340
Ui::KbLightWidget, 457
Ui::KPerfWidget, 692
Ui::MPerfWidget, 768
Ui::RebindWidget, 954
Ui::SettingsWidget, 1026
Ui_AnimAddDialog, 1036
Ui_ExtraSettingsWidget, 1058
Ui_GradientDialog, 1074
Ui_KbAnimWidget, 1082
Ui_KbLightWidget, 1097
Ui_KPerfWidget, 1124
Ui_MPerfWidget, 1157
Ui_RebindWidget, 1184
Ui_SettingsWidget, 1203
horizontalLayout_2
Ui::ExtraSettingsWidget, 166
Ui::GradientDialog, 207

Ui::KbBindWidget, 396
Ui::KbLightWidget, 457
Ui::KPerfWidget, 692
Ui::MPerfWidget, 768
Ui::RebindWidget, 954
Ui::SettingsWidget, 1027
Ui_ExtraSettingsWidget, 1059
Ui_GradientDialog, 1074
Ui_KbBindWidget, 1089
Ui_KbLightWidget, 1097
Ui_KPerfWidget, 1124
Ui_MPerfWidget, 1157
Ui_RebindWidget, 1184
Ui_SettingsWidget, 1203
horizontalLayout_3
Ui::KbLightWidget, 457
Ui::RebindWidget, 954
Ui_KbLightWidget, 1097
Ui_RebindWidget, 1184
horizontalLayout_4
Ui::MPerfWidget, 768
Ui_MPerfWidget, 1157
horizontalSpacer
Ui::AnimAddDialog, 51
Ui::AnimSettingDialog, 92
Ui::ExtraSettingsWidget, 166
Ui::FwUpgradeDialog, 194
Ui::GradientDialog, 207
Ui::KbAnimWidget, 340
Ui::KbBindWidget, 396
Ui::KbProfileDialog, 564
Ui::KPerfWidget, 692
Ui::LayoutDialog, 716
Ui::MPerfWidget, 768
Ui::RebindWidget, 954
Ui::SettingsWidget, 1027
Ui_AnimAddDialog, 1036
Ui_AnimSettingDialog, 1045
Ui_ExtraSettingsWidget, 1059
Ui_FwUpgradeDialog, 1067
Ui_GradientDialog, 1074
Ui_KbAnimWidget, 1083
Ui_KbBindWidget, 1090
Ui_KbProfileDialog, 1103
Ui_KPerfWidget, 1124
Ui_LayoutDialog, 1136
Ui_MPerfWidget, 1157
Ui_RebindWidget, 1184
Ui_SettingsWidget, 1203
horizontalSpacer_10
Ui::RebindWidget, 954
Ui::SettingsWidget, 1027
Ui_RebindWidget, 1184
Ui_SettingsWidget, 1204
horizontalSpacer_11
Ui::RebindWidget, 954
Ui_RebindWidget, 1184
horizontalSpacer_12
Ui::RebindWidget, 954
Ui_RebindWidget, 1184
horizontalSpacer_13
Ui::RebindWidget, 954
Ui_RebindWidget, 1185
horizontalSpacer_14
Ui::RebindWidget, 955
Ui_RebindWidget, 1185
horizontalSpacer_15
Ui::RebindWidget, 955
Ui_RebindWidget, 1185
horizontalSpacer_16
Ui::RebindWidget, 955
Ui_RebindWidget, 1185
horizontalSpacer_17
Ui::RebindWidget, 955
Ui_RebindWidget, 1185
horizontalSpacer_18
Ui::RebindWidget, 955
Ui_RebindWidget, 1185
horizontalSpacer_19
Ui::RebindWidget, 955
Ui_RebindWidget, 1185
horizontalSpacer_2
Ui::AnimAddDialog, 51
Ui::AnimSettingDialog, 92
Ui::ExtraSettingsWidget, 166
Ui::FwUpgradeDialog, 194
Ui::GradientDialog, 207
Ui::KbAnimWidget, 341
Ui::KbBindWidget, 396
Ui::KbProfileDialog, 564
Ui::KbWidget, 594
Ui::KPerfWidget, 692
Ui::LayoutDialog, 716
Ui::MPerfWidget, 769
Ui::RebindWidget, 955
Ui::SettingsWidget, 1027
Ui_AnimAddDialog, 1036
Ui_AnimSettingDialog, 1045
Ui_ExtraSettingsWidget, 1059
Ui_FwUpgradeDialog, 1067
Ui_GradientDialog, 1074
Ui_KbAnimWidget, 1083
Ui_KbBindWidget, 1090
Ui_KbProfileDialog, 1103
Ui_KbWidget, 1110
Ui_KPerfWidget, 1125
Ui_LayoutDialog, 1136
Ui_MPerfWidget, 1158
Ui_RebindWidget, 1185
Ui_SettingsWidget, 1204
horizontalSpacer_20
Ui::RebindWidget, 955
Ui_RebindWidget, 1185
horizontalSpacer_21
Ui::RebindWidget, 955
Ui_RebindWidget, 1186

horizontalSpacer_22
Ui::RebindWidget, 956
Ui_RebindWidget, 1186
horizontalSpacer_23
Ui::RebindWidget, 956
Ui_RebindWidget, 1186
horizontalSpacer_24
Ui::RebindWidget, 956
Ui_RebindWidget, 1186
horizontalSpacer_25
Ui::RebindWidget, 956
Ui_RebindWidget, 1186
horizontalSpacer_26
Ui::RebindWidget, 956
Ui_RebindWidget, 1186
horizontalSpacer_3
Ui::AnimSettingDialog, 92
Ui::ExtraSettingsWidget, 167
Ui::FwUpgradeDialog, 194
Ui::GradientDialog, 207
Ui::KbAnimWidget, 341
Ui::KbLightWidget, 457
Ui::KbWidget, 595
Ui::KPerfWidget, 692
Ui::MPerfWidget, 769
Ui::RebindWidget, 956
Ui_AnimSettingDialog, 1045
Ui_ExtraSettingsWidget, 1059
Ui_FwUpgradeDialog, 1067
Ui_GradientDialog, 1074
Ui_KbAnimWidget, 1083
Ui_KbLightWidget, 1098
Ui_KbWidget, 1110
Ui_KPerfWidget, 1125
Ui_MPerfWidget, 1158
Ui_RebindWidget, 1186
horizontalSpacer_4
Ui::AnimSettingDialog, 92
Ui::FwUpgradeDialog, 194
Ui::KbAnimWidget, 341
Ui::KbLightWidget, 457
Ui::KPerfWidget, 692
Ui::MPerfWidget, 769
Ui::RebindWidget, 956
Ui_AnimSettingDialog, 1045
Ui_FwUpgradeDialog, 1067
Ui_KbAnimWidget, 1083
Ui_KbLightWidget, 1098
Ui_KPerfWidget, 1125
Ui_MPerfWidget, 1158
Ui_RebindWidget, 1186
horizontalSpacer_5
Ui::AnimSettingDialog, 92
Ui::KbAnimWidget, 341
Ui::KbLightWidget, 457
Ui::KPerfWidget, 692
Ui::MPerfWidget, 769
Ui::RebindWidget, 956
Ui_AnimSettingDialog, 1045
Ui_KbAnimWidget, 1083
Ui_KbLightWidget, 1098
Ui_KPerfWidget, 1125
Ui_MPerfWidget, 1158
Ui_RebindWidget, 1186
horizontalSpacer_6
Ui::AnimSettingDialog, 92
Ui::KbLightWidget, 457
Ui::KPerfWidget, 692
Ui::MPerfWidget, 769
Ui::RebindWidget, 956
Ui_AnimSettingDialog, 1045
Ui_KbLightWidget, 1098
Ui_KPerfWidget, 1125
Ui_MPerfWidget, 1158
Ui_RebindWidget, 1187
horizontalSpacer_7
Ui::KPerfWidget, 692
Ui::MPerfWidget, 769
Ui::RebindWidget, 957
Ui::SettingsWidget, 1027
Ui_KPerfWidget, 1125
Ui_MPerfWidget, 1158
Ui_RebindWidget, 1187
Ui_SettingsWidget, 1204
horizontalSpacer_8
Ui::KPerfWidget, 693
Ui::RebindWidget, 957
Ui_KPerfWidget, 1125
Ui_RebindWidget, 1187
horizontalSpacer_9
Ui::RebindWidget, 957
Ui::SettingsWidget, 1027
Ui_RebindWidget, 1187
Ui_SettingsWidget, 1204
hwIType
KbPerf, 532
hwLoading
Kb, 304
HwMode
KPerfWidget, 704
hwModeCount
Kb, 304
hwModified
UsbId, 1212
hwModifiedString
UsbId, 1210
hwProfile
Kb, 275
hwSave
Kb, 276
hwSaveButton
Ui::KbWidget, 595
Ui_KbWidget, 1111

IT
KeyMap, 633
I_COUNT

KbPerf, 532
KPerfWidget, 712
i_hw
KbPerf, 502
KPerfWidget, 703
iButton0
Ui::MPerfWidget, 769
Ui_MPerfWidget, 1158
iButton1
Ui::MPerfWidget, 769
Ui_MPerfWidget, 1158
iButton2
Ui::MPerfWidget, 769
Ui_MPerfWidget, 1158
iButton3
Ui::MPerfWidget, 770
Ui_MPerfWidget, 1159
iButton4
Ui::MPerfWidget, 770
Ui_MPerfWidget, 1159
iButton5
Ui::MPerfWidget, 770
Ui_MPerfWidget, 1159
iButtonO
Ui::MPerfWidget, 770
Ui_MPerfWidget, 1159
iColor
KbPerf, 532
iEnable
KbPerf, 532
iOpacity
KbPerf, 516, 517
iState
Kb, 304
iconClicked
MainWindow, 740
id
KbMode, 490, 491
KbProfile, 540
ids
fwupgradedialog.cpp, 1253
inFrame
AnimScript, 82
indexOf
Kb, 277
KbProfile, 541
indicBox
Ui::MPerfWidget, 770
Ui_MPerfWidget, 1159
indicator
KbPerf, 503
KPerfWidget, 704
indicators
KPerfWidget, 712
infoTab
Ui::AnimSettingDialog, 92
Ui_AnimSettingDialog, 1046
infoUpdated
Kb, 278
init
AnimScript, 66
ColorMap, 155
KbManager, 478
init_keys
crypt.h, 1336
init_linkedlist
zip.c, 1468
initialized
AnimScript, 82
insert
KbProfile, 541
instance
KbFirmware, 420
intensityBox
Ui::KPerfWidget, 693
Ui_KPerfWidget, 1125
internal
QuaZipFilePrivate, 914
internalAttr
QuaZipFileInfo, 903
QuaZipFileInfo64, 910
QuaZipNewInfo, 924
ioDevice
QuaZipPrivate, 932
ioapi.h
_FILE_OFFSET_BIT, 1341
_LARGEFILE64_SOURCE, 1342
_Z_OF, 1344
__USE_LARGEFILE64, 1341
fill_zlib_filefunc64_32_def_from_filefunc32, 1344
OF, 1342
ZCALLBACK, 1342
ZCLOSE64, 1342
ZERROR64, 1342
ZFAKECLOSE64, 1342
ZOPEN64, 1343
ZPOS64_T, 1344
ZREAD64, 1343
ZSEEK64, 1343
ZTELL64, 1344
ZWRITE64, 1344
zlib_filefunc64_32_def, 1344
zlib_filefunc64_def, 1344
zlib_filefunc_def, 1344
isActive
KbAnim, 317
isAnim
KeyAction, 611
isAutoClose
QuaZip, 847
isBusy
CkbSettings, 130
isDPI
KeyAction, 612
isDataDescriptorWritingEnabled
QuaZip, 847

isEnabled
AutoRun, 120
isEncrypted
QuaZipFileInfo64, 908
isISO
KbBind, 372
KeyMap, 643
isJP
KeyMap, 644
isKeyboard
Kb, 278
KbBind, 373
KeyMap, 645
isMacro
KeyAction, 612
isMedia
KeyAction, 612
isMouse
Kb, 279
KbBind, 373
KeyAction, 613
KeyMap, 646
isNormal
KeyAction, 613
isOpen
Kb, 280
QuaZip, 848
isPTBR
KeyMap, 647
isProgram
KeyAction, 614
isRaw
QuaZipFile, 889
isRoot
QuaZipDir, 872
isRunning
KbAnim, 317
main.cpp, 1317
isSequential
QuaGzipFile, 817
QuaZIODevice, 824
QuaZipFile, 890
isSetting
MPerfWidget, 792
isSpecial
KeyAction, 614
isStarted
KbLight, 436
isUnbound
KeyAction, 615
isValidMacro
KeyAction, 615
isZip64Enabled
QuaZip, 848

JP
KeyMap, 633
JlCompress, 238
compressDir, 239
compressFile, 240, 241
compressFiles, 242
compressSubDir, 244
extractDir, 247
extractFile, 249, 251
extractFiles, 252
getFileList, 253
removeFile, 254
JlCompress.cpp
copyData, 1346

K63
KeyMap, 634
K63_HEIGHT
keymap.cpp, 1287
K63_TOP_COUNT
keymap.cpp, 1287
K63_WIDTH
keymap.cpp, 1288
K63TopRow
keymap.cpp, 1300
K65
KeyMap, 634
K65_HEIGHT
keymap.cpp, 1288
K65_TOP_COUNT
keymap.cpp, 1288
K65_WIDTH
keymap.cpp, 1288
K65TopRow
keymap.cpp, 1301
K68
KeyMap, 634
K68_HEIGHT
keymap.cpp, 1288
K68_TOP_COUNT
keymap.cpp, 1288
K68_WIDTH
keymap.cpp, 1288
K68TopRow
keymap.cpp, 1301
K70
KeyMap, 634
K70_HEIGHT
keymap.cpp, 1288
K70_WIDTH
keymap.cpp, 1288
K70_X_START
keymap.cpp, 1288
K95
KeyMap, 634
K95P
KeyMap, 634
K95_HEIGHT
keymap.cpp, 1289
K95_WIDTH
keymap.cpp, 1289
K95BaseMap
keymap.cpp, 1301

K95Keys
keymap.cpp, 1301
k95Label1
Ui::KPerfWidget, 693
Ui_KPerfWidget, 1125
k95Label2
Ui::KPerfWidget, 693
Ui_KPerfWidget, 1126
k95Label3
Ui::KPerfWidget, 693
Ui_KPerfWidget, 1126
k95Label4
Ui::KPerfWidget, 693
Ui_KPerfWidget, 1126
k95Label5
Ui::KPerfWidget, 693
Ui_KPerfWidget, 1126
k95Label6
Ui::KPerfWidget, 693
Ui_KPerfWidget, 1126
k95Line
Ui::KPerfWidget, 693
Ui_KPerfWidget, 1126
K95P_HEIGHT
keymap.cpp, 1289
K95P_WIDTH
keymap.cpp, 1289
K95P_X_START
keymap.cpp, 1289
K95PLbar
keymap.cpp, 1302
k95Spacer
Ui::KPerfWidget, 694
Ui_KPerfWidget, 1126
k95Widgets
KPerfWidget, 712
KPerfWidget
ALWAYS_OFF, 704
ALWAYS_ON, 704
BOTH, 704
NORMAL, 704
RGB, 704
KPerfWidget::IndicatorUi, 702, 1311
KEY_SIZE
keywidget.cpp, 1309
KEYCOUNT_GLAIVE
keymap.cpp, 1289
KEYCOUNT_HARPOON
keymap.cpp, 1289
KEYCOUNT_K95
keymap.cpp, 1289
KEYCOUNT_M65
keymap.cpp, 1289
KEYCOUNT_SABRE
keymap.cpp, 1290
KEYCOUNT_SCIM
keymap.cpp, 1290
KP_NAME
AnimScript, 82
KP_NONE
AnimScript, 82
KP_POSITION
AnimScript, 82
kPerfTab
Ui::KbWidget, 595
Ui_KbWidget, 1111
KPerfWidget, 700
∼KPerfWidget, 706
HW_I_COUNT, 711
HwMode, 704
I_COUNT, 712
i_hw, 703
indicator, 704
indicators, 712
k95Widgets, 712
KPerfWidget, 704
KPerfWidget, 704
mode2Raw, 706
on_copyButton_clicked, 707
on_intensityBox_valueChanged, 708
perf, 712
profile, 712
raw2Mode, 708
setPerf, 709
ui, 712
uiUpdated, 710
updateMapper, 712
kPerfWidget
Ui::KbWidget, 595
Ui_KbWidget, 1111
KSTRAFE_HEIGHT
keymap.cpp, 1290
KSTRAFE_WIDTH
keymap.cpp, 1290
KSTRAFE_X_START
keymap.cpp, 1290
KStrafeKeys
keymap.cpp, 1302
Kb, 256
∼Kb, 261
_currentMode, 302
_currentProfile, 302
_delay, 302
_dither, 302
_frameRate, 302
_hwProfile, 302
_layout, 303
_model, 303
_mouseAccel, 303
_needsSave, 303
_profiles, 303
_scrollSpeed, 303
appendProfile, 264
autoSave, 264
cmd, 303
cmdpath, 303

currentBind, 264
currentLight, 265
currentMode, 266
currentPerf, 266
currentProfile, 267
deleteHw, 267
deletePrevious, 268
devpath, 303
dither, 268, 269
features, 304
find, 269
firmware, 304
frameRate, 270
frameUpdate, 271
fwUpdPath, 304
fwUpdate, 272
fwUpdateFinished, 273
fwUpdateProgress, 273
getKeyMap, 274
getMacroNumber, 274
getMacroPath, 274
HWMODE_MAX, 304
hwLoading, 304
hwModeCount, 304
hwProfile, 275
hwSave, 276
iState, 304
indexOf, 277
infoUpdated, 278
isKeyboard, 278
isMouse, 279
isOpen, 280
Kb, 261
KbManager, 302
lastAutoSave, 304
layout, 280
load, 281
macroDelay, 282
macroNumber, 304
macroPath, 305
matches, 283
modeChanged, 283
modeRenamed, 284
model, 283
monochrome, 305
mouseAccel, 284
needsSave, 285
newMode, 285, 286
newProfile, 286, 287
notifyNumber, 305
notifyPath, 305
pollrate, 305
prefsPath, 305
prevMode, 305
prevProfile, 305
profileAdded, 288
profileChanged, 288
profileRenamed, 288
profiles, 289
readNotify, 290
run, 294
save, 295
scrollSpeed, 296
setCurrentMode, 297, 298
setCurrentProfile, 299
updateLayout, 300
usbModel, 305
usbSerial, 306
writeProfileHeader, 301
kb
FwUpgradeDialog, 188
kb.cpp
activeDevices, 1259
notifyPathMutex, 1259
notifyPaths, 1259
KbAnim
Add, 309
Divide, 310
Multiply, 309
Normal, 309
Subtract, 309
KbFirmware
NO, 412
UNKNOWN, 412
YES, 412
KbId, 1251
KbPerf
CAPS, 503
HIGH, 502
HW_IMAX, 503
LIGHT, 503
LOCK, 503
LOW, 502
LOWMED, 502
MACRO, 503
MEDHIGH, 502
MEDIUM, 502
MODE, 503
MUTE, 503
NONE, 503
NORMAL, 503
NUM, 503
OFF, 503
ON, 503
SCROLL, 503
KbAnim, 306
_guid, 331
_isActive, 331
_isActiveKp, 331
_keys, 331
_map, 331
_mode, 331
_name, 331
_needsSave, 331
_opacity, 332
_parameters, 332

_script, 332
_scriptGuid, 332
_scriptName, 332
_tempParameters, 332
blend, 312
catchUp, 313
commitParams, 315
effectiveParams, 315
guid, 316
hasParameter, 316
isActive, 317
isRunning, 317
KbAnim, 310–312
KbAnim, 310–312
keypress, 318
keys, 319, 320
kpRepeatMsec, 332
kpRepeatTime, 332
kpStopTime, 332
map, 320, 321
Mode, 309
mode, 321
name, 321, 322
needsSave, 322
newId, 323
opacity, 323
parameter, 323, 324
reInit, 325
repeatKey, 333
repeatMsec, 333
repeatTime, 333
resetParams, 325
save, 326
script, 327
scriptName, 327
stop, 328
stopTime, 333
trigger, 329
updateParams, 330
KbAnimWidget, 343
~KbAnimWidget, 346
addAnim, 346
animChanged, 347
animations, 360
clearSelection, 348
current, 360
didUpdateSelection, 348
duplicateAnim, 349
KbAnimWidget, 345
KbAnimWidget, 345
light, 360
noReorder, 360
on_animList_currentItemChanged, 349
on_animList_customContextMenuRequested, 350
on_animList_itemChanged, 351
on_blendBox_activated, 351
on_deleteButton_clicked, 352
on_keyButton_clicked, 352
on_nameBox_textEdited, 353
on_opacityBox_valueChanged, 354
on_propertyButton_clicked, 354
refreshList, 355
reorderAnims, 356
selectedKeys, 360
setCurrent, 357
setLight, 358
setSelectedKeys, 359
ui, 360
KbBind, 361
_bind, 390
_devParent, 390
_globalRemap, 390
_map, 390
_needsSave, 390
_needsUpdate, 391
_winLock, 391
action, 365
bindAction, 366
defaultAction, 366, 367
devParent, 367
didLoad, 368
friendlyActionName, 368
friendlyName, 369
getMacroNumber, 369
getMacroPath, 370
globalRemap, 371
globalRemapTime, 391
handleNotificationChannel, 371
isISO, 372
isKeyboard, 373
isMouse, 373
KbBind, 364
KbBind, 364
KeyAction, 390
keyEvent, 374
lastCmd, 391
lastGlobalRemapTime, 391
layoutChanged, 375
light, 376
load, 376
loadGlobalRemap, 378
map, 378, 379
modeParent, 379
needsSave, 379
noAction, 380
perf, 381
resetAction, 382
save, 383
saveGlobalRemap, 384
setAction, 385
setGlobalRemap, 386
setNeedsUpdate, 387
update, 387
updated, 389
winLock, 389, 390
KbBindWidget, 398

~KbBindWidget, 400
bind, 408
currentSelection, 408
KbBindWidget, 400
KbBindWidget, 400
newLayout, 401
newSelection, 402
on_copyButton_clicked, 403
on_resetButton_clicked, 404
profile, 408
setBind, 405
ui, 409
updateBind, 406
updateSelDisplay, 407
kbConnected
KbManager, 479
kbDisconnected
KbManager, 479
KbFirmware, 409
~KbFirmware, 412
_checkUpdates, 412
_fileForBoard, 413
_latestForBoard, 414
checkUpdates, 415
dataForBoard, 415
downloadFinished, 416
downloaded, 416
fwTable, 420
fwTableHash, 420
hasDownloaded, 417
hasGPG, 420
instance, 420
KbFirmware, 412
KbFirmware, 412
lastCheck, 420
lastFinished, 420
networkManager, 420
processDownload, 417
tableDownload, 420
versionForBoard, 419
KbFirmware::FW, 179
ckbVersion, 180
FW, 179
fileName, 180
fwVersion, 180
hash, 180
url, 180
KbLight, 421
~KbLight, 425
_animList, 448
_animMap, 448
_colorMap, 449
_dimming, 449
_indicatorList, 449
_indicatorMap, 449
_lastFrameAnimMap, 449
_lastFrameDimming, 449
_lastFrameIndicatorMap, 449
_map, 449
_needsMapRefresh, 449
_needsSave, 449
_previewAnim, 450
_qColorMap, 450
_start, 450
addAnim, 425
animKeypress, 426
AnimList, 424
animList, 427
base, 427
close, 428
color, 429, 430
colorMap, 430
didLoad, 431
dimming, 431, 432
duplicateAnim, 432
findAnim, 433
findAnimIdx, 434
frameDisplayed, 434
frameUpdate, 434
isStarted, 436
KbLight, 424
KbLight, 424
lastFrameSignal, 450
load, 437
MAX_DIM, 450
map, 438, 439
needsSave, 439
open, 440
previewAnim, 441
printRGB, 442
rebuildBaseMap, 443
resetIndicators, 443
restartAnimation, 444
save, 445
setIndicator, 446
shareDimming, 446, 447
stopPreview, 447
updated, 448
KbLightWidget, 459
~KbLightWidget, 461
changeAnim, 462
changeAnimKeys, 462
changeColor, 463
currentSelection, 471
KbLightWidget, 461
KbLightWidget, 461
light, 471
newSelection, 464
on_animButton_clicked, 465
on_bgButton_clicked, 466
on_brightnessBox_activated, 466
on_showAnimBox_clicked, 467
setLight, 467
setMonochrome, 468
toggleSidelight, 469
ui, 471

updateLight, 470
KbManager, 472
_daemonVersion, 483
_devices, 483
_eventTimer, 483
_guiVersion, 483
_kbManager, 484
_scanTimer, 484
ckbDaemonVersion, 475
ckbDaemonVersionF, 475
ckbGuiVersion, 476
ckbGuiVersionF, 476
devices, 477
eventTimer, 477
fps, 477
init, 478
Kb, 302
kbConnected, 479
kbDisconnected, 479
KbManager, 474
kbManager, 479
KbManager, 474
parseVersionString, 480
scanKeyboards, 480
scanTimer, 482
stop, 482
versionUpdated, 483
kbManager
KbManager, 479
KbMode, 484
~KbMode, 489
_bind, 498
_id, 498
_light, 498
_name, 498
_needsSave, 498
_perf, 498
bind, 489
doUpdate, 490
id, 490, 491
KbMode, 487, 488
KbMode, 487, 488
keyMap, 491
light, 492
name, 492, 493
needsSave, 493
newId, 494
perf, 495
save, 495
setNeedsSave, 496
setNeedsUpdate, 496
updated, 497
KbPerf, 498
_angleSnap, 530
_curDpi, 505
_dpiIndicator, 530
_iOpacity, 530
_liftHeight, 531
_needsSave, 531
_needsUpdate, 531
angleSnap, 505, 506
applyIndicators, 506
baseDpiIdx, 507, 508
bind, 508
DPI_COUNT, 531
DPI_MAX, 531
DPI_MIN, 531
didLoad, 509
dpi, 509, 510
dpiBaseIdx, 531
dpiClr, 531
dpiColor, 511, 512
dpiCurX, 531
dpiCurY, 532
dpiCycleDown, 512
dpiCycleUp, 513
dpiDown, 513
dpiEnabled, 514
dpiIndicator, 515
dpiOn, 532
dpiUp, 515
dpiX, 532
dpiY, 532
getIndicator, 516
HW_I_COUNT, 532
height, 502
hwIType, 532
I_COUNT, 532
i_hw, 502
iColor, 532
iEnable, 532
iOpacity, 516, 517
indicator, 503
KbPerf, 503, 504
KbPerf, 503, 504
liftHeight, 517
light, 517
light100Color, 533
lightIndicator, 518
load, 519
modeParent, 521
muteNAColor, 533
needsSave, 521
OTHER, 533
operator=, 522
popDpi, 522
pushDpi, 523, 524
pushSniper, 525
pushedDpis, 533
runningPushIdx, 533
SNIPER, 533
save, 525
setIndicator, 526
setNeedsUpdate, 527
settingsUpdated, 527
sniperDpi, 527, 528

update, 529
KbProfile, 533
_currentMode, 550
_id, 550
_keyMap, 550
_modes, 550
_name, 550
_needsSave, 551
append, 538
currentMode, 538, 539
find, 539
id, 540
indexOf, 541
insert, 541
KbProfile, 536, 537
KbProfile, 536, 537
keyMap, 542
modeCount, 543
ModeList, 536
modes, 543, 544
move, 544
name, 545
needsSave, 545
newId, 546
removeAll, 547
save, 548
setNeedsSave, 549
setNeedsUpdate, 549
KbProfileDialog, 551
~KbProfileDialog, 553
addNewProfileItem, 553
device, 559
GUID, 560
KbProfileDialog, 553
KbProfileDialog, 553
NEW_FLAG, 560
on_profileList_currentItemChanged, 554
on_profileList_customContextMenuRequested, 554
on_profileList_itemChanged, 556
on_profileList_itemClicked, 557
profileList_reordered, 557
repopulate, 558
ui, 560
KbWidget, 565
~KbWidget, 569
addNewModeItem, 569
currentMode, 586
devUpdate, 570
device, 586
GUID, 587
hasShownNewFW, 587
KbWidget, 567
KbWidget, 567
lastAutoSave, 587
modeChanged, 570
modeIcon, 572
modeUpdate, 574
modesList_reordered, 573
NEW_FLAG, 587
name, 574
on_fwUpdButton_clicked, 574
on_hwSaveButton_clicked, 576
on_modesList_currentItemChanged, 577
on_modesList_customContextMenuRequested, 578
on_modesList_itemChanged, 579
on_modesList_itemClicked, 580
on_profileBox_activated, 581
on_tabWidget_currentChanged, 581
profileChanged, 582
showFirstTab, 583
showFwUpdate, 583
showLastTab, 584
ui, 587
updateFwButton, 584
updateProfileList, 585
kbWidgets
MainWindow, 746
kbanim.cpp
blendAdd, 1261
blendDivide, 1261
blendFunc, 1261
blendMultiply, 1261
blendNormal, 1261
blendSubtract, 1262
functions, 1262
kbfirmware.cpp
AUTO_CHECK_TIME, 1268
tableName, 1268
kblight.cpp
_shareDimming, 1272
activeLights, 1272
lToS, 1270
monoRgb, 1270
sToL, 1271
kbmanager.cpp
devpath, 1275
kbperf.cpp
map_last, 1277
Key, 597
_friendlyName, 599
_storageName, 599
friendlyName, 598
hasLed, 599
hasScan, 599
height, 599
name, 600
operator bool, 598
storageName, 599
width, 600
x, 600
y, 600
key
KeyMap, 648
KeyAction

NORMAL, 604
SPECIAL, 604
UNBOUND, 604
KeyMap
_LAYOUT_MAX, 633
_MODEL_MAX, 634
DE, 633
DK, 633
ES, 633
EU, 633
EU_DVORAK, 633
FR, 633
GB, 633
GB_DVORAK, 633
GLAIVE, 634
HARPOON, 634
IT, 633
JP, 633
K63, 634
K65, 634
K68, 634
K70, 634
K95, 634
K95P, 634
M65, 634
MX, 633
NO, 633
NO_LAYOUT, 633
NO_MODEL, 634
PL, 633
PT_BR, 633
SABRE, 634
SCIMITAR, 634
SE, 633
STRAFE, 634
US, 633
US_DVORAK, 633
KeyPatch, 1286
KeyWidget
ADD, 660
NONE, 660
SET, 660
SUBTRACT, 660
TOGGLE, 660
KeyAction, 600
∼KeyAction, 605
_value, 627
adjustDisplay, 605
animAction, 606
animInfo, 607
DPI_CUSTOM, 627
DPI_CYCLE_DOWN, 628
DPI_CYCLE_UP, 628
DPI_DOWN, 628
DPI_SNIPER, 628
DPI_UP, 628
defaultAction, 607
dpiAction, 608
dpiInfo, 609
driverName, 609
friendlyName, 610
isAnim, 611
isDPI, 612
isMacro, 612
isMedia, 612
isMouse, 613
isNormal, 613
isProgram, 614
isSpecial, 614
isUnbound, 615
isValidMacro, 615
KbBind, 390
KeyAction, 605
keyEvent, 616
keyRelease, 619
KeyAction, 605
LIGHT_DOWN, 628
LIGHT_DOWN_WRAP, 628
LIGHT_UP, 628
LIGHT_UP_WRAP, 628
LOCK_OFF, 629
LOCK_ON, 629
LOCK_TOGGLE, 629
lightAction, 620
lockAction, 620
MODE_NEXT, 629
MODE_NEXT_WRAP, 629
MODE_PREV, 629
MODE_PREV_WRAP, 629
macroAction, 620
macroContent, 621
macroDisplay, 621
macroFullLine, 622
macroLine, 623
macroTiming, 623
modeAction, 624
noAction, 624
operator QString, 625
operator=, 625
PROGRAM_PR_INDEF, 629
PROGRAM_PR_KPSTOP, 630
PROGRAM_PR_KRSTOP, 630
PROGRAM_PR_MULTI, 630
PROGRAM_RE_INDEF, 630
PROGRAM_RE_KPSTOP, 630
PROGRAM_RE_MULTI, 630
preProgram, 629
programAction, 625
programInfo, 625
relProgram, 630
sniperValue, 630
specialInfo, 626
Type, 604
type, 626
value, 627
keyButton

Ui::KbAnimWidget, 341
Ui_KbAnimWidget, 1083
keyEvent
KbBind, 374
KeyAction, 616
keyHeight
KeyMap, 656
keyLayout
KeyMap, 656
KeyMap, 631
_keys, 656
byPosition, 635
contains, 636
count, 636
friendlyName, 637
fromName, 638
fromStorage, 639
getLayout, 639, 640
getModel, 641, 642
height, 642
isISO, 643
isJP, 644
isKeyboard, 645
isMouse, 646
isPTBR, 647
key, 648
keyHeight, 656
keyLayout, 656
KeyMap, 634, 635
keyModel, 656
keyWidth, 656
KeyMap, 634, 635
keys, 648
Layout, 633
layout, 649
layoutNames, 649
locale, 650
map, 651
Model, 633
model, 651
modelHeight, 651
modelWidth, 652
name, 652
operator const QHash< QString, Key > &, 653
positions, 653
strLayout, 654
strModel, 654
toStorage, 655
width, 655
keyMap
KbMode, 491
KbProfile, 542
KeyWidget, 678
keyModel
KeyMap, 656
keyNames
ColorMap, 156
keyRelease
KeyAction, 619
keyTab
Ui::RebindWidget, 957
Ui_RebindWidget, 1187
KeyWidget, 656
_bindMap, 677
_colorMap, 677
_displayColorMap, 678
_indicators, 678
_monochrome, 678
_rgbMode, 678
animation, 678
BindMap, 659
bindMap, 660
clearAnimation, 661
clearSelection, 661
colorMap, 661, 662
displayColorMap, 662
drawInfo, 662
keyMap, 678
KeyWidget, 660
KeyWidget, 660
map, 663, 664
monochrome, 664, 665
mouseCurrentX, 678
mouseCurrentY, 678
mouseDownMode, 678
mouseDownX, 679
mouseDownY, 679
mouseMoveEvent, 665
mousePressEvent, 666
mouseReleaseEvent, 667
newSelection, 679
paintEvent, 668
rgbMode, 673, 674
selectAll, 674
selection, 679
selectionChanged, 675
setAnimation, 675
setAnimationToSelection, 676
setSelection, 676
sidelightToggled, 677
keyWidget
Ui::KbBindWidget, 396
Ui::KbLightWidget, 457
Ui_KbBindWidget, 1090
Ui_KbLightWidget, 1098
keyWidth
KeyMap, 656
keyaction.cpp
ANIM_KRSTOP, 1283
ANIM_ONCE, 1283
keymap.cpp
_patch, 1291
ANSI_ENTER_H, 1286
ANSI_ENTER_W, 1287
ANSI_ENTER_X, 1287
ANSI_ENTER_Y, 1287

ANSI_LSHIFT_W, 1287
ANSI_LSHIFT_X, 1287
GLAIVE_HEIGHT, 1287
GLAIVE_WIDTH, 1287
getMap, 1292
GlaiveKeys, 1300
HARPOON_HEIGHT, 1287
HARPOON_WIDTH, 1287
HarpoonKeys, 1300
K63_HEIGHT, 1287
K63_TOP_COUNT, 1287
K63_WIDTH, 1288
K63TopRow, 1300
K65_HEIGHT, 1288
K65_TOP_COUNT, 1288
K65_WIDTH, 1288
K65TopRow, 1301
K68_HEIGHT, 1288
K68_TOP_COUNT, 1288
K68_WIDTH, 1288
K68TopRow, 1301
K70_HEIGHT, 1288
K70_WIDTH, 1288
K70_X_START, 1288
K95_HEIGHT, 1289
K95_WIDTH, 1289
K95BaseMap, 1301
K95Keys, 1301
K95P_HEIGHT, 1289
K95P_WIDTH, 1289
K95P_X_START, 1289
K95PLbar, 1302
KEYCOUNT_GLAIVE, 1289
KEYCOUNT_HARPOON, 1289
KEYCOUNT_K95, 1289
KEYCOUNT_M65, 1289
KEYCOUNT_SABRE, 1290
KEYCOUNT_SCIM, 1290
KSTRAFE_HEIGHT, 1290
KSTRAFE_WIDTH, 1290
KSTRAFE_X_START, 1290
KStrafeKeys, 1302
LBARCOUNT_K95P, 1290
LBS, 1290
M65_HEIGHT, 1290
M65_WIDTH, 1290
M65Keys, 1303
N_LAYOUTS, 1290
N_MODELS, 1291
NS, 1291
PATCH_COUNT, 1291
patch, 1291
patchABNT2, 1296
patchANSI, 1297
patchDE, 1303
patchDK, 1303
patchDvorak, 1303
patchES, 1303
patchEU, 1304
patchFR, 1304
patchISO, 1297
patchIT, 1304
patchJP, 1304
patchJP106, 1298
patchJP106fn, 1299
patchMX, 1304
patchNO, 1305
patchPTBR, 1305
patchSE, 1305
patchnonJP106, 1299
SABRE_HEIGHT, 1291
SABRE_WIDTH, 1291
SCIM_HEIGHT, 1291
SCIM_WIDTH, 1291
SabreKeys, 1305
ScimKeys, 1306
standardMaps, 1306
keypress
AnimScript, 67
KbAnim, 318
keys
KbAnim, 319, 320
KeyMap, 648
keywidget.cpp
glaiveOverlay, 1308
harpOverlay, 1308
KEY_SIZE, 1309
m65Overlay, 1309
monoRgb, 1308
sabOverlay, 1309
scimOverlay, 1309
kpDelayBox
Ui::AnimSettingDialog, 92
Ui_AnimSettingDialog, 1046
kpModeStopBox
Ui::AnimSettingDialog, 93
Ui_AnimSettingDialog, 1046
kpReleaseBox
Ui::AnimSettingDialog, 93
Ui_AnimSettingDialog, 1046
kpRepeatBox
Ui::AnimSettingDialog, 93
Ui_AnimSettingDialog, 1046
kpRepeatLabel
Ui::AnimSettingDialog, 93
Ui_AnimSettingDialog, 1046
kpRepeatLabel_2
Ui::AnimSettingDialog, 93
Ui_AnimSettingDialog, 1046
kpRepeatMsec
KbAnim, 332
kpRepeatTime
KbAnim, 332
kpStopCheck
AnimSettingDialog, 117
kpStopTime

KbAnim, 332

LIGHT
KbPerf, 503
LOCK
KbPerf, 503
LOW
KbPerf, 502
LOWMED
KbPerf, 502
LBARCOUNT_K95P
keymap.cpp, 1290
LBS
keymap.cpp, 1290
lHeightBox
Ui::MPerfWidget, 772
Ui_MPerfWidget, 1161
LIGHT_DOWN
KeyAction, 628
LIGHT_DOWN_WRAP
KeyAction, 628
LIGHT_UP
KeyAction, 628
LIGHT_UP_WRAP
KeyAction, 628
LOCALHEADERMAGIC
zip.c, 1463
LOCK_OFF
KeyAction, 629
LOCK_ON
KeyAction, 629
LOCK_TOGGLE
KeyAction, 629
lToS
kblight.cpp, 1270
label
Ui::AnimAddDialog, 51
Ui::AnimSettingDialog, 93
Ui::ExtraSettingsWidget, 167
Ui::GradientDialog, 207
Ui::KbBindWidget, 396
Ui::KbLightWidget, 457
Ui::KPerfWidget, 694
Ui::LayoutDialog, 716
Ui::ModeSelectDialog, 751
Ui::MPerfWidget, 770
Ui::RebindWidget, 957
Ui::SettingsWidget, 1027
Ui_AnimAddDialog, 1037
Ui_AnimSettingDialog, 1046
Ui_ExtraSettingsWidget, 1059
Ui_GradientDialog, 1074
Ui_KbBindWidget, 1090
Ui_KbLightWidget, 1098
Ui_KPerfWidget, 1126
Ui_LayoutDialog, 1136
Ui_ModeSelectDialog, 1144
Ui_MPerfWidget, 1159
Ui_RebindWidget, 1187
Ui_SettingsWidget, 1204
label_10
Ui::AnimSettingDialog, 93
Ui::ExtraSettingsWidget, 167
Ui::KPerfWidget, 694
Ui_AnimSettingDialog, 1046
Ui_ExtraSettingsWidget, 1059
Ui_KPerfWidget, 1126
label_11
Ui::AnimSettingDialog, 93
Ui::ExtraSettingsWidget, 167
Ui::KPerfWidget, 694
Ui::MPerfWidget, 770
Ui_AnimSettingDialog, 1047
Ui_ExtraSettingsWidget, 1059
Ui_KPerfWidget, 1127
Ui_MPerfWidget, 1159
label_12
Ui::AnimSettingDialog, 93
Ui::KPerfWidget, 694
Ui::MPerfWidget, 770
Ui_AnimSettingDialog, 1047
Ui_KPerfWidget, 1127
Ui_MPerfWidget, 1159
label_13
Ui::AnimSettingDialog, 94
Ui::MPerfWidget, 770
Ui_AnimSettingDialog, 1047
Ui_MPerfWidget, 1159
label_14
Ui::KPerfWidget, 694
Ui::MPerfWidget, 771
Ui_KPerfWidget, 1127
Ui_MPerfWidget, 1160
label_15
Ui::KPerfWidget, 694
Ui::MPerfWidget, 771
Ui_KPerfWidget, 1127
Ui_MPerfWidget, 1160
label_16
Ui::KPerfWidget, 694
Ui_KPerfWidget, 1127
label_17
Ui::KPerfWidget, 694
Ui_KPerfWidget, 1127
label_18
Ui::KPerfWidget, 695
Ui_KPerfWidget, 1127
label_19
Ui::KPerfWidget, 695
Ui_KPerfWidget, 1127
label_2
Ui::AnimAddDialog, 51
Ui::AnimSettingDialog, 94
Ui::ExtraSettingsWidget, 167
Ui::FwUpgradeDialog, 194
Ui::GradientDialog, 207
Ui::KbBindWidget, 397

Ui::KbLightWidget, 458
Ui::KPerfWidget, 695
Ui::LayoutDialog, 717
Ui::MPerfWidget, 771
Ui::SettingsWidget, 1027
Ui_AnimAddDialog, 1037
Ui_AnimSettingDialog, 1047
Ui_ExtraSettingsWidget, 1059
Ui_FwUpgradeDialog, 1068
Ui_GradientDialog, 1074
Ui_KbBindWidget, 1090
Ui_KbLightWidget, 1098
Ui_KPerfWidget, 1127
Ui_LayoutDialog, 1137
Ui_MPerfWidget, 1160
Ui_SettingsWidget, 1204
label_20
Ui::KPerfWidget, 695
Ui_KPerfWidget, 1128
label_21
Ui::KPerfWidget, 695
Ui_KPerfWidget, 1128
label_22
Ui::KPerfWidget, 695
Ui_KPerfWidget, 1128
label_23
Ui::KPerfWidget, 695
Ui_KPerfWidget, 1128
label_24
Ui::KPerfWidget, 695
Ui_KPerfWidget, 1128
label_25
Ui::MPerfWidget, 771
Ui_MPerfWidget, 1160
label_3
Ui::AnimSettingDialog, 94
Ui::ExtraSettingsWidget, 167
Ui::KbLightWidget, 458
Ui::KbWidget, 595
Ui::KPerfWidget, 695
Ui::MPerfWidget, 771
Ui::SettingsWidget, 1027
Ui_AnimSettingDialog, 1047
Ui_ExtraSettingsWidget, 1059
Ui_KbLightWidget, 1098
Ui_KbWidget, 1111
Ui_KPerfWidget, 1128
Ui_MPerfWidget, 1160
Ui_SettingsWidget, 1204
label_4
Ui::AnimSettingDialog, 94
Ui::FwUpgradeDialog, 194
Ui::KbWidget, 595
Ui::KPerfWidget, 696
Ui::MPerfWidget, 771
Ui_AnimSettingDialog, 1047
Ui_FwUpgradeDialog, 1068
Ui_KbWidget, 1111
Ui_KPerfWidget, 1128
Ui_MPerfWidget, 1160
label_5
Ui::AnimSettingDialog, 94
Ui::ExtraSettingsWidget, 167
Ui::KbAnimWidget, 341
Ui::KPerfWidget, 696
Ui::MPerfWidget, 771
Ui::SettingsWidget, 1027
Ui_AnimSettingDialog, 1047
Ui_ExtraSettingsWidget, 1060
Ui_KbAnimWidget, 1083
Ui_KPerfWidget, 1128
Ui_MPerfWidget, 1160
Ui_SettingsWidget, 1204
label_6
Ui::AnimSettingDialog, 94
Ui::ExtraSettingsWidget, 167
Ui::KbAnimWidget, 341
Ui::KPerfWidget, 696
Ui::MPerfWidget, 771
Ui::SettingsWidget, 1028
Ui_AnimSettingDialog, 1047
Ui_ExtraSettingsWidget, 1060
Ui_KbAnimWidget, 1083
Ui_KPerfWidget, 1128
Ui_MPerfWidget, 1160
Ui_SettingsWidget, 1204
label_7
Ui::AnimSettingDialog, 94
Ui::KbAnimWidget, 341
Ui::KPerfWidget, 696
Ui::MPerfWidget, 771
Ui_AnimSettingDialog, 1047
Ui_KbAnimWidget, 1083
Ui_KPerfWidget, 1129
Ui_MPerfWidget, 1160
label_8
Ui::AnimSettingDialog, 94
Ui::KPerfWidget, 696
Ui::MPerfWidget, 772
Ui::SettingsWidget, 1028
Ui_AnimSettingDialog, 1048
Ui_KPerfWidget, 1129
Ui_MPerfWidget, 1161
Ui_SettingsWidget, 1205
label_9
Ui::AnimSettingDialog, 94
Ui::KbWidget, 595
Ui_AnimSettingDialog, 1048
Ui_KbWidget, 1111
lastAutoSave
Kb, 304
KbWidget, 587
lastCheck
KbFirmware, 420
lastCmd
KbBind, 391

lastDuration
AnimSettingDialog, 117
lastFinished
KbFirmware, 420
lastFrame
AnimScript, 82
lastFrameSignal
KbLight, 450
lastGlobalRemapTime
KbBind, 391
lastKnown
media_linux.cpp, 1327
lastMappedDirectoryEntry
QuaZipPrivate, 932
lastSharedDimming
extrasettingswidget.cpp, 1249
Layout
KeyMap, 633
layout
Kb, 280
KeyMap, 649
layoutBox
Ui::SettingsWidget, 1028
Ui_SettingsWidget, 1205
layoutChanged
KbBind, 375
LayoutDialog, 717
∼LayoutDialog, 719
LayoutDialog, 719
LayoutDialog, 719
selected, 720
ui, 720
layoutNames
KeyMap, 649
layoutWidget
Ui::RebindWidget, 957
Ui_RebindWidget, 1187
layoutWidget1
Ui::RebindWidget, 957
Ui_RebindWidget, 1187
layoutWidget2
Ui::RebindWidget, 957
Ui_RebindWidget, 1187
layoutWidget_2
Ui::RebindWidget, 957
Ui_RebindWidget, 1188
lbl_macro
Ui::RebindWidget, 958
Ui_RebindWidget, 1188
license
AnimScript, 68
liftHeight
KbPerf, 517
light
KbAnimWidget, 360
KbBind, 376
KbLightWidget, 471
KbMode, 492
KbPerf, 517
light100Color
KbPerf, 533
lightAction
KeyAction, 620
lightBox
Ui::KPerfWidget, 696
Ui::RebindWidget, 958
Ui_KPerfWidget, 1129
Ui_RebindWidget, 1188
lightButton
Ui::RebindWidget, 958
Ui_RebindWidget, 1188
lightColor1
Ui::KPerfWidget, 696
Ui_KPerfWidget, 1129
lightColor2
Ui::KPerfWidget, 696
Ui_KPerfWidget, 1129
lightColor3
Ui::KPerfWidget, 696
Ui_KPerfWidget, 1129
lightIndicator
KbPerf, 518
lightTab
Ui::KbWidget, 595
Ui_KbWidget, 1111
lightWidget
Ui::KbWidget, 595
Ui_KbWidget, 1111
lightWrapBox
Ui::RebindWidget, 958
Ui_RebindWidget, 1188
line
Ui::AnimSettingDialog, 95
Ui::ExtraSettingsWidget, 167
Ui::KbBindWidget, 397
Ui::KbLightWidget, 458
Ui::KPerfWidget, 697
Ui::MPerfWidget, 772
Ui::SettingsWidget, 1028
Ui_AnimSettingDialog, 1048
Ui_ExtraSettingsWidget, 1060
Ui_KbBindWidget, 1090
Ui_KbLightWidget, 1098
Ui_KPerfWidget, 1129
Ui_MPerfWidget, 1161
Ui_SettingsWidget, 1205
line_2
Ui::AnimSettingDialog, 95
Ui::ExtraSettingsWidget, 168
Ui::KbBindWidget, 397
Ui::KbLightWidget, 458
Ui::KPerfWidget, 697
Ui::MPerfWidget, 772
Ui::SettingsWidget, 1028
Ui_AnimSettingDialog, 1048
Ui_ExtraSettingsWidget, 1060

Ui_KbBindWidget, 1090
Ui_KbLightWidget, 1099
Ui_KPerfWidget, 1129
Ui_MPerfWidget, 1161
Ui_SettingsWidget, 1205
line_3
Ui::ExtraSettingsWidget, 168
Ui::KPerfWidget, 697
Ui::MPerfWidget, 772
Ui::SettingsWidget, 1028
Ui_ExtraSettingsWidget, 1060
Ui_KPerfWidget, 1129
Ui_MPerfWidget, 1161
Ui_SettingsWidget, 1205
linkedlist_data_s, 1460
linkedlist_datablock_internal_s, 1459
linkedlist_data
zip.c, 1465
linkedlist_datablock_internal
zip.c, 1465
list
AnimScript, 68
load
AnimScript, 69
Kb, 281
KbBind, 376
KbLight, 437
KbPerf, 519
LoadCentralDirectoryRecord
zip.c, 1469
loadGlobalRemap
KbBind, 378
local
unzip.c, 1391
zip.c, 1463
locale
KeyMap, 650
lockAction
KeyAction, 620
lockBox
Ui::KPerfWidget, 697
Ui::RebindWidget, 958
Ui_KPerfWidget, 1130
Ui_RebindWidget, 1188
lockButton
Ui::RebindWidget, 958
Ui_RebindWidget, 1188
lockColorOff
Ui::KPerfWidget, 697
Ui_KPerfWidget, 1130
lockColorOn
Ui::KPerfWidget, 697
Ui_KPerfWidget, 1130
lockMutex
ckbsettings.cpp, 1240
ckbsettingswriter.cpp, 1244
lockMutexCache
ckbsettings.cpp, 1240
ckbsettingswriter.cpp, 1244
lockMutexStatic
ckbsettings.cpp, 1241
ckbsettingswriter.cpp, 1244
lockMutexStatic2
ckbsettings.cpp, 1241
ckbsettingswriter.cpp, 1244
loginItemBox
Ui::SettingsWidget, 1028
Ui_SettingsWidget, 1205

M65
KeyMap, 634
M65_HEIGHT
keymap.cpp, 1290
M65_WIDTH
keymap.cpp, 1290
M65Keys
keymap.cpp, 1303
m65Overlay
keywidget.cpp, 1309
MACRO
KbPerf, 503
MAX_FILE_NAME_LENGTH
QuaZip, 830
MEDHIGH
KbPerf, 502
MEDIUM
KbPerf, 502
MODE
KbPerf, 503
MPerfWidget::DpiUi, 778, 1333
MUTE
KbPerf, 503
MUTED
media.h, 1324
MX
KeyMap, 633
M_PI
ckb-anim.h, 1231
MAKEULONG64
zip.c, 1463
MAX_DIM
KbLight, 450
mAccelBox
Ui::ExtraSettingsWidget, 168
Ui_ExtraSettingsWidget, 1060
MODE_NEXT
KeyAction, 629
MODE_NEXT_WRAP
KeyAction, 629
MODE_PREV
KeyAction, 629
MODE_PREV_WRAP
KeyAction, 629
mPerfTab
Ui::KbWidget, 596
Ui_KbWidget, 1112
MPerfWidget, 776

~MPerfWidget, 781
_xyLink, 791
boxXChanged, 781
boxXMapper, 791
boxYChanged, 782
boxYMapper, 791
buttonMapper1, 791
buttonMapper2, 791
colorChanged, 783
colorClicked, 784
colorLink, 792
DPI_COUNT, 792
DPI_MAX, 792
DPI_MIN, 792
enableChanged, 784
enableMapper, 792
isSetting, 792
MPerfWidget, 779
MPerfWidget, 779
on_aSnapBox_clicked, 785
on_copyButton_clicked, 785
on_indicBox_clicked, 786
on_lHeightBox_activated, 786
on_spinBox_valueChanged, 787
on_xyBox_clicked, 787
perf, 792
profile, 792
setPerf, 788
sliderXMapper, 792
sliderXMoved, 789
sliderYMapper, 792
sliderYMoved, 790
stages, 793
ui, 793
mPerfWidget
Ui::KbWidget, 596
Ui_KbWidget, 1112
macReader
RebindWidget, 999
macroAction
KeyAction, 620
macroBox
MacroReaderThread, 728
Ui::KPerfWidget, 697
Ui_KPerfWidget, 1130
macroColorOff
Ui::KPerfWidget, 697
Ui_KPerfWidget, 1130
macroColorOn
Ui::KPerfWidget, 697
Ui_KPerfWidget, 1130
macroContent
KeyAction, 621
macroDelay
Kb, 282
macroDisplay
KeyAction, 621
macroFullLine
KeyAction, 622
macroLine
KeyAction, 623
macroNumber
Kb, 304
MacroReaderThread, 728
macroPath
Kb, 305
MacroReaderThread, 728
MacroReader, 720
~MacroReader, 723
MacroReader, 723
macroReaderThread, 724
MacroReader, 723
startWorkInAThread, 723
MacroReaderThread, 724
macroBox, 728
macroNumber, 728
macroPath, 728
MacroReaderThread, 726
macroText, 728
MacroReaderThread, 726
readMacro, 727
run, 727
macroReaderThread
MacroReader, 724
macroTab
Ui::RebindWidget, 958
Ui_RebindWidget, 1188
macroText
MacroReaderThread, 728
macroTiming
KeyAction, 623
main
ckb-anim.h, 1236
main.cpp, 1318
main.cpp
CommandLineBackground, 1316
CommandLineClose, 1316
CommandLineError, 1316
CommandLineHelpRequested, 1316
CommandLineOK, 1316
CommandLineVersionRequested, 1316
main.cpp
appShare, 1317
CommandLineParseResults, 1316
isRunning, 1317
main, 1318
parseCommandLine, 1319
pidActive, 1321
SHMEM_SIZE, 1316
SHMEM_SIZE_V015, 1316
MainWindow, 732
~MainWindow, 737
addDevice, 737
checkFwUpdates, 738
cleanup, 739
closeAction, 746

closeEvent, 739
iconClicked, 740
kbWidgets, 746
MainWindow, 735
mainWindow, 746
MainWindow, 735
quitApp, 741
removeDevice, 741
restoreAction, 746
settingsWidget, 747
SettingsWidget, 1017
showFwUpdateNotification, 742
showWindow, 742
stateChange, 743
timerTick, 744
toggleTrayIcon, 745
trayIcon, 747
trayIconMenu, 747
ui, 747
updateVersion, 745
mainWindow
MainWindow, 746
mainwindow.cpp
appShare, 1322
configLabel, 1322
daemonDialogText, 1322
devpath, 1322
makeIcon
GradientDialog, 216
makeStops
GradientDialogWidget, 227
map
KbAnim, 320, 321
KbBind, 378, 379
KbLight, 438, 439
KeyMap, 651
KeyWidget, 663, 664
map_last
kbperf.cpp, 1277
matches
Kb, 283
mb2Box
Ui::RebindWidget, 958
Ui_RebindWidget, 1189
mb2Button
Ui::RebindWidget, 959
Ui_RebindWidget, 1189
mbBox
Ui::RebindWidget, 959
Ui_RebindWidget, 1189
mbButton
Ui::RebindWidget, 959
Ui_RebindWidget, 1189
mdAdd
QuaZip, 830
mdAppend
QuaZip, 830
mdCreate
QuaZip, 830
mdNotOpen
QuaZip, 830
mdUnzip
QuaZip, 830
media.h
MUTED, 1324
UNKNOWN, 1324
UNMUTED, 1324
media.h
ENUM_C, 1324
ENUM_END_C, 1324
EXTERN_C, 1324
getMuteState, 1325
muteState, 1324
media_linux.cpp
getMuteState, 1326
lastKnown, 1327
media_mac.m
disableAppNap, 1327
getMuteState, 1328
mediaBox
Ui::RebindWidget, 959
Ui_RebindWidget, 1189
mediaButton
Ui::RebindWidget, 959
Ui_RebindWidget, 1189
mediaKeys
RebindWidget, 999
method
QuaZipFileInfo, 903
QuaZipFileInfo64, 910
minX
AnimScript, 83
minY
AnimScript, 83
moc_animadddialog.cpp
QT_MOC_LITERAL, 1526
qt_meta_data_AnimAddDialog, 1526
qt_meta_stringdata_AnimAddDialog, 1527
moc_animscript.cpp
QT_MOC_LITERAL, 1529
qt_meta_data_AnimScript, 1529
qt_meta_stringdata_AnimScript, 1530
moc_animsettingdialog.cpp
QT_MOC_LITERAL, 1531
qt_meta_data_AnimSettingDialog, 1532
qt_meta_stringdata_AnimSettingDialog, 1532
moc_ckbsettingswriter.cpp
QT_MOC_LITERAL, 1534
qt_meta_data_CkbSettingsWriter, 1534
qt_meta_stringdata_CkbSettingsWriter, 1535
moc_colorbutton.cpp
QT_MOC_LITERAL, 1536
qt_meta_data_ColorButton, 1537
qt_meta_stringdata_ColorButton, 1537
moc_extrasettingswidget.cpp
QT_MOC_LITERAL, 1539

qt_meta_data_ExtraSettingsWidget, 1539
qt_meta_stringdata_ExtraSettingsWidget, 1539
moc_fwupgradedialog.cpp
QT_MOC_LITERAL, 1542
qt_meta_data_FwUpgradeDialog, 1542
qt_meta_stringdata_FwUpgradeDialog, 1543
moc_gradientbutton.cpp
QT_MOC_LITERAL, 1545
qt_meta_data_GradientButton, 1545
qt_meta_stringdata_GradientButton, 1546
moc_gradientdialog.cpp
QT_MOC_LITERAL, 1547
qt_meta_data_GradientDialog, 1548
qt_meta_stringdata_GradientDialog, 1548
moc_gradientdialogwidget.cpp
QT_MOC_LITERAL, 1550
qt_meta_data_GradientDialogWidget, 1550
qt_meta_stringdata_GradientDialogWidget, 1551
moc_kb.cpp
QT_MOC_LITERAL, 1553
qt_meta_data_Kb, 1553
qt_meta_stringdata_Kb, 1553
moc_kbanim.cpp
QT_MOC_LITERAL, 1555
qt_meta_data_KbAnim, 1555
qt_meta_stringdata_KbAnim, 1556
moc_kbanimwidget.cpp
QT_MOC_LITERAL, 1558
qt_meta_data_KbAnimWidget, 1558
qt_meta_stringdata_KbAnimWidget, 1558
moc_kbbind.cpp
QT_MOC_LITERAL, 1560
qt_meta_data_KbBind, 1560
qt_meta_stringdata_KbBind, 1560
moc_kbbindwidget.cpp
QT_MOC_LITERAL, 1562
qt_meta_data_KbBindWidget, 1563
qt_meta_stringdata_KbBindWidget, 1563
moc_kbfirmware.cpp
QT_MOC_LITERAL, 1565
qt_meta_data_KbFirmware, 1565
qt_meta_stringdata_KbFirmware, 1566
moc_kblight.cpp
QT_MOC_LITERAL, 1568
qt_meta_data_KbLight, 1568
qt_meta_stringdata_KbLight, 1569
moc_kblightwidget.cpp
QT_MOC_LITERAL, 1571
qt_meta_data_KbLightWidget, 1571
qt_meta_stringdata_KbLightWidget, 1571
moc_kbmanager.cpp
QT_MOC_LITERAL, 1573
qt_meta_data_KbManager, 1573
qt_meta_stringdata_KbManager, 1573
moc_kbmode.cpp
QT_MOC_LITERAL, 1575
qt_meta_data_KbMode, 1576
qt_meta_stringdata_KbMode, 1576
moc_kbperf.cpp
QT_MOC_LITERAL, 1578
qt_meta_data_KbPerf, 1578
qt_meta_stringdata_KbPerf, 1579
moc_kbprofile.cpp
QT_MOC_LITERAL, 1580
qt_meta_data_KbProfile, 1581
qt_meta_stringdata_KbProfile, 1581
moc_kbprofiledialog.cpp
QT_MOC_LITERAL, 1583
qt_meta_data_KbProfileDialog, 1583
qt_meta_stringdata_KbProfileDialog, 1584
moc_kbwidget.cpp
QT_MOC_LITERAL, 1586
qt_meta_data_KbWidget, 1586
qt_meta_stringdata_KbWidget, 1586
moc_keyaction.cpp
QT_MOC_LITERAL, 1588
qt_meta_data_KeyAction, 1588
qt_meta_stringdata_KeyAction, 1589
moc_keywidget.cpp
QT_MOC_LITERAL, 1590
qt_meta_data_KeyWidget, 1591
qt_meta_stringdata_KeyWidget, 1591
moc_kperfwidget.cpp
QT_MOC_LITERAL, 1593
qt_meta_data_KPerfWidget, 1593
qt_meta_stringdata_KPerfWidget, 1594
moc_layoutdialog.cpp
QT_MOC_LITERAL, 1596
qt_meta_data_LayoutDialog, 1596
qt_meta_stringdata_LayoutDialog, 1597
moc_macroreader.cpp
QT_MOC_LITERAL, 1599
qt_meta_data_MacroReader, 1599
qt_meta_data_MacroReaderThread, 1600
qt_meta_stringdata_MacroReader, 1600
qt_meta_stringdata_MacroReaderThread, 1600
moc_mainwindow.cpp
QT_MOC_LITERAL, 1602
qt_meta_data_MainWindow, 1602
qt_meta_stringdata_MainWindow, 1602
moc_modeselectdialog.cpp
QT_MOC_LITERAL, 1604
qt_meta_data_ModeSelectDialog, 1604
qt_meta_stringdata_ModeSelectDialog, 1605
moc_mperfwidget.cpp
QT_MOC_LITERAL, 1606
qt_meta_data_MPerfWidget, 1607
qt_meta_stringdata_MPerfWidget, 1607
moc_quagzipfile.cpp
QT_MOC_LITERAL, 1609
qt_meta_data_QuaGzipFile, 1609
qt_meta_stringdata_QuaGzipFile, 1610
moc_quaziodevice.cpp
QT_MOC_LITERAL, 1611
qt_meta_data_QuaZIODevice, 1612
qt_meta_stringdata_QuaZIODevice, 1612

moc_quazipfile.cpp
QT_MOC_LITERAL, 1613
qt_meta_data_QuaZipFile, 1614
qt_meta_stringdata_QuaZipFile, 1614
moc_rebindwidget.cpp
QT_MOC_LITERAL, 1616
qt_meta_data_RebindWidget, 1616
qt_meta_stringdata_RebindWidget, 1616
moc_rlistwidget.cpp
QT_MOC_LITERAL, 1618
qt_meta_data_RListWidget, 1618
qt_meta_stringdata_RListWidget, 1618
moc_settingswidget.cpp
QT_MOC_LITERAL, 1620
qt_meta_data_SettingsWidget, 1621
qt_meta_stringdata_SettingsWidget, 1621
modBox
Ui::RebindWidget, 959
Ui_RebindWidget, 1189
modButton
Ui::RebindWidget, 959
Ui_RebindWidget, 1189
modKeys
RebindWidget, 999
settingswidget.cpp, 1632
modNames
settingswidget.cpp, 1633
Mode
KbAnim, 309
QuaZip, 830
mode
KbAnim, 321
QuaZipPrivate, 932
mode2Raw
KPerfWidget, 706
modeAction
KeyAction, 624
modeBox
Ui::KPerfWidget, 698
Ui::RebindWidget, 960
Ui_KPerfWidget, 1130
Ui_RebindWidget, 1189
modeButton
Ui::RebindWidget, 960
Ui_RebindWidget, 1190
modeChanged
Kb, 283
KbWidget, 570
modeColorOff
Ui::KPerfWidget, 698
Ui_KPerfWidget, 1130
modeColorOn
Ui::KPerfWidget, 698
Ui_KPerfWidget, 1130
modeCount
KbProfile, 543
modeIcon
KbWidget, 572
ModeList
KbProfile, 536
modeList
Ui::ModeSelectDialog, 751
Ui_ModeSelectDialog, 1144
modeParent
KbBind, 379
KbPerf, 521
modeRenamed
Kb, 284
ModeSelectDialog, 751
∼ModeSelectDialog, 753
_modeList, 755
ModeSelectDialog, 753
ModeSelectDialog, 753
on_selAllButton_clicked, 754
on_selNoneButton_clicked, 754
selection, 754
ui, 755
modeUpdate
KbWidget, 574
modeWrapBox
Ui::RebindWidget, 960
Ui_RebindWidget, 1190
Model
KeyMap, 633
model
Kb, 283
KeyMap, 651
modelHeight
KeyMap, 651
modelWidth
KeyMap, 652
modes
KbProfile, 543, 544
modesList
Ui::KbWidget, 596
Ui_KbWidget, 1111
modesList_reordered
KbWidget, 573
modified
UsbId, 1212
modifiedString
UsbId, 1210, 1211
monoRgb
kblight.cpp, 1270
keywidget.cpp, 1308
monochrome
Kb, 305
KeyWidget, 664, 665
mouseAccel
Kb, 284
mouseCurrentX
KeyWidget, 678
mouseCurrentY
KeyWidget, 678
mouseDownMode
KeyWidget, 678

mouseDownX
KeyWidget, 679
mouseDownY
KeyWidget, 679
mouseExtKeys
RebindWidget, 999
mouseKeys
RebindWidget, 999
mouseMoveEvent
GradientDialogWidget, 228
KeyWidget, 665
mousePressEvent
GradientDialogWidget, 229
KeyWidget, 666
mouseReleaseEvent
GradientDialogWidget, 229
KeyWidget, 667
mouseTab
Ui::RebindWidget, 960
Ui_RebindWidget, 1190
move
KbProfile, 544
moveCurrent
GradientDialogWidget, 230
mperfwidget.cpp
DPI_BASE, 1332
dpiExp, 1330
dpiLog, 1331
dpiRound, 1331
SET_END, 1330
SET_START, 1330
xyLinkPath, 1332
Multiply
KbAnim, 309
muteBox
Ui::KPerfWidget, 698
Ui_KPerfWidget, 1131
muteColorNA
Ui::KPerfWidget, 698
Ui_KPerfWidget, 1131
muteColorOff
Ui::KPerfWidget, 698
Ui_KPerfWidget, 1131
muteColorOn
Ui::KPerfWidget, 698
Ui_KPerfWidget, 1131
muteNAColor
KbPerf, 533
muteState
media.h, 1324

NO
KbFirmware, 412
KeyMap, 633
NO_LAYOUT
KeyMap, 633
NO_MODEL
KeyMap, 634
NONE
KbPerf, 503
KeyWidget, 660
NORMAL
KPerfWidget, 704
KbPerf, 503
KeyAction, 604
NUM
KbPerf, 503
N_LAYOUTS
keymap.cpp, 1290
N_MODELS
keymap.cpp, 1291
NEW_FLAG
KbProfileDialog, 560
KbWidget, 587
NS
keymap.cpp, 1291
name
AnimScript, 72
AnimSettingDialog, 108
GradientDialog::Preset, 794
KbAnim, 321, 322
KbMode, 492, 493
KbProfile, 545
KbWidget, 574
Key, 600
KeyMap, 652
QuaZipFileInfo, 903
QuaZipFileInfo64, 910
QuaZipNewInfo, 924
nameBox
Ui::KbAnimWidget, 341
Ui_KbAnimWidget, 1084
nameFilters
QuaZipDir, 872
needsSave
Kb, 285
KbAnim, 322
KbBind, 379
KbLight, 439
KbMode, 493
KbPerf, 521
KbProfile, 545
networkManager
KbFirmware, 420
newDuration
AnimSettingDialog, 108
newGuid
UsbId, 1211
newId
KbAnim, 323
KbMode, 494
KbProfile, 546
newLabel
Ui::FwUpgradeDialog, 194
Ui_FwUpgradeDialog, 1068
newLayout
KbBindWidget, 401

newMode
Kb, 285, 286
newModified
UsbId, 1211
newProfile
Kb, 286, 287
newSelection
KbBindWidget, 402
KbLightWidget, 464
KeyWidget, 679
noAction
KbBind, 380
KeyAction, 624
noAnimLabel
Ui::KbAnimWidget, 342
Ui_KbAnimWidget, 1084
noReorder
KbAnimWidget, 360
noSelPage
Ui::KbAnimWidget, 342
Ui_KbAnimWidget, 1084
Normal
KbAnim, 309
notifyNumber
Kb, 305
notifyPath
Kb, 305
notifyPathMutex
kb.cpp, 1259
notifyPaths
kb.cpp, 1259
numBox
Ui::KPerfWidget, 698
Ui::RebindWidget, 960
Ui_KPerfWidget, 1131
Ui_RebindWidget, 1190
numButton
Ui::RebindWidget, 960
Ui_RebindWidget, 1190
numColorOff
Ui::KPerfWidget, 698
Ui_KPerfWidget, 1131
numColorOn
Ui::KPerfWidget, 699
Ui_KPerfWidget, 1131
numKeys
RebindWidget, 999

OFF
KbPerf, 503
ON
KbPerf, 503
OF
ioapi.h, 1342
ONE_DAY
animscript.cpp, 1215
OTHER
KbPerf, 533
on_aSnapBox_clicked
MPerfWidget, 785
on_actionButton_clicked
FwUpgradeDialog, 186
on_altBox_activated
SettingsWidget, 1010
on_animBox_activated
AnimAddDialog, 44
on_animBox_currentIndexChanged
RebindWidget, 974
on_animButton_clicked
KbLightWidget, 465
RebindWidget, 974
on_animList_currentItemChanged
KbAnimWidget, 349
on_animList_customContextMenuRequested
KbAnimWidget, 350
on_animList_itemChanged
KbAnimWidget, 351
on_animScanButton_clicked
ExtraSettingsWidget, 173
on_applyButton_clicked
RebindWidget, 975
on_autoFWBox_clicked
SettingsWidget, 1011
on_bgButton_clicked
KbLightWidget, 466
on_blendBox_activated
KbAnimWidget, 351
on_brightnessBox_activated
KbLightWidget, 466
on_brightnessBox_clicked
ExtraSettingsWidget, 174
on_btnClearMacro_clicked
RebindWidget, 975
on_btnStartMacro_clicked
RebindWidget, 976
on_btnStopMacro_clicked
RebindWidget, 976
on_cancelButton_clicked
FwUpgradeDialog, 187
RebindWidget, 977
on_capsBox_activated
SettingsWidget, 1011
on_copyButton_clicked
KbBindWidget, 403
KPerfWidget, 707
MPerfWidget, 785
on_ctrlBox_activated
SettingsWidget, 1011
on_delayBox_clicked
ExtraSettingsWidget, 175
on_delayBox_valueChanged
AnimSettingDialog, 109
on_deleteButton_clicked
KbAnimWidget, 352
on_ditherBox_clicked
ExtraSettingsWidget, 175
on_dpiBox_currentIndexChanged

RebindWidget, 978
on_dpiButton_clicked
RebindWidget, 979
on_extraButton_clicked
SettingsWidget, 1012
on_fnBox_currentIndexChanged
RebindWidget, 979
on_fnButton_clicked
RebindWidget, 979
on_fpsBox_valueChanged
ExtraSettingsWidget, 175
on_fwUpdButton_clicked
KbWidget, 574
on_hwSaveButton_clicked
KbWidget, 576
on_indicBox_clicked
MPerfWidget, 786
on_intensityBox_valueChanged
KPerfWidget, 708
on_keyButton_clicked
KbAnimWidget, 352
on_kpDelayBox_valueChanged
AnimSettingDialog, 110
on_kpReleaseBox_stateChanged
AnimSettingDialog, 110
on_kpRepeatBox_valueChanged
AnimSettingDialog, 111
on_lHeightBox_activated
MPerfWidget, 786
on_layoutBox_activated
SettingsWidget, 1012
on_lightBox_currentIndexChanged
RebindWidget, 979
on_lightButton_clicked
RebindWidget, 980
on_lockBox_currentIndexChanged
RebindWidget, 980
on_lockButton_clicked
RebindWidget, 980
on_loginItemBox_clicked
SettingsWidget, 1013
on_mAccelBox_clicked
ExtraSettingsWidget, 176
on_mb2Box_currentIndexChanged
RebindWidget, 981
on_mb2Button_clicked
RebindWidget, 981
on_mbBox_currentIndexChanged
RebindWidget, 981
on_mbButton_clicked
RebindWidget, 982
on_mediaBox_currentIndexChanged
RebindWidget, 982
on_mediaButton_clicked
RebindWidget, 982
on_modBox_currentIndexChanged
RebindWidget, 982
on_modButton_clicked
RebindWidget, 983
on_modeBox_currentIndexChanged
RebindWidget, 983
on_modeButton_clicked
RebindWidget, 983
on_modesList_currentItemChanged
KbWidget, 577
on_modesList_customContextMenuRequested
KbWidget, 578
on_modesList_itemChanged
KbWidget, 579
on_modesList_itemClicked
KbWidget, 580
on_nameBox_textEdited
KbAnimWidget, 353
on_numBox_currentIndexChanged
RebindWidget, 984
on_numButton_clicked
RebindWidget, 984
on_opacityBox_valueChanged
KbAnimWidget, 354
on_presetBox_activated
AnimAddDialog, 45
on_presetDelete_clicked
GradientDialog, 217
on_presetList_currentItemChanged
GradientDialog, 217
on_presetName_textEdited
GradientDialog, 218
on_presetSave_clicked
GradientDialog, 218
on_previewBox_clicked
AnimAddDialog, 46
on_profileBox_activated
KbWidget, 581
on_profileList_currentItemChanged
KbProfileDialog, 554
on_profileList_customContextMenuRequested
KbProfileDialog, 554
on_profileList_itemChanged
KbProfileDialog, 556
on_profileList_itemClicked
KbProfileDialog, 557
on_programKpBox_textChanged
RebindWidget, 984
on_programKpButton_clicked
RebindWidget, 985
on_programKpSIBox_clicked
RebindWidget, 985
on_programKrBox_textChanged
RebindWidget, 985
on_programKrButton_clicked
RebindWidget, 986
on_programKrSIBox_clicked
RebindWidget, 986
on_propertyButton_clicked
KbAnimWidget, 354
on_pteMacroBox_textChanged

RebindWidget, 986
on_pushButton_clicked
SettingsWidget, 1013
on_rb_delay_asTyped_toggled
RebindWidget, 987
on_rb_delay_default_toggled
RebindWidget, 987
on_rb_delay_no_toggled
RebindWidget, 988
on_repeatBox_valueChanged
AnimSettingDialog, 111
on_resetButton_clicked
KbBindWidget, 404
RebindWidget, 988
on_sAccelBox_clicked
ExtraSettingsWidget, 176
on_sSpeedBox_valueChanged
ExtraSettingsWidget, 176
on_selAllButton_clicked
ModeSelectDialog, 754
on_selNoneButton_clicked
ModeSelectDialog, 754
on_shiftBox_activated
SettingsWidget, 1013
on_showAnimBox_clicked
KbLightWidget, 467
on_spinBox_valueChanged
MPerfWidget, 787
on_stopOpacity_valueChanged
GradientDialog, 219
on_stopPos_valueChanged
GradientDialog, 219
on_tabWidget_currentChanged
KbWidget, 581
on_trayBox_clicked
ExtraSettingsWidget, 177
on_typingBox_currentIndexChanged
RebindWidget, 989
on_typingButton_clicked
RebindWidget, 989
on_unbindButton_clicked
RebindWidget, 989
on_wheelBox_currentIndexChanged
RebindWidget, 989
on_wheelButton_clicked
RebindWidget, 990
on_winBox_activated
SettingsWidget, 1014
on_xyBox_clicked
MPerfWidget, 787
once
AutoRun, 121
opacity
KbAnim, 323
opacityBox
Ui::KbAnimWidget, 342
Ui_KbAnimWidget, 1084
open
KbLight, 440
QuaGzipFile, 817, 818
QuaZIODevice, 824
QuaZip, 849
QuaZipFile, 890–892
operator bool
Key, 598
operator const QHash< QString, Key > &
KeyMap, 653
operator QString
KeyAction, 625
operator=
ColorMap, 156
KbPerf, 522
KeyAction, 625
QuaGzipFile, 818
QuaZip, 852
QuaZipDir, 873
QuaZipFile, 894
operator==
ColorMap, 157
QuaZipDir, 873
orderChanged
RListWidget, 1004
osxLabel
Ui::ExtraSettingsWidget, 168
Ui_ExtraSettingsWidget, 1060
osxLine
Ui::ExtraSettingsWidget, 168
Ui_ExtraSettingsWidget, 1060

p
QuaZip, 859
QuaZipFile, 900
PL
KeyMap, 633
PT_BR
KeyMap, 633
PATCH_COUNT
keymap.cpp, 1291
PROGRAM_PR_INDEF
KeyAction, 629
PROGRAM_PR_KPSTOP
KeyAction, 630
PROGRAM_PR_KRSTOP
KeyAction, 630
PROGRAM_PR_MULTI
KeyAction, 630
PROGRAM_RE_INDEF
KeyAction, 630
PROGRAM_RE_KPSTOP
KeyAction, 630
PROGRAM_RE_MULTI
KeyAction, 630
paintEvent
GradientDialogWidget, 231
KeyWidget, 668
param
AnimScript, 72

paramIterator
AnimScript, 74
parameter
KbAnim, 323, 324
parameters
AnimScript, 73
parseCommandLine
main.cpp, 1319
parseVersionString
KbManager, 480
patch
keymap.cpp, 1291
patchABNT2
keymap.cpp, 1296
patchANSI
keymap.cpp, 1297
patchDE
keymap.cpp, 1303
patchDK
keymap.cpp, 1303
patchDvorak
keymap.cpp, 1303
patchES
keymap.cpp, 1303
patchEU
keymap.cpp, 1304
patchFR
keymap.cpp, 1304
patchISO
keymap.cpp, 1297
patchIT
keymap.cpp, 1304
patchJP
keymap.cpp, 1304
patchJP106
keymap.cpp, 1298
patchJP106fn
keymap.cpp, 1299
patchMX
keymap.cpp, 1304
patchNO
keymap.cpp, 1305
patchPTBR
keymap.cpp, 1305
patchSE
keymap.cpp, 1305
patchnonJP106
keymap.cpp, 1299
path
AnimScript, 74
QuaZipDir, 873
perf
KbBind, 381
KbMode, 495
KPerfWidget, 712
MPerfWidget, 792
permissionsFromExternalAttr
quazipfileinfo.cpp, 1379
pickColor
ColorButton, 146
pickGradient
GradientButton, 198
pidActive
main.cpp, 1321
pollLabel
Ui::KbWidget, 596
Ui_KbWidget, 1112
pollLabel2
Ui::KbWidget, 596
Ui_KbWidget, 1112
pollUpdates
ExtraSettingsWidget, 177
SettingsWidget, 1014
pollrate
Kb, 305
popDpi
KbPerf, 522
pos
QIODevice_descriptor, 797
QuaZipFile, 894
positions
KeyMap, 653
preProgram
KeyAction, 629
prefsPath
gradientdialog.cpp, 1256
Kb, 305
Preset
GradientDialog::Preset, 794
preset
AnimScript, 74
presetBox
Ui::AnimAddDialog, 51
Ui_AnimAddDialog, 1037
presetDelete
Ui::GradientDialog, 208
Ui_GradientDialog, 1074
presetList
Ui::GradientDialog, 208
Ui_GradientDialog, 1075
presetName
Ui::GradientDialog, 208
Ui_GradientDialog, 1075
presetSave
Ui::GradientDialog, 208
Ui_GradientDialog, 1075
PresetValue
AnimScript, 57
presets
AnimScript, 75
GradientDialog, 221
prevMode
Kb, 305
prevProfile
Kb, 305
previewAnim

KbLight, 441
previewBox
Ui::AnimAddDialog, 51
Ui_AnimAddDialog, 1037
previousItems
RListWidget, 1006
printParams
AnimScript, 75
printRGB
KbLight, 442
printurl
ckb-anim.h, 1238
process
AnimScript, 83
processDownload
KbFirmware, 417
profile
KbBindWidget, 408
KPerfWidget, 712
MPerfWidget, 792
RebindWidget, 999
profileAdded
Kb, 288
profileBox
Ui::KbWidget, 596
Ui_KbWidget, 1112
profileChanged
Kb, 288
KbWidget, 582
profileList
Ui::KbProfileDialog, 564
Ui_KbProfileDialog, 1103
profileList_reordered
KbProfileDialog, 557
profileRenamed
Kb, 288
profiles
Kb, 289
progTipLabel
Ui::RebindWidget, 962
Ui_RebindWidget, 1191
programAction
KeyAction, 625
programInfo
KeyAction, 625
programKpBox
Ui::RebindWidget, 960
Ui_RebindWidget, 1190
programKpButton
Ui::RebindWidget, 960
Ui_RebindWidget, 1190
programKpExtra
Ui::RebindWidget, 961
Ui_RebindWidget, 1190
programKpModeBox
Ui::RebindWidget, 961
Ui_RebindWidget, 1191
programKpSIBox
Ui::RebindWidget, 961
Ui_RebindWidget, 1191
programKrBox
Ui::RebindWidget, 961
Ui_RebindWidget, 1191
programKrButton
Ui::RebindWidget, 961
Ui_RebindWidget, 1191
programKrExtra
Ui::RebindWidget, 961
Ui_RebindWidget, 1191
programKrModeBox
Ui::RebindWidget, 961
Ui_RebindWidget, 1191
programKrSIBox
Ui::RebindWidget, 961
Ui_RebindWidget, 1191
programTab
Ui::RebindWidget, 962
Ui_RebindWidget, 1191
progressBar
Ui::FwUpgradeDialog, 195
Ui_FwUpgradeDialog, 1068
propertyButton
Ui::KbAnimWidget, 342
Ui_KbAnimWidget, 1084
pteMacroBox
Ui::RebindWidget, 962
Ui_RebindWidget, 1192
pteMacroComment
Ui::RebindWidget, 962
Ui_RebindWidget, 1192
pteMacroText
Ui::RebindWidget, 962
Ui_RebindWidget, 1192
pushButton
Ui::SettingsWidget, 1028
Ui_SettingsWidget, 1205
pushDpi
KbPerf, 523, 524
pushSniper
KbPerf, 525
pushedDpis
KbPerf, 533
pwd
CkbSettings, 131, 132

q
QuaZipFilePrivate, 915
QuaZipPrivate, 933
qCleanupResources_binary
qrc_binary.cpp, 1622
qCleanupResources_image
qrc_image.cpp, 1625
qCleanupResources_text
qrc_text.cpp, 1627
QColorMap
colormap.h, 1249
QColorMapIterator

colormap.h, 1249
QDialog, 795
QIODevice, 796
QIODevice_descriptor, 797
pos, 797
QIODevice_descriptor, 797
QIODevice_descriptor, 797
qInitResources_binary
qrc_binary.cpp, 1622
qInitResources_image
qrc_image.cpp, 1625
qInitResources_text
qrc_text.cpp, 1628
QListWidget, 798
QMainWindow, 799
QMutableColorMapIterator
colormap.h, 1249
QObject, 800
QPushButton, 801
qRegisterResourceData
qrc_binary.cpp, 1623
qrc_image.cpp, 1626
qrc_text.cpp, 1628
QT_MOC_LITERAL
moc_animadddialog.cpp, 1526
moc_animscript.cpp, 1529
moc_animsettingdialog.cpp, 1531
moc_ckbsettingswriter.cpp, 1534
moc_colorbutton.cpp, 1536
moc_extrasettingswidget.cpp, 1539
moc_fwupgradedialog.cpp, 1542
moc_gradientbutton.cpp, 1545
moc_gradientdialog.cpp, 1547
moc_gradientdialogwidget.cpp, 1550
moc_kb.cpp, 1553
moc_kbanim.cpp, 1555
moc_kbanimwidget.cpp, 1558
moc_kbbind.cpp, 1560
moc_kbbindwidget.cpp, 1562
moc_kbfirmware.cpp, 1565
moc_kblight.cpp, 1568
moc_kblightwidget.cpp, 1571
moc_kbmanager.cpp, 1573
moc_kbmode.cpp, 1575
moc_kbperf.cpp, 1578
moc_kbprofile.cpp, 1580
moc_kbprofiledialog.cpp, 1583
moc_kbwidget.cpp, 1586
moc_keyaction.cpp, 1588
moc_keywidget.cpp, 1590
moc_kperfwidget.cpp, 1593
moc_layoutdialog.cpp, 1596
moc_macroreader.cpp, 1599
moc_mainwindow.cpp, 1602
moc_modeselectdialog.cpp, 1604
moc_mperfwidget.cpp, 1606
moc_quagzipfile.cpp, 1609
moc_quaziodevice.cpp, 1611
moc_quazipfile.cpp, 1613
moc_rebindwidget.cpp, 1616
moc_rlistwidget.cpp, 1618
moc_settingswidget.cpp, 1620
QThread, 802
QUAZIO_INBUFSIZE
quaziodevice.cpp, 1367
QUAZIO_OUTBUFSIZE
quaziodevice.cpp, 1367
QUAZIP_EXPORT
quazip_global.h, 1372
qUnregisterResourceData
qrc_binary.cpp, 1623
qrc_image.cpp, 1626
qrc_text.cpp, 1628
QWidget, 933
qioapi.cpp
call_zopen64, 1349
call_zseek64, 1349
call_ztell64, 1349
fill_qiodevice64_filefunc, 1350
fill_qiodevice_filefunc, 1351
fill_zlib_filefunc64_32_def_from_filefunc32, 1352
qiodevice64_seek_file_func, 1353
qiodevice64_tell_file_func, 1354
qiodevice_close_file_func, 1354
qiodevice_error_file_func, 1355
qiodevice_fakeclose_file_func, 1355
qiodevice_open_file_func, 1356
qiodevice_read_file_func, 1357
qiodevice_seek_file_func, 1357
qiodevice_tell_file_func, 1358
qiodevice_write_file_func, 1358
SEEK_CUR, 1349
SEEK_END, 1349
SEEK_SET, 1349
qiodevice64_seek_file_func
qioapi.cpp, 1353
qiodevice64_tell_file_func
qioapi.cpp, 1354
qiodevice_close_file_func
qioapi.cpp, 1354
qiodevice_error_file_func
qioapi.cpp, 1355
qiodevice_fakeclose_file_func
qioapi.cpp, 1355
qiodevice_open_file_func
qioapi.cpp, 1356
qiodevice_read_file_func
qioapi.cpp, 1357
qiodevice_seek_file_func
qioapi.cpp, 1357
qiodevice_tell_file_func
qioapi.cpp, 1358
qiodevice_write_file_func
qioapi.cpp, 1358
qrc_binary.cpp
qCleanupResources_binary, 1622

qInitResources_binary, 1622
qRegisterResourceData, 1623
qUnregisterResourceData, 1623
qt_resource_data, 1624
qt_resource_name, 1624
qt_resource_struct, 1624
qrc_image.cpp
qCleanupResources_image, 1625
qInitResources_image, 1625
qRegisterResourceData, 1626
qUnregisterResourceData, 1626
qt_resource_data, 1626
qt_resource_name, 1626
qt_resource_struct, 1627
qrc_text.cpp
qCleanupResources_text, 1627
qInitResources_text, 1628
qRegisterResourceData, 1628
qUnregisterResourceData, 1628
qt_resource_data, 1629
qt_resource_name, 1629
qt_resource_struct, 1629
qs_strcmp
colormap.cpp, 1247
qt_meta_stringdata_AnimAddDialog_t, 1525
qt_meta_stringdata_AnimScript_t, 1528
qt_meta_stringdata_AnimSettingDialog_t, 1531
qt_meta_stringdata_CkbSettingsWriter_t, 1533
qt_meta_stringdata_ColorButton_t, 1536
qt_meta_stringdata_ExtraSettingsWidget_t, 1538
qt_meta_stringdata_FwUpgradeDialog_t, 1541
qt_meta_stringdata_GradientButton_t, 1544
qt_meta_stringdata_GradientDialog_t, 1547
qt_meta_stringdata_GradientDialogWidget_t, 1549
qt_meta_stringdata_KPerfWidget_t, 1592
qt_meta_stringdata_Kb_t, 1552
qt_meta_stringdata_KbAnim_t, 1554
qt_meta_stringdata_KbAnimWidget_t, 1557
qt_meta_stringdata_KbBind_t, 1559
qt_meta_stringdata_KbBindWidget_t, 1562
qt_meta_stringdata_KbFirmware_t, 1564
qt_meta_stringdata_KbLight_t, 1567
qt_meta_stringdata_KbLightWidget_t, 1570
qt_meta_stringdata_KbManager_t, 1572
qt_meta_stringdata_KbMode_t, 1575
qt_meta_stringdata_KbPerf_t, 1577
qt_meta_stringdata_KbProfile_t, 1580
qt_meta_stringdata_KbProfileDialog_t, 1582
qt_meta_stringdata_KbWidget_t, 1585
qt_meta_stringdata_KeyAction_t, 1587
qt_meta_stringdata_KeyWidget_t, 1590
qt_meta_stringdata_LayoutDialog_t, 1595
qt_meta_stringdata_MPerfWidget_t, 1606
qt_meta_stringdata_MacroReader_t, 1598
qt_meta_stringdata_MacroReaderThread_t, 1598
qt_meta_stringdata_MainWindow_t, 1602
qt_meta_stringdata_ModeSelectDialog_t, 1603
qt_meta_stringdata_QuaGzipFile_t, 1608
qt_meta_stringdata_QuaZIODevice_t, 1611
qt_meta_stringdata_QuaZipFile_t, 1613
qt_meta_stringdata_RListWidget_t, 1617
qt_meta_stringdata_RebindWidget_t, 1615
qt_meta_stringdata_SettingsWidget_t, 1620
qt_meta_data_AnimAddDialog
moc_animadddialog.cpp, 1526
qt_meta_data_AnimScript
moc_animscript.cpp, 1529
qt_meta_data_AnimSettingDialog
moc_animsettingdialog.cpp, 1532
qt_meta_data_CkbSettingsWriter
moc_ckbsettingswriter.cpp, 1534
qt_meta_data_ColorButton
moc_colorbutton.cpp, 1537
qt_meta_data_ExtraSettingsWidget
moc_extrasettingswidget.cpp, 1539
qt_meta_data_FwUpgradeDialog
moc_fwupgradedialog.cpp, 1542
qt_meta_data_GradientButton
moc_gradientbutton.cpp, 1545
qt_meta_data_GradientDialog
moc_gradientdialog.cpp, 1548
qt_meta_data_GradientDialogWidget
moc_gradientdialogwidget.cpp, 1550
qt_meta_data_KPerfWidget
moc_kperfwidget.cpp, 1593
qt_meta_data_Kb
moc_kb.cpp, 1553
qt_meta_data_KbAnim
moc_kbanim.cpp, 1555
qt_meta_data_KbAnimWidget
moc_kbanimwidget.cpp, 1558
qt_meta_data_KbBind
moc_kbbind.cpp, 1560
qt_meta_data_KbBindWidget
moc_kbbindwidget.cpp, 1563
qt_meta_data_KbFirmware
moc_kbfirmware.cpp, 1565
qt_meta_data_KbLight
moc_kblight.cpp, 1568
qt_meta_data_KbLightWidget
moc_kblightwidget.cpp, 1571
qt_meta_data_KbManager
moc_kbmanager.cpp, 1573
qt_meta_data_KbMode
moc_kbmode.cpp, 1576
qt_meta_data_KbPerf
moc_kbperf.cpp, 1578
qt_meta_data_KbProfile
moc_kbprofile.cpp, 1581
qt_meta_data_KbProfileDialog
moc_kbprofiledialog.cpp, 1583
qt_meta_data_KbWidget
moc_kbwidget.cpp, 1586
qt_meta_data_KeyAction
moc_keyaction.cpp, 1588
qt_meta_data_KeyWidget

moc_keywidget.cpp, 1591
qt_meta_data_LayoutDialog
moc_layoutdialog.cpp, 1596
qt_meta_data_MPerfWidget
moc_mperfwidget.cpp, 1607
qt_meta_data_MacroReader
moc_macroreader.cpp, 1599
qt_meta_data_MacroReaderThread
moc_macroreader.cpp, 1600
qt_meta_data_MainWindow
moc_mainwindow.cpp, 1602
qt_meta_data_ModeSelectDialog
moc_modeselectdialog.cpp, 1604
qt_meta_data_QuaGzipFile
moc_quagzipfile.cpp, 1609
qt_meta_data_QuaZIODevice
moc_quaziodevice.cpp, 1612
qt_meta_data_QuaZipFile
moc_quazipfile.cpp, 1614
qt_meta_data_RListWidget
moc_rlistwidget.cpp, 1618
qt_meta_data_RebindWidget
moc_rebindwidget.cpp, 1616
qt_meta_data_SettingsWidget
moc_settingswidget.cpp, 1621
qt_meta_stringdata_AnimAddDialog
moc_animadddialog.cpp, 1527
qt_meta_stringdata_AnimScript
moc_animscript.cpp, 1530
qt_meta_stringdata_AnimSettingDialog
moc_animsettingdialog.cpp, 1532
qt_meta_stringdata_CkbSettingsWriter
moc_ckbsettingswriter.cpp, 1535
qt_meta_stringdata_ColorButton
moc_colorbutton.cpp, 1537
qt_meta_stringdata_ExtraSettingsWidget
moc_extrasettingswidget.cpp, 1539
qt_meta_stringdata_FwUpgradeDialog
moc_fwupgradedialog.cpp, 1543
qt_meta_stringdata_GradientButton
moc_gradientbutton.cpp, 1546
qt_meta_stringdata_GradientDialog
moc_gradientdialog.cpp, 1548
qt_meta_stringdata_GradientDialogWidget
moc_gradientdialogwidget.cpp, 1551
qt_meta_stringdata_KPerfWidget
moc_kperfwidget.cpp, 1594
qt_meta_stringdata_Kb
moc_kb.cpp, 1553
qt_meta_stringdata_KbAnim
moc_kbanim.cpp, 1556
qt_meta_stringdata_KbAnimWidget
moc_kbanimwidget.cpp, 1558
qt_meta_stringdata_KbBind
moc_kbbind.cpp, 1560
qt_meta_stringdata_KbBindWidget
moc_kbbindwidget.cpp, 1563
qt_meta_stringdata_KbFirmware
moc_kbfirmware.cpp, 1566
qt_meta_stringdata_KbLight
moc_kblight.cpp, 1569
qt_meta_stringdata_KbLightWidget
moc_kblightwidget.cpp, 1571
qt_meta_stringdata_KbManager
moc_kbmanager.cpp, 1573
qt_meta_stringdata_KbMode
moc_kbmode.cpp, 1576
qt_meta_stringdata_KbPerf
moc_kbperf.cpp, 1579
qt_meta_stringdata_KbProfile
moc_kbprofile.cpp, 1581
qt_meta_stringdata_KbProfileDialog
moc_kbprofiledialog.cpp, 1584
qt_meta_stringdata_KbWidget
moc_kbwidget.cpp, 1586
qt_meta_stringdata_KeyAction
moc_keyaction.cpp, 1589
qt_meta_stringdata_KeyWidget
moc_keywidget.cpp, 1591
qt_meta_stringdata_LayoutDialog
moc_layoutdialog.cpp, 1597
qt_meta_stringdata_MPerfWidget
moc_mperfwidget.cpp, 1607
qt_meta_stringdata_MacroReader
moc_macroreader.cpp, 1600
qt_meta_stringdata_MacroReaderThread
moc_macroreader.cpp, 1600
qt_meta_stringdata_MainWindow
moc_mainwindow.cpp, 1602
qt_meta_stringdata_ModeSelectDialog
moc_modeselectdialog.cpp, 1605
qt_meta_stringdata_QuaGzipFile
moc_quagzipfile.cpp, 1610
qt_meta_stringdata_QuaZIODevice
moc_quaziodevice.cpp, 1612
qt_meta_stringdata_QuaZipFile
moc_quazipfile.cpp, 1614
qt_meta_stringdata_RListWidget
moc_rlistwidget.cpp, 1618
qt_meta_stringdata_RebindWidget
moc_rebindwidget.cpp, 1616
qt_meta_stringdata_SettingsWidget
moc_settingswidget.cpp, 1621
qt_resource_data
qrc_binary.cpp, 1624
qrc_image.cpp, 1626
qrc_text.cpp, 1629
qt_resource_name
qrc_binary.cpp, 1624
qrc_image.cpp, 1626
qrc_text.cpp, 1629
qt_resource_struct
qrc_binary.cpp, 1624
qrc_image.cpp, 1627
qrc_text.cpp, 1629
QuaZip

csDefault, 830
csInsensitive, 830
csSensitive, 830
MAX_FILE_NAME_LENGTH, 830
mdAdd, 830
mdAppend, 830
mdCreate, 830
mdNotOpen, 830
mdUnzip, 830
QuaAdler32, 803
calculate, 805
checksum, 806
QuaAdler32, 805
QuaAdler32, 805
reset, 805
update, 806
value, 806
QuaChecksum32, 806
calculate, 808
reset, 808
update, 808
value, 808
QuaCrc32, 809
calculate, 811
checksum, 812
QuaCrc32, 811
QuaCrc32, 811
reset, 811
update, 812
value, 812
QuaGzipFile, 812
~QuaGzipFile, 816
close, 816
d, 819
flush, 817
getFileName, 817
isSequential, 817
open, 817, 818
operator=, 818
QuaGzipFile, 815, 816
QuaGzipFile, 815, 816
readData, 818
setFileName, 818
writeData, 818
QuaZIODevice, 819
~QuaZIODevice, 822
close, 822
d, 826
flush, 823
getIoDevice, 824
isSequential, 824
open, 824
QuaZIODevice, 822
QuaZIODevice, 822
readData, 825
writeData, 826
QuaZip, 827
~QuaZip, 831
CaseSensitivity, 830
close, 832
Constants, 830
convertCaseSensitivity, 833
getComment, 834
getCommentCodec, 834
getCurrentFileInfo, 835, 836
getCurrentFileName, 837
getEntriesCount, 838
getFileInfoList, 839
getFileInfoList64, 839
getFileNameCodec, 840
getFileNameList, 840
getIoDevice, 841
getMode, 841
getUnzFile, 842
getZipError, 843
getZipFile, 844
getZipName, 844
goToFirstFile, 845
goToNextFile, 846
hasCurrentFile, 847
isAutoClose, 847
isDataDescriptorWritingEnabled, 847
isOpen, 848
isZip64Enabled, 848
Mode, 830
open, 849
operator=, 852
p, 859
QuaZip, 831
QuaZipPrivate, 859
QuaZip, 831
QuaZipPrivate, 931
setAutoClose, 852
setComment, 853
setCommentCodec, 853
setCurrentFile, 853
setDataDescriptorWritingEnabled, 855
setDefaultFileNameCodec, 856
setFileNameCodec, 857
setIoDevice, 857
setZip64Enabled, 858
setZipName, 858
QuaZip_getFileInfo
quazip.cpp, 1369, 1370
QuaZipDir, 859
~QuaZipDir, 862
caseSensitivity, 863
cd, 863
cdUp, 865
count, 865
d, 876
dirName, 866
entryInfoList, 866, 867
entryInfoList64, 867, 868
entryList, 868, 869
exists, 869, 870

filePath, 871
filter, 871
isRoot, 872
nameFilters, 872
operator=, 873
operator==, 873
path, 873
QuaZipDir, 862
QuaZipDir, 862
relativeFilePath, 874
setCaseSensitivity, 874
setFilter, 875
setNameFilters, 875
setPath, 875
setSorting, 876
sorting, 876
QuaZipDir_convertInfoList
quazipdir.cpp, 1373, 1374
QuaZipDir_getFileInfo
quazipdir.cpp, 1374
QuaZipFile, 877
~QuaZipFile, 882
atEnd, 882
bytesAvailable, 883
close, 883
csize, 884
getActualFileName, 885
getCaseSensitivity, 886
getFileInfo, 886, 887
getFileName, 888
getZip, 888
getZipError, 888
getZipName, 889
isRaw, 889
isSequential, 890
open, 890–892
operator=, 894
p, 900
pos, 894
QuaZipFile, 880, 881
QuaZipFilePrivate, 899
QuaZipFile, 880, 881
QuaZipFilePrivate, 914
readData, 895
setFileName, 896
setZip, 896
setZipName, 897
size, 897
usize, 898
writeData, 899
QuaZipFileInfo, 900
comment, 902
compressedSize, 902
crc, 902
dateTime, 902
diskNumberStart, 902
externalAttr, 902
extra, 902
flags, 902
getPermissions, 901
internalAttr, 903
method, 903
name, 903
uncompressedSize, 903
versionCreated, 903
versionNeeded, 903
QuaZipFileInfo64, 903
comment, 909
compressedSize, 909
crc, 909
dateTime, 909
diskNumberStart, 909
externalAttr, 909
extra, 910
flags, 910
getNTFSaTime, 905
getNTFScTime, 906
getNTFSmTime, 906
getPermissions, 907
internalAttr, 910
isEncrypted, 908
method, 910
name, 910
toQuaZipFileInfo, 908
uncompressedSize, 910
versionCreated, 910
versionNeeded, 910
QuaZipFilePrivate, 911
~QuaZipFilePrivate, 913
caseSensitivity, 914
crc, 914
fileName, 914
internal, 914
q, 915
QuaZipFile, 914
QuaZipFilePrivate, 912, 913
QuaZipFile, 899
QuaZipFilePrivate, 912, 913
raw, 915
resetZipError, 913
setZipError, 913
uncompressedSize, 915
writePos, 915
zip, 915
zipError, 915
QuaZipNewInfo, 916
comment, 923
dateTime, 923
externalAttr, 923
extraGlobal, 924
extraLocal, 924
internalAttr, 924
name, 924
QuaZipNewInfo, 917, 918
QuaZipNewInfo, 917, 918
setFileDateTime, 919

setFileNTFSTimes, 922
setFileNTFSaTime, 919
setFileNTFScTime, 920
setFileNTFSmTime, 921
setFilePermissions, 922
setPermissions, 923
uncompressedSize, 924
QuaZipNewInfo_init
quazipnewinfo.cpp, 1381
QuaZipNewInfo_setPermissions
quazipnewinfo.cpp, 1382
QuaZipPrivate, 924
addCurrentFileToDirectoryMap, 927
autoClose, 931
clearDirectoryMap, 928
comment, 931
commentCodec, 931
dataDescriptorWritingEnabled, 931
defaultFileNameCodec, 932
directoryCaseInsensitive, 932
directoryCaseSensitive, 932
fileNameCodec, 932
getDefaultFileNameCodec, 929
getFileInfoList, 929
goToFirstUnmappedFile, 930
hasCurrentFile_f, 932
ioDevice, 932
lastMappedDirectoryEntry, 932
mode, 932
q, 933
QuaZip, 931
QuaZipPrivate, 926, 927
QuaZip, 859
QuaZipPrivate, 926, 927
zip64, 933
zipError, 933
zipName, 933
quaziodevice.cpp
QUAZIO_INBUFSIZE, 1367
QUAZIO_OUTBUFSIZE, 1367
quazip.cpp
QuaZip_getFileInfo, 1369, 1370
quazip.h
UNZ_OPENERROR, 1371
quazip_global.h
QUAZIP_EXPORT, 1372
UNUSED, 1373
quazipdir.cpp
QuaZipDir_convertInfoList, 1373, 1374
QuaZipDir_getFileInfo, 1374
quazipfileinfo.cpp
getNTFSTime, 1377
permissionsFromExternalAttr, 1379
quazipnewinfo.cpp
QuaZipNewInfo_init, 1381
QuaZipNewInfo_setPermissions, 1382
setNTFSTime, 1382
quitApp
MainWindow, 741
RGB
KPerfWidget, 704
README.md, 1213
RListWidget, 1001
change, 1003
currentData, 1006
DATA_ROLE, 1006
dragged, 1006
enter, 1003
orderChanged, 1004
previousItems, 1006
RListWidget, 1003
reorderTimer, 1006
rescanItems, 1004
RListWidget, 1003
timerTick, 1005
ROADMAP.md, 1213
raw
QuaZipFilePrivate, 915
raw2Mode
KPerfWidget, 708
rb_delay_asTyped
Ui::RebindWidget, 962
Ui_RebindWidget, 1192
rb_delay_default
Ui::RebindWidget, 962
Ui_RebindWidget, 1192
rb_delay_no
Ui::RebindWidget, 962
Ui_RebindWidget, 1192
rbWidget
Ui::KbBindWidget, 397
Ui_KbBindWidget, 1090
reInit
KbAnim, 325
readAnyFrame
AnimScript, 83
readData
QuaGzipFile, 818
QuaZIODevice, 825
QuaZipFile, 895
readFrame
AnimScript, 83
readMacro
MacroReaderThread, 727
readNotify
Kb, 290
readProcess
AnimScript, 76
RebindWidget, 965
~RebindWidget, 970
applyChanges, 970
bind, 998
convertMacroBox, 972
fnKeys, 999
helpStatus, 973
macReader, 999

mediaKeys, 999
modKeys, 999
mouseExtKeys, 999
mouseKeys, 999
numKeys, 999
on_animBox_currentIndexChanged, 974
on_animButton_clicked, 974
on_applyButton_clicked, 975
on_btnClearMacro_clicked, 975
on_btnStartMacro_clicked, 976
on_btnStopMacro_clicked, 976
on_cancelButton_clicked, 977
on_dpiBox_currentIndexChanged, 978
on_dpiButton_clicked, 979
on_fnBox_currentIndexChanged, 979
on_fnButton_clicked, 979
on_lightBox_currentIndexChanged, 979
on_lightButton_clicked, 980
on_lockBox_currentIndexChanged, 980
on_lockButton_clicked, 980
on_mb2Box_currentIndexChanged, 981
on_mb2Button_clicked, 981
on_mbBox_currentIndexChanged, 981
on_mbButton_clicked, 982
on_mediaBox_currentIndexChanged, 982
on_mediaButton_clicked, 982
on_modBox_currentIndexChanged, 982
on_modButton_clicked, 983
on_modeBox_currentIndexChanged, 983
on_modeButton_clicked, 983
on_numBox_currentIndexChanged, 984
on_numButton_clicked, 984
on_programKpBox_textChanged, 984
on_programKpButton_clicked, 985
on_programKpSIBox_clicked, 985
on_programKrBox_textChanged, 985
on_programKrButton_clicked, 986
on_programKrSIBox_clicked, 986
on_pteMacroBox_textChanged, 986
on_rb_delay_asTyped_toggled, 987
on_rb_delay_default_toggled, 987
on_rb_delay_no_toggled, 988
on_resetButton_clicked, 988
on_typingBox_currentIndexChanged, 989
on_typingButton_clicked, 989
on_unbindButton_clicked, 989
on_wheelBox_currentIndexChanged, 989
on_wheelButton_clicked, 990
profile, 999
RebindWidget, 969
RebindWidget, 969
selection, 999
setBind, 990
setBox, 992
setCorrectRadioButton, 994
setSelection, 995
TAB_ANIM, 1000
TAB_KB, 1000
TAB_MACRO, 1000
TAB_MOUSE, 1000
TAB_PROGRAM, 1000
TAB_SPECIAL, 1000
typingKeys, 1000
ui, 1000
wheelKeys, 1001
rebindwidget.cpp
DPI_CUST_IDX, 1523
DPI_OFFSET, 1523
rebuildBaseMap
KbLight, 443
refreshList
KbAnimWidget, 355
relProgram
KeyAction, 630
relativeFilePath
QuaZipDir, 874
remove
CkbSettings, 133
removeAll
KbProfile, 547
removeCache
CkbSettings, 137
removeDev
FwUpgradeDialog, 187
removeDevice
MainWindow, 741
removeFile
JlCompress, 254
reorderAnims
KbAnimWidget, 356
reorderTimer
RListWidget, 1006
repeatBox
Ui::AnimSettingDialog, 95
Ui_AnimSettingDialog, 1048
repeatKey
KbAnim, 333
repeatLabel
Ui::AnimSettingDialog, 95
Ui_AnimSettingDialog, 1048
repeatLabel_2
Ui::AnimSettingDialog, 95
Ui_AnimSettingDialog, 1048
repeatMsec
AnimScript, 83
KbAnim, 333
repeatTime
KbAnim, 333
repopulate
KbProfileDialog, 558
rescanItems
RListWidget, 1004
reset
QuaAdler32, 805
QuaChecksum32, 808
QuaCrc32, 811

resetAction
KbBind, 382
resetButton
Ui::KbBindWidget, 397
Ui::RebindWidget, 963
Ui_KbBindWidget, 1090
Ui_RebindWidget, 1192
resetIndicators
KbLight, 443
resetParams
KbAnim, 325
resetZipError
QuaZipFilePrivate, 913
restartAnimation
KbLight, 444
restoreAction
MainWindow, 746
retranslateUi
Ui::AnimAddDialog, 48
Ui::AnimSettingDialog, 86
Ui::ExtraSettingsWidget, 160
Ui::FwUpgradeDialog, 191
Ui::GradientDialog, 203
Ui::KbAnimWidget, 335
Ui::KbBindWidget, 393
Ui::KbLightWidget, 452
Ui::KbProfileDialog, 562
Ui::KbWidget, 589
Ui::KPerfWidget, 683
Ui::LayoutDialog, 714
Ui::MainWindow, 730
Ui::ModeSelectDialog, 749
Ui::MPerfWidget, 758
Ui::RebindWidget, 937
Ui::SettingsWidget, 1020
Ui_AnimAddDialog, 1034
Ui_AnimSettingDialog, 1040
Ui_ExtraSettingsWidget, 1053
Ui_FwUpgradeDialog, 1064
Ui_GradientDialog, 1070
Ui_KbAnimWidget, 1078
Ui_KbBindWidget, 1087
Ui_KbLightWidget, 1093
Ui_KbProfileDialog, 1101
Ui_KbWidget, 1105
Ui_KPerfWidget, 1116
Ui_LayoutDialog, 1134
Ui_MainWindow, 1139
Ui_ModeSelectDialog, 1142
Ui_MPerfWidget, 1148
Ui_RebindWidget, 1168
Ui_SettingsWidget, 1197
retrigger
AnimScript, 77
rgbMode
KeyWidget, 673, 674
right
settingswidget.cpp, 1632
run
CkbSettingsWriter, 139
Kb, 294
MacroReaderThread, 727
runningPushIdx
KbPerf, 533

SABRE
KeyMap, 634
SCIMITAR
KeyMap, 634
SCROLL
KbPerf, 503
SE
KeyMap, 633
SET
KeyWidget, 660
SPECIAL
KeyAction, 604
STRAFE
KeyMap, 634
SUBTRACT
KeyWidget, 660
SABRE_HEIGHT
keymap.cpp, 1291
SABRE_WIDTH
keymap.cpp, 1291
sAccelBox
Ui::ExtraSettingsWidget, 168
Ui_ExtraSettingsWidget, 1060
sAuthorLabel
Ui::AnimSettingDialog, 95
Ui_AnimSettingDialog, 1048
SCIM_HEIGHT
keymap.cpp, 1291
SCIM_WIDTH
keymap.cpp, 1291
sDescLabel
Ui::AnimSettingDialog, 95
Ui_AnimSettingDialog, 1048
SEEK_CUR
qioapi.cpp, 1349
zip.c, 1464
SEEK_END
qioapi.cpp, 1349
zip.c, 1464
SEEK_SET
qioapi.cpp, 1349
zip.c, 1464
SET_END
mperfwidget.cpp, 1330
SET_START
mperfwidget.cpp, 1330
SGroup, 1030
∼SGroup, 1032
_settings, 1032
SGroup, 1032
SGroup, 1032
SHMEM_SIZE

main.cpp, 1316
SHMEM_SIZE_V015
main.cpp, 1316
SIZECENTRALDIRITEM
unzip.c, 1391
SIZECENTRALHEADER
zip.c, 1464
SIZEDATA_INDATABLOCK
zip.c, 1464
SIZEZIPLOCALHEADER
unzip.c, 1391
sLicenseLabel
Ui::AnimSettingDialog, 96
Ui_AnimSettingDialog, 1049
SNIPER
KbPerf, 533
sNameLabel
Ui::AnimSettingDialog, 96
Ui_AnimSettingDialog, 1049
sSpeedBox
Ui::ExtraSettingsWidget, 168
Ui_ExtraSettingsWidget, 1061
sSpeedWidget
Ui::ExtraSettingsWidget, 168
Ui_ExtraSettingsWidget, 1061
sToL
kblight.cpp, 1271
sVerLabel
Ui::AnimSettingDialog, 96
Ui_AnimSettingDialog, 1049
sYearLabel
Ui::AnimSettingDialog, 96
Ui_AnimSettingDialog, 1049
sabOverlay
keywidget.cpp, 1309
SabreKeys
keymap.cpp, 1305
save
Kb, 295
KbAnim, 326
KbBind, 383
KbLight, 445
KbMode, 495
KbPerf, 525
KbProfile, 548
saveBlob
FwUpgradeDialog, 187
saveGlobalRemap
KbBind, 384
savePath
FwUpgradeDialog, 188
scan
AnimScript, 78
scanKeyboards
KbManager, 480
scanTimer
KbManager, 482
ScimKeys
keymap.cpp, 1306
scimOverlay
keywidget.cpp, 1309
script
KbAnim, 327
scriptName
KbAnim, 327
scripts
AnimAddDialog, 46
AnimScript, 83
scrollBox
Ui::KPerfWidget, 699
Ui_KPerfWidget, 1131
scrollColorOff
Ui::KPerfWidget, 699
Ui_KPerfWidget, 1131
scrollColorOn
Ui::KPerfWidget, 699
Ui_KPerfWidget, 1132
scrollSpeed
Kb, 296
selAllButton
Ui::ModeSelectDialog, 751
Ui_ModeSelectDialog, 1144
selLabel
Ui::KbLightWidget, 458
Ui_KbLightWidget, 1099
selNoneButton
Ui::ModeSelectDialog, 751
Ui_ModeSelectDialog, 1145
selectAll
KeyWidget, 674
selectLabel
Ui::KbBindWidget, 397
Ui_KbBindWidget, 1091
selected
GradientDialogWidget, 237
LayoutDialog, 720
selectedKeys
KbAnimWidget, 360
selectedOffset
GradientDialogWidget, 237
selectedPos
GradientDialogWidget, 237
selectedStop
GradientDialogWidget, 232
selection
KeyWidget, 679
ModeSelectDialog, 754
RebindWidget, 999
selectionChanged
KeyWidget, 675
selectionColors
GradientDialogWidget, 233
selectionStack
Ui::KbAnimWidget, 342
Ui_KbAnimWidget, 1084
serialLabel

Ui::KbWidget, 596
Ui_KbWidget, 1112
set
CkbSettings, 133
setAction
KbBind, 385
setAnimation
KeyWidget, 675
setAnimationToSelection
KeyWidget, 676
setAutoClose
QuaZip, 852
setBind
KbBindWidget, 405
RebindWidget, 990
setBox
RebindWidget, 992
setCaseSensitivity
QuaZipDir, 874
setComment
QuaZip, 853
setCommentCodec
QuaZip, 853
setCorrectRadioButton
RebindWidget, 994
setCurrent
KbAnimWidget, 357
setCurrentColor
GradientDialogWidget, 233
setCurrentFile
QuaZip, 853
setCurrentMode
Kb, 297, 298
setCurrentProfile
Kb, 299
setDataDescriptorWritingEnabled
QuaZip, 855
setDefaultFileNameCodec
QuaZip, 856
setDuration
AnimScript, 78
setFileDateTime
QuaZipNewInfo, 919
setFileNTFSTimes
QuaZipNewInfo, 922
setFileNTFSaTime
QuaZipNewInfo, 919
setFileNTFScTime
QuaZipNewInfo, 920
setFileNTFSmTime
QuaZipNewInfo, 921
setFileName
QuaGzipFile, 818
QuaZipFile, 896
setFileNameCodec
QuaZip, 857
setFilePermissions
QuaZipNewInfo, 922
setFilter
QuaZipDir, 875
setGlobalRemap
KbBind, 386
setIndicator
KbLight, 446
KbPerf, 526
setIoDevice
QuaZip, 857
setLabel
ColorButton, 146, 147
setLight
KbAnimWidget, 358
KbLightWidget, 467
setMonochrome
KbLightWidget, 468
setNTFSTime
quazipnewinfo.cpp, 1382
setNameFilters
QuaZipDir, 875
setNeedsSave
KbMode, 496
KbProfile, 549
setNeedsUpdate
KbBind, 387
KbMode, 496
KbPerf, 527
KbProfile, 549
setPath
QuaZipDir, 875
setPerf
KPerfWidget, 709
MPerfWidget, 788
setPermissions
QuaZipNewInfo, 923
setPreset
GradientDialog, 219
setSelectedKeys
KbAnimWidget, 359
setSelection
KeyWidget, 676
RebindWidget, 995
setSorting
QuaZipDir, 876
setStatus
SettingsWidget, 1014
setStops
GradientDialogWidget, 234
setValue
CkbSettings, 134
setVersion
SettingsWidget, 1015
setZip
QuaZipFile, 896
setZip64Enabled
QuaZip, 858
setZipError
QuaZipFilePrivate, 913

setZipName
QuaZip, 858
QuaZipFile, 897
settingPath
autorun.cpp, 1220
settingTab
Ui::AnimSettingDialog, 95
Ui_AnimSettingDialog, 1049
settingWidgets
AnimSettingDialog, 117
settingsCacheMutex
ckbsettings.cpp, 1242
ckbsettingswriter.cpp, 1245
settingsGrid
Ui::AnimSettingDialog, 95
Ui_AnimSettingDialog, 1049
settingsMutex
ckbsettings.cpp, 1242
ckbsettingswriter.cpp, 1245
settingsUpdated
KbPerf, 527
SettingsWidget, 1006
∼SettingsWidget, 1010
extra, 1017
MainWindow, 1017
on_altBox_activated, 1010
on_autoFWBox_clicked, 1011
on_capsBox_activated, 1011
on_ctrlBox_activated, 1011
on_extraButton_clicked, 1012
on_layoutBox_activated, 1012
on_loginItemBox_clicked, 1013
on_pushButton_clicked, 1013
on_shiftBox_activated, 1013
on_winBox_activated, 1014
pollUpdates, 1014
setStatus, 1014
setVersion, 1015
SettingsWidget, 1009
SettingsWidget, 1009
showLayoutDialog, 1015
ui, 1017
updateModifiers, 1016
settingsWidget
MainWindow, 747
settingswidget.cpp
devpath, 1632
modKeys, 1632
modNames, 1633
right, 1632
setupUi
Ui::AnimAddDialog, 49
Ui::AnimSettingDialog, 87
Ui::ExtraSettingsWidget, 161
Ui::FwUpgradeDialog, 191
Ui::GradientDialog, 204
Ui::KbAnimWidget, 336
Ui::KbBindWidget, 393
Ui::KbLightWidget, 453
Ui::KbProfileDialog, 563
Ui::KbWidget, 590
Ui::KPerfWidget, 684
Ui::LayoutDialog, 714
Ui::MainWindow, 730
Ui::ModeSelectDialog, 749
Ui::MPerfWidget, 760
Ui::RebindWidget, 940
Ui::SettingsWidget, 1021
Ui_AnimAddDialog, 1034
Ui_AnimSettingDialog, 1041
Ui_ExtraSettingsWidget, 1054
Ui_FwUpgradeDialog, 1064
Ui_GradientDialog, 1071
Ui_KbAnimWidget, 1078
Ui_KbBindWidget, 1087
Ui_KbLightWidget, 1094
Ui_KbProfileDialog, 1101
Ui_KbWidget, 1106
Ui_KPerfWidget, 1117
Ui_LayoutDialog, 1134
Ui_MainWindow, 1139
Ui_ModeSelectDialog, 1143
Ui_MPerfWidget, 1149
Ui_RebindWidget, 1170
Ui_SettingsWidget, 1198
shareDimming
KbLight, 446, 447
shiftBox
Ui::SettingsWidget, 1028
Ui_SettingsWidget, 1205
shiftLabel
Ui::SettingsWidget, 1029
Ui_SettingsWidget, 1205
showAnimBox
Ui::KbLightWidget, 458
Ui_KbLightWidget, 1099
showFirstTab
KbWidget, 583
showFwUpdate
KbWidget, 583
showFwUpdateNotification
MainWindow, 742
showLastTab
KbWidget, 584
showLayoutDialog
SettingsWidget, 1015
showPreview
AnimAddDialog, 46
showWindow
MainWindow, 742
sidelightToggled
KeyWidget, 677
size
QuaZipFile, 897
sliderXMapper
MPerfWidget, 792

sliderXMoved
MPerfWidget, 789
sliderYMapper
MPerfWidget, 792
sliderYMoved
MPerfWidget, 790
sniperDpi
KbPerf, 527, 528
sniperValue
KeyAction, 630
sorting
QuaZipDir, 876
specialInfo
KeyAction, 626
spinBox
Ui::MPerfWidget, 772
Ui_MPerfWidget, 1161
src/ckb/animadddialog.cpp, 1213
src/ckb/animadddialog.h, 1213
src/ckb/animscript.cpp, 1214
src/ckb/animscript.h, 1216
src/ckb/animsettingdialog.cpp, 1217
src/ckb/animsettingdialog.h, 1219
src/ckb/autorun.cpp, 1219
src/ckb/autorun.h, 1220
src/ckb/ckb-anim.h, 1221
src/ckb/ckbsettings.cpp, 1240
src/ckb/ckbsettings.h, 1242
src/ckb/ckbsettingswriter.cpp, 1243
src/ckb/ckbsettingswriter.h, 1245
src/ckb/colorbutton.cpp, 1245
src/ckb/colorbutton.h, 1246
src/ckb/colormap.cpp, 1247
src/ckb/colormap.h, 1248
src/ckb/extrasettingswidget.cpp, 1249
src/ckb/extrasettingswidget.h, 1250
src/ckb/fwupgradedialog.cpp, 1250
src/ckb/fwupgradedialog.h, 1253
src/ckb/gradientbutton.cpp, 1254
src/ckb/gradientbutton.h, 1255
src/ckb/gradientdialog.cpp, 1256
src/ckb/gradientdialog.h, 1256
src/ckb/gradientdialogwidget.cpp, 1257
src/ckb/gradientdialogwidget.h, 1257
src/ckb/kb.cpp, 1258
src/ckb/kb.h, 1259
src/ckb/kbanim.cpp, 1260
src/ckb/kbanim.h, 1262
src/ckb/kbanimwidget.cpp, 1263
src/ckb/kbanimwidget.h, 1263
src/ckb/kbbind.cpp, 1264
src/ckb/kbbind.h, 1265
src/ckb/kbbindwidget.cpp, 1266
src/ckb/kbbindwidget.h, 1266
src/ckb/kbfirmware.cpp, 1267
src/ckb/kbfirmware.h, 1268
src/ckb/kblight.cpp, 1269
src/ckb/kblight.h, 1272
src/ckb/kblightwidget.cpp, 1273
src/ckb/kblightwidget.h, 1273
src/ckb/kbmanager.cpp, 1274
src/ckb/kbmanager.h, 1275
src/ckb/kbmode.cpp, 1276
src/ckb/kbmode.h, 1276
src/ckb/kbperf.cpp, 1277
src/ckb/kbperf.h, 1277
src/ckb/kbprofile.cpp, 1278
src/ckb/kbprofile.h, 1279
src/ckb/kbprofiledialog.cpp, 1279
src/ckb/kbprofiledialog.h, 1280
src/ckb/kbwidget.cpp, 1281
src/ckb/kbwidget.h, 1281
src/ckb/keyaction.cpp, 1282
src/ckb/keyaction.h, 1283
src/ckb/keymap.cpp, 1284
src/ckb/keymap.h, 1306
src/ckb/keywidget.cpp, 1307
src/ckb/keywidget.h, 1309
src/ckb/kperfwidget.cpp, 1310
src/ckb/kperfwidget.h, 1310
src/ckb/layoutdialog.cpp, 1312
src/ckb/layoutdialog.h, 1313
src/ckb/macroreader.cpp, 1314
src/ckb/macroreader.h, 1314
src/ckb/main.cpp, 1315
src/ckb/mainwindow.cpp, 1321
src/ckb/mainwindow.h, 1322
src/ckb/media.h, 1324
src/ckb/media_linux.cpp, 1326
src/ckb/media_mac.m, 1327
src/ckb/modeselectdialog.cpp, 1328
src/ckb/modeselectdialog.h, 1329
src/ckb/mperfwidget.cpp, 1330
src/ckb/mperfwidget.h, 1332
src/ckb/quazip/JlCompress.cpp, 1345
src/ckb/quazip/JlCompress.h, 1347
src/ckb/quazip/crypt.h, 1334
src/ckb/quazip/ioapi.h, 1338
src/ckb/quazip/qioapi.cpp, 1347
src/ckb/quazip/quaadler32.cpp, 1359
src/ckb/quazip/quaadler32.h, 1360
src/ckb/quazip/quachecksum32.h, 1361
src/ckb/quazip/quacrc32.cpp, 1362
src/ckb/quazip/quacrc32.h, 1363
src/ckb/quazip/quagzipfile.cpp, 1364
src/ckb/quazip/quagzipfile.h, 1365
src/ckb/quazip/quaziodevice.cpp, 1366
src/ckb/quazip/quaziodevice.h, 1367
src/ckb/quazip/quazip.cpp, 1368
src/ckb/quazip/quazip.h, 1370
src/ckb/quazip/quazip_global.h, 1371
src/ckb/quazip/quazipdir.cpp, 1373
src/ckb/quazip/quazipdir.h, 1375
src/ckb/quazip/quazipfile.cpp, 1375
src/ckb/quazip/quazipfile.h, 1376
src/ckb/quazip/quazipfileinfo.cpp, 1377

src/ckb/quazip/quazipfileinfo.h, 1379
src/ckb/quazip/quazipnewinfo.cpp, 1380
src/ckb/quazip/quazipnewinfo.h, 1384
src/ckb/quazip/unzip.c, 1385
src/ckb/quazip/unzip.h, 1436
src/ckb/quazip/zip.c, 1456
src/ckb/quazip/zip.h, 1512
src/ckb/rebindwidget.cpp, 1523
src/ckb/rebindwidget.h, 1523
src/ckb/release/moc_animadddialog.cpp, 1524
src/ckb/release/moc_animscript.cpp, 1527
src/ckb/release/moc_animsettingdialog.cpp, 1530
src/ckb/release/moc_ckbsettingswriter.cpp, 1533
src/ckb/release/moc_colorbutton.cpp, 1535
src/ckb/release/moc_extrasettingswidget.cpp, 1537
src/ckb/release/moc_fwupgradedialog.cpp, 1540
src/ckb/release/moc_gradientbutton.cpp, 1543
src/ckb/release/moc_gradientdialog.cpp, 1546
src/ckb/release/moc_gradientdialogwidget.cpp, 1549
src/ckb/release/moc_kb.cpp, 1551
src/ckb/release/moc_kbanim.cpp, 1554
src/ckb/release/moc_kbanimwidget.cpp, 1556
src/ckb/release/moc_kbbind.cpp, 1559
src/ckb/release/moc_kbbindwidget.cpp, 1561
src/ckb/release/moc_kbfirmware.cpp, 1563
src/ckb/release/moc_kblight.cpp, 1566
src/ckb/release/moc_kblightwidget.cpp, 1569
src/ckb/release/moc_kbmanager.cpp, 1572
src/ckb/release/moc_kbmode.cpp, 1574
src/ckb/release/moc_kbperf.cpp, 1576
src/ckb/release/moc_kbprofile.cpp, 1579
src/ckb/release/moc_kbprofiledialog.cpp, 1581
src/ckb/release/moc_kbwidget.cpp, 1585
src/ckb/release/moc_keyaction.cpp, 1587
src/ckb/release/moc_keywidget.cpp, 1589
src/ckb/release/moc_kperfwidget.cpp, 1591
src/ckb/release/moc_layoutdialog.cpp, 1594
src/ckb/release/moc_macroreader.cpp, 1597
src/ckb/release/moc_mainwindow.cpp, 1601
src/ckb/release/moc_modeselectdialog.cpp, 1603
src/ckb/release/moc_mperfwidget.cpp, 1605
src/ckb/release/moc_quagzipfile.cpp, 1608
src/ckb/release/moc_quaziodevice.cpp, 1610
src/ckb/release/moc_quazipfile.cpp, 1612
src/ckb/release/moc_rebindwidget.cpp, 1614
src/ckb/release/moc_rlistwidget.cpp, 1617
src/ckb/release/moc_settingswidget.cpp, 1619
src/ckb/release/qrc_binary.cpp, 1621
src/ckb/release/qrc_image.cpp, 1624
src/ckb/release/qrc_text.cpp, 1627
src/ckb/rlistwidget.cpp, 1630
src/ckb/rlistwidget.h, 1630
src/ckb/settingswidget.cpp, 1631
src/ckb/settingswidget.h, 1633
src/ckb/ui_animadddialog.h, 1634
src/ckb/ui_animsettingdialog.h, 1635
src/ckb/ui_extrasettingswidget.h, 1635
src/ckb/ui_fwupgradedialog.h, 1636
src/ckb/ui_gradientdialog.h, 1637
src/ckb/ui_kbanimwidget.h, 1638
src/ckb/ui_kbbindwidget.h, 1639
src/ckb/ui_kblightwidget.h, 1640
src/ckb/ui_kbprofiledialog.h, 1641
src/ckb/ui_kbwidget.h, 1642
src/ckb/ui_kperfwidget.h, 1643
src/ckb/ui_layoutdialog.h, 1644
src/ckb/ui_mainwindow.h, 1645
src/ckb/ui_modeselectdialog.h, 1646
src/ckb/ui_mperfwidget.h, 1647
src/ckb/ui_rebindwidget.h, 1648
src/ckb/ui_settingswidget.h, 1649
stages
    MPerfWidget, 793
standardMaps
    keymap.cpp, 1306
startWorkInAThread
    MacroReader, 723
stateChange
    MainWindow, 743
statusLabel
    Ui::FwUpgradeDialog, 195
    Ui_FwUpgradeDialog, 1068
stop
    AnimScript, 79
    KbAnim, 328
    KbManager, 482
stopBox
    Ui::GradientDialog, 208
    Ui_GradientDialog, 1075
stopCheck
    AnimSettingDialog, 117
stopColor
    Ui::GradientDialog, 208
    Ui_GradientDialog, 1075
stopCount
    GradientDialogWidget, 235
stopOpacity
    Ui::GradientDialog, 208
    Ui_GradientDialog, 1075
stopPos
    Ui::GradientDialog, 208
    Ui_GradientDialog, 1075
stopPreview
    KbLight, 447
stopRect
    GradientDialogWidget, 235
stopTime
    KbAnim, 333
stopped
    AnimScript, 83
stops
    GradientDialogWidget, 236
storageName
    Key, 599
strLayout
    KeyMap, 654

strModel
KeyMap, 654
strcmpcasenosensitive_internal
unzip.c, 1392
Subtract
KbAnim, 309

TOGGLE
KeyWidget, 660
TAB_ANIM
RebindWidget, 1000
TAB_KB
RebindWidget, 1000
TAB_MACRO
RebindWidget, 1000
TAB_MOUSE
RebindWidget, 1000
TAB_PROGRAM
RebindWidget, 1000
TAB_SPECIAL
RebindWidget, 1000
TRUE
ckb-anim.h, 1231
TRYFREE
unzip.c, 1391
zip.c, 1464
tabWidget
Ui::AnimSettingDialog, 96
Ui::KbWidget, 596
Ui::MainWindow, 732
Ui::RebindWidget, 963
Ui_AnimSettingDialog, 1049
Ui_KbWidget, 1112
Ui_MainWindow, 1141
Ui_RebindWidget, 1192
tableDownload
KbFirmware, 420
tableName
kbfirmware.cpp, 1268
timeGrid
Ui::AnimSettingDialog, 96
Ui_AnimSettingDialog, 1049
timeTab
Ui::AnimSettingDialog, 96
Ui_AnimSettingDialog, 1049
timerTick
MainWindow, 744
RListWidget, 1005
tm_unz_s, 1438
tm_zip_s, 1514
tm_unz
unzip.h, 1446
tm_zip
zip.h, 1518
toQuaZipFileInfo
QuaZipFileInfo64, 908
toStorage
KeyMap, 655
toString
GradientButton, 199
toggleSidelight
KbLightWidget, 469
toggleTrayIcon
MainWindow, 745
trayBox
Ui::ExtraSettingsWidget, 169
Ui_ExtraSettingsWidget, 1061
trayIcon
MainWindow, 747
trayIconMenu
MainWindow, 747
trigger
KbAnim, 329
txtBuffer
Ui::RebindWidget, 963
Ui_RebindWidget, 1193
Type
KeyAction, 604
type
KeyAction, 626
typingBox
Ui::RebindWidget, 963
Ui_RebindWidget, 1193
typingButton
Ui::RebindWidget, 963
Ui_RebindWidget, 1193
typingKeys
RebindWidget, 1000

UNBOUND
KeyAction, 604
UNKNOWN
KbFirmware, 412
media.h, 1324
UNMUTED
media.h, 1324
US
KeyMap, 633
US_DVORAK
KeyMap, 633
UNUSED
quazip_global.h, 1373
UNZ_AUTO_CLOSE
unzip.h, 1444
UNZ_BADZIPFILE
unzip.h, 1444
UNZ_BUFSIZE
unzip.c, 1391
UNZ_CRCERROR
unzip.h, 1444
UNZ_DEFAULT_FLAGS
unzip.h, 1444
UNZ_EOF
unzip.h, 1445
UNZ_ERRNO
unzip.h, 1445
UNZ_INTERNALERROR
unzip.h, 1445

UNZ_MAXFILENAMEINZIP
unzip.c, 1391
UNZ_OK
unzip.h, 1445
UNZ_OPENERROR
quazip.h, 1371
UNZ_PARAMERROR
unzip.h, 1445
Ui, 39
ui
AnimAddDialog, 47
AnimSettingDialog, 117
ExtraSettingsWidget, 178
FwUpgradeDialog, 189
GradientDialog, 222
KbAnimWidget, 360
KbBindWidget, 409
KbLightWidget, 471
KbProfileDialog, 560
KbWidget, 587
KPerfWidget, 712
LayoutDialog, 720
MainWindow, 747
ModeSelectDialog, 755
MPerfWidget, 793
RebindWidget, 1000
SettingsWidget, 1017
Ui::AnimAddDialog, 47
animBox, 50
buttonBox, 51
gridLayout, 51
horizontalLayout, 51
horizontalSpacer, 51
horizontalSpacer_2, 51
label, 51
label_2, 51
presetBox, 51
previewBox, 51
retranslateUi, 48
setupUi, 49
verticalSpacer_2, 52
verticalSpacer_3, 52
Ui::AnimSettingDialog, 84
animName, 91
buttonBox, 91
delayBox, 91
gridLayout, 92
gridLayout_2, 92
horizontalSpacer, 92
horizontalSpacer_2, 92
horizontalSpacer_3, 92
horizontalSpacer_4, 92
horizontalSpacer_5, 92
horizontalSpacer_6, 92
infoTab, 92
kpDelayBox, 92
kpModeStopBox, 93
kpReleaseBox, 93
kpRepeatBox, 93
kpRepeatLabel, 93
kpRepeatLabel_2, 93
label, 93
label_10, 93
label_11, 93
label_12, 93
label_13, 94
label_2, 94
label_3, 94
label_4, 94
label_5, 94
label_6, 94
label_7, 94
label_8, 94
label_9, 94
line, 95
line_2, 95
repeatBox, 95
repeatLabel, 95
repeatLabel_2, 95
retranslateUi, 86
sAuthorLabel, 95
sDescLabel, 95
sLicenseLabel, 96
sNameLabel, 96
sVerLabel, 96
sYearLabel, 96
settingTab, 95
settingsGrid, 95
setupUi, 87
tabWidget, 96
timeGrid, 96
timeTab, 96
verticalSpacer, 96
verticalSpacer_2, 96
verticalSpacer_3, 97
Ui::ExtraSettingsWidget, 158
animCountLabel, 165
animPathLabel, 165
animScanButton, 165
brightnessBox, 165
buttonBox, 165
delayBox, 166
ditherBox, 166
fpsBox, 166
fpsWarnLabel, 166
gridLayout, 166
horizontalLayout, 166
horizontalLayout_2, 166
horizontalSpacer, 166
horizontalSpacer_2, 166
horizontalSpacer_3, 167
label, 167
label_10, 167
label_11, 167
label_2, 167
label_3, 167

label_5, 167
label_6, 167
line, 167
line_2, 168
line_3, 168
mAccelBox, 168
osxLabel, 168
osxLine, 168
retranslateUi, 160
sAccelBox, 168
sSpeedBox, 168
sSpeedWidget, 168
setupUi, 161
trayBox, 169
verticalSpacer, 169
verticalSpacer_10, 169
verticalSpacer_11, 169
verticalSpacer_2, 169
verticalSpacer_3, 169
verticalSpacer_4, 169
verticalSpacer_5, 169
verticalSpacer_6, 169
verticalSpacer_7, 170
verticalSpacer_8, 170
verticalSpacer_9, 170
Ui::FwUpgradeDialog, 189
actionButton, 193
cancelButton, 193
curLabel, 193
devLabel, 193
gridLayout, 194
gridLayout_2, 194
horizontalSpacer, 194
horizontalSpacer_2, 194
horizontalSpacer_3, 194
horizontalSpacer_4, 194
label_2, 194
label_4, 194
newLabel, 194
progressBar, 195
retranslateUi, 191
setupUi, 191
statusLabel, 195
Ui::GradientDialog, 201
buttonBox, 206
gridLayout, 206
gridLayout_2, 206
gridLayout_3, 207
groupBox_2, 207
horizontalLayout, 207
horizontalLayout_2, 207
horizontalSpacer, 207
horizontalSpacer_2, 207
horizontalSpacer_3, 207
label, 207
label_2, 207
presetDelete, 208
presetList, 208
presetName, 208
presetSave, 208
retranslateUi, 203
setupUi, 204
stopBox, 208
stopColor, 208
stopOpacity, 208
stopPos, 208
verticalSpacer, 209
verticalSpacer_2, 209
widget, 209
Ui::KPerfWidget, 679
capsBox, 691
capsColorOff, 691
capsColorOn, 691
copyButton, 691
gridLayout, 691
horizontalLayout, 692
horizontalLayout_2, 692
horizontalSpacer, 692
horizontalSpacer_2, 692
horizontalSpacer_3, 692
horizontalSpacer_4, 692
horizontalSpacer_5, 692
horizontalSpacer_6, 692
horizontalSpacer_7, 692
horizontalSpacer_8, 693
intensityBox, 693
k95Label1, 693
k95Label2, 693
k95Label3, 693
k95Label4, 693
k95Label5, 693
k95Label6, 693
k95Line, 693
k95Spacer, 694
label, 694
label_10, 694
label_11, 694
label_12, 694
label_14, 694
label_15, 694
label_16, 694
label_17, 694
label_18, 695
label_19, 695
label_2, 695
label_20, 695
label_21, 695
label_22, 695
label_23, 695
label_24, 695
label_3, 695
label_4, 696
label_5, 696
label_6, 696
label_7, 696
label_8, 696

lightBox, 696
lightColor1, 696
lightColor2, 696
lightColor3, 696
line, 697
line_2, 697
line_3, 697
lockBox, 697
lockColorOff, 697
lockColorOn, 697
macroBox, 697
macroColorOff, 697
macroColorOn, 697
modeBox, 698
modeColorOff, 698
modeColorOn, 698
muteBox, 698
muteColorNA, 698
muteColorOff, 698
muteColorOn, 698
numBox, 698
numColorOff, 698
numColorOn, 699
retranslateUi, 683
scrollBox, 699
scrollColorOff, 699
scrollColorOn, 699
setupUi, 684
verticalSpacer, 699
verticalSpacer_2, 699
verticalSpacer_3, 699
verticalSpacer_4, 699
Ui::KbAnimWidget, 333
aCopyLabel, 339
aMissingLabel, 339
aMissingPage, 339
aNameLabel, 339
aVerLabel, 340
animList, 339
animPage, 339
blendBox, 340
deleteButton, 340
gridLayout, 340
gridLayout_2, 340
gridLayout_3, 340
gridLayout_4, 340
horizontalLayout, 340
horizontalSpacer, 340
horizontalSpacer_2, 341
horizontalSpacer_3, 341
horizontalSpacer_4, 341
horizontalSpacer_5, 341
keyButton, 341
label_5, 341
label_6, 341
label_7, 341
nameBox, 341
noAnimLabel, 342
noSelPage, 342
opacityBox, 342
propertyButton, 342
retranslateUi, 335
selectionStack, 342
setupUi, 336
verticalLayout, 342
verticalSpacer, 342
verticalSpacer_2, 342
verticalSpacer_3, 342
Ui::KbBindWidget, 391
copyButton, 396
gridLayout, 396
gridLayout_2, 396
horizontalLayout_2, 396
horizontalSpacer, 396
horizontalSpacer_2, 396
keyWidget, 396
label, 396
label_2, 397
line, 397
line_2, 397
rbWidget, 397
resetButton, 397
retranslateUi, 393
selectLabel, 397
setupUi, 393
verticalLayout, 397
verticalSpacer, 397
widget, 397
Ui::KbLightWidget, 450
animButton, 456
animWidget, 456
bgButton, 456
brightnessBox, 456
gridLayout, 456
gridLayout_2, 456
gridLayout_4, 456
horizontalLayout, 457
horizontalLayout_2, 457
horizontalLayout_3, 457
horizontalSpacer_3, 457
horizontalSpacer_4, 457
horizontalSpacer_5, 457
horizontalSpacer_6, 457
keyWidget, 457
label, 457
label_2, 458
label_3, 458
line, 458
line_2, 458
retranslateUi, 452
selLabel, 458
setupUi, 453
showAnimBox, 458
verticalLayout, 458
verticalSpacer_2, 458
verticalSpacer_3, 458

widget, 459
Ui::KbProfileDialog, 560
buttonBox, 564
gridLayout, 564
horizontalSpacer, 564
horizontalSpacer_2, 564
profileList, 564
retranslateUi, 562
setupUi, 563
Ui::KbWidget, 587
bindTab, 593
bindWidget, 593
devTab, 593
fwLabel, 593
fwUpdButton, 593
fwUpdLabel, 594
fwUpdLayout, 594
gridLayout, 594
gridLayout_2, 594
gridLayout_3, 594
gridLayout_4, 594
gridLayout_5, 594
gridLayout_7, 594
horizontalSpacer_2, 594
horizontalSpacer_3, 595
hwSaveButton, 595
kPerfTab, 595
kPerfWidget, 595
label_3, 595
label_4, 595
label_9, 595
lightTab, 595
lightWidget, 595
mPerfTab, 596
mPerfWidget, 596
modesList, 596
pollLabel, 596
pollLabel2, 596
profileBox, 596
retranslateUi, 589
serialLabel, 596
setupUi, 590
tabWidget, 596
verticalSpacer, 596
Ui::LayoutDialog, 712
buttonBox, 716
comboBox, 716
gridLayout, 716
horizontalSpacer, 716
horizontalSpacer_2, 716
label, 716
label_2, 717
retranslateUi, 714
setupUi, 714
verticalSpacer, 717
verticalSpacer_2, 717
verticalSpacer_3, 717
verticalSpacer_4, 717
Ui::MPerfWidget, 755
aSnapBox, 767
copyButton, 767
eBox1, 767
eBox2, 768
eBox3, 768
eBox4, 768
eBox5, 768
gridLayout, 768
horizontalLayout, 768
horizontalLayout_2, 768
horizontalLayout_4, 768
horizontalSpacer, 768
horizontalSpacer_2, 769
horizontalSpacer_3, 769
horizontalSpacer_4, 769
horizontalSpacer_5, 769
horizontalSpacer_6, 769
horizontalSpacer_7, 769
iButton0, 769
iButton1, 769
iButton2, 769
iButton3, 770
iButton4, 770
iButton5, 770
iButtonO, 770
indicBox, 770
lHeightBox, 772
label, 770
label_11, 770
label_12, 770
label_13, 770
label_14, 771
label_15, 771
label_2, 771
label_25, 771
label_3, 771
label_4, 771
label_5, 771
label_6, 771
label_7, 771
label_8, 772
line, 772
line_2, 772
line_3, 772
retranslateUi, 758
setupUi, 760
spinBox, 772
verticalLayout, 772
verticalSpacer, 772
verticalSpacer_2, 772
verticalSpacer_3, 773
xBox0, 773
xBox1, 773
xBox2, 773
xBox3, 773
xBox4, 773
xBox5, 773

xSlider0, 773
xSlider1, 773
xSlider2, 774
xSlider3, 774
xSlider4, 774
xSlider5, 774
xyBox, 774
yBox0, 774
yBox1, 774
yBox2, 774
yBox3, 774
yBox4, 775
yBox5, 775
ySlider0, 775
ySlider1, 775
ySlider2, 775
ySlider3, 775
ySlider4, 775
ySlider5, 775
Ui::MainWindow, 728
actionAbout, 731
actionExit, 731
centralWidget, 731
gridLayout, 732
retranslateUi, 730
setupUi, 730
tabWidget, 732
Ui::ModeSelectDialog, 747
buttonBox, 750
gridLayout, 751
label, 751
modeList, 751
retranslateUi, 749
selAllButton, 751
selNoneButton, 751
setupUi, 749
Ui::RebindWidget, 934
actionTab, 950
animBox, 950
animButton, 950
animKrBox, 951
animOnceBox, 951
animTab, 951
applyButton, 951
btnClearMacro, 951
btnStartMacro, 951
btnStopMacro, 951
cancelButton, 951
delayButtons, 951
dpiBox, 952
dpiButton, 952
dpiCustLabel, 952
dpiCustXBox, 952
dpiCustYBox, 952
dpiWarning, 952
fnBox, 952
fnButton, 952
gridLayout, 953
gridLayout_10, 953
gridLayout_2, 953
gridLayout_3, 953
gridLayout_4, 953
gridLayout_5, 953
gridLayout_8, 953
groupBox_2, 953
groupBox_3, 953
groupBox_4, 954
horizontalLayout, 954
horizontalLayout_2, 954
horizontalLayout_3, 954
horizontalSpacer, 954
horizontalSpacer_10, 954
horizontalSpacer_11, 954
horizontalSpacer_12, 954
horizontalSpacer_13, 954
horizontalSpacer_14, 955
horizontalSpacer_15, 955
horizontalSpacer_16, 955
horizontalSpacer_17, 955
horizontalSpacer_18, 955
horizontalSpacer_19, 955
horizontalSpacer_2, 955
horizontalSpacer_20, 955
horizontalSpacer_21, 955
horizontalSpacer_22, 956
horizontalSpacer_23, 956
horizontalSpacer_24, 956
horizontalSpacer_25, 956
horizontalSpacer_26, 956
horizontalSpacer_3, 956
horizontalSpacer_4, 956
horizontalSpacer_5, 956
horizontalSpacer_6, 956
horizontalSpacer_7, 957
horizontalSpacer_8, 957
horizontalSpacer_9, 957
keyTab, 957
label, 957
layoutWidget, 957
layoutWidget1, 957
layoutWidget2, 957
layoutWidget_2, 957
lbl_macro, 958
lightBox, 958
lightButton, 958
lightWrapBox, 958
lockBox, 958
lockButton, 958
macroTab, 958
mb2Box, 958
mb2Button, 959
mbBox, 959
mbButton, 959
mediaBox, 959
mediaButton, 959
modBox, 959

modButton, 959
modeBox, 960
modeButton, 960
modeWrapBox, 960
mouseTab, 960
numBox, 960
numButton, 960
progTipLabel, 962
programKpBox, 960
programKpButton, 960
programKpExtra, 961
programKpModeBox, 961
programKpSIBox, 961
programKrBox, 961
programKrButton, 961
programKrExtra, 961
programKrModeBox, 961
programKrSIBox, 961
programTab, 962
pteMacroBox, 962
pteMacroComment, 962
pteMacroText, 962
rb_delay_asTyped, 962
rb_delay_default, 962
rb_delay_no, 962
resetButton, 963
retranslateUi, 937
setupUi, 940
tabWidget, 963
txtBuffer, 963
typingBox, 963
typingButton, 963
unbindButton, 963
verticalLayout, 963
verticalLayout_2, 963
verticalSpacer, 964
verticalSpacer_10, 964
verticalSpacer_11, 964
verticalSpacer_2, 964
verticalSpacer_3, 964
verticalSpacer_4, 964
verticalSpacer_5, 964
verticalSpacer_6, 964
verticalSpacer_7, 964
verticalSpacer_8, 965
wheelBox, 965
wheelButton, 965
Ui::SettingsWidget, 1018
altBox, 1025
altLabel, 1025
autoFWBox, 1025
capsBox, 1025
capsLabel, 1026
ctrlBox, 1026
ctrlLabel, 1026
devicesLabel, 1026
extraButton, 1026
gridLayout, 1026
gridLayout_3, 1026
gridLayout_5, 1026
horizontalLayout, 1026
horizontalLayout_2, 1027
horizontalSpacer, 1027
horizontalSpacer_10, 1027
horizontalSpacer_2, 1027
horizontalSpacer_7, 1027
horizontalSpacer_9, 1027
label, 1027
label_2, 1027
label_3, 1027
label_5, 1027
label_6, 1028
label_8, 1028
layoutBox, 1028
line, 1028
line_2, 1028
line_3, 1028
loginItemBox, 1028
pushButton, 1028
retranslateUi, 1020
setupUi, 1021
shiftBox, 1028
shiftLabel, 1029
versionLabel, 1029
verticalSpacer, 1029
verticalSpacer_10, 1029
verticalSpacer_2, 1029
verticalSpacer_3, 1029
verticalSpacer_4, 1029
verticalSpacer_5, 1029
verticalSpacer_6, 1029
verticalSpacer_7, 1030
verticalSpacer_8, 1030
winBox, 1030
winLabel, 1030
Ui_AnimAddDialog, 1033
animBox, 1036
buttonBox, 1036
gridLayout, 1036
horizontalLayout, 1036
horizontalSpacer, 1036
horizontalSpacer_2, 1036
label, 1037
label_2, 1037
presetBox, 1037
previewBox, 1037
retranslateUi, 1034
setupUi, 1034
verticalSpacer_2, 1037
verticalSpacer_3, 1037
Ui_AnimSettingDialog, 1037
animName, 1044
buttonBox, 1044
delayBox, 1045
gridLayout, 1045
gridLayout_2, 1045

horizontalSpacer, 1045
horizontalSpacer_2, 1045
horizontalSpacer_3, 1045
horizontalSpacer_4, 1045
horizontalSpacer_5, 1045
horizontalSpacer_6, 1045
infoTab, 1046
kpDelayBox, 1046
kpModeStopBox, 1046
kpReleaseBox, 1046
kpRepeatBox, 1046
kpRepeatLabel, 1046
kpRepeatLabel_2, 1046
label, 1046
label_10, 1046
label_11, 1047
label_12, 1047
label_13, 1047
label_2, 1047
label_3, 1047
label_4, 1047
label_5, 1047
label_6, 1047
label_7, 1047
label_8, 1048
label_9, 1048
line, 1048
line_2, 1048
repeatBox, 1048
repeatLabel, 1048
repeatLabel_2, 1048
retranslateUi, 1040
sAuthorLabel, 1048
sDescLabel, 1048
sLicenseLabel, 1049
sNameLabel, 1049
sVerLabel, 1049
sYearLabel, 1049
settingTab, 1049
settingsGrid, 1049
setupUi, 1041
tabWidget, 1049
timeGrid, 1049
timeTab, 1049
verticalSpacer, 1050
verticalSpacer_2, 1050
verticalSpacer_3, 1050
Ui_ExtraSettingsWidget, 1050
animCountLabel, 1057
animPathLabel, 1058
animScanButton, 1058
brightnessBox, 1058
buttonBox, 1058
delayBox, 1058
ditherBox, 1058
fpsBox, 1058
fpsWarnLabel, 1058
gridLayout, 1058
horizontalLayout, 1058
horizontalLayout_2, 1059
horizontalSpacer, 1059
horizontalSpacer_2, 1059
horizontalSpacer_3, 1059
label, 1059
label_10, 1059
label_11, 1059
label_2, 1059
label_3, 1059
label_5, 1060
label_6, 1060
line, 1060
line_2, 1060
line_3, 1060
mAccelBox, 1060
osxLabel, 1060
osxLine, 1060
retranslateUi, 1053
sAccelBox, 1060
sSpeedBox, 1061
sSpeedWidget, 1061
setupUi, 1054
trayBox, 1061
verticalSpacer, 1061
verticalSpacer_10, 1061
verticalSpacer_11, 1061
verticalSpacer_2, 1061
verticalSpacer_3, 1061
verticalSpacer_4, 1061
verticalSpacer_5, 1062
verticalSpacer_6, 1062
verticalSpacer_7, 1062
verticalSpacer_8, 1062
verticalSpacer_9, 1062
Ui_FwUpgradeDialog, 1062
actionButton, 1066
cancelButton, 1067
curLabel, 1067
devLabel, 1067
gridLayout, 1067
gridLayout_2, 1067
horizontalSpacer, 1067
horizontalSpacer_2, 1067
horizontalSpacer_3, 1067
horizontalSpacer_4, 1067
label_2, 1068
label_4, 1068
newLabel, 1068
progressBar, 1068
retranslateUi, 1064
setupUi, 1064
statusLabel, 1068
Ui_GradientDialog, 1068
buttonBox, 1073
gridLayout, 1073
gridLayout_2, 1073
gridLayout_3, 1073

groupBox_2, 1074
horizontalLayout, 1074
horizontalLayout_2, 1074
horizontalSpacer, 1074
horizontalSpacer_2, 1074
horizontalSpacer_3, 1074
label, 1074
label_2, 1074
presetDelete, 1074
presetList, 1075
presetName, 1075
presetSave, 1075
retranslateUi, 1070
setupUi, 1071
stopBox, 1075
stopColor, 1075
stopOpacity, 1075
stopPos, 1075
verticalSpacer, 1075
verticalSpacer_2, 1076
widget, 1076
Ui_KPerfWidget, 1113
capsBox, 1124
capsColorOff, 1124
capsColorOn, 1124
copyButton, 1124
gridLayout, 1124
horizontalLayout, 1124
horizontalLayout_2, 1124
horizontalSpacer, 1124
horizontalSpacer_2, 1125
horizontalSpacer_3, 1125
horizontalSpacer_4, 1125
horizontalSpacer_5, 1125
horizontalSpacer_6, 1125
horizontalSpacer_7, 1125
horizontalSpacer_8, 1125
intensityBox, 1125
k95Label1, 1125
k95Label2, 1126
k95Label3, 1126
k95Label4, 1126
k95Label5, 1126
k95Label6, 1126
k95Line, 1126
k95Spacer, 1126
label, 1126
label_10, 1126
label_11, 1127
label_12, 1127
label_14, 1127
label_15, 1127
label_16, 1127
label_17, 1127
label_18, 1127
label_19, 1127
label_2, 1127
label_20, 1128
label_21, 1128
label_22, 1128
label_23, 1128
label_24, 1128
label_3, 1128
label_4, 1128
label_5, 1128
label_6, 1128
label_7, 1129
label_8, 1129
lightBox, 1129
lightColor1, 1129
lightColor2, 1129
lightColor3, 1129
line, 1129
line_2, 1129
line_3, 1129
lockBox, 1130
lockColorOff, 1130
lockColorOn, 1130
macroBox, 1130
macroColorOff, 1130
macroColorOn, 1130
modeBox, 1130
modeColorOff, 1130
modeColorOn, 1130
muteBox, 1131
muteColorNA, 1131
muteColorOff, 1131
muteColorOn, 1131
numBox, 1131
numColorOff, 1131
numColorOn, 1131
retranslateUi, 1116
scrollBox, 1131
scrollColorOff, 1131
scrollColorOn, 1132
setupUi, 1117
verticalSpacer, 1132
verticalSpacer_2, 1132
verticalSpacer_3, 1132
verticalSpacer_4, 1132
Ui_KbAnimWidget, 1076
aCopyLabel, 1081
aMissingLabel, 1081
aMissingPage, 1081
aNameLabel, 1081
aVerLabel, 1082
animList, 1081
animPage, 1082
blendBox, 1082
deleteButton, 1082
gridLayout, 1082
gridLayout_2, 1082
gridLayout_3, 1082
gridLayout_4, 1082
horizontalLayout, 1082
horizontalSpacer, 1083

horizontalSpacer_2, 1083
horizontalSpacer_3, 1083
horizontalSpacer_4, 1083
horizontalSpacer_5, 1083
keyButton, 1083
label_5, 1083
label_6, 1083
label_7, 1083
nameBox, 1084
noAnimLabel, 1084
noSelPage, 1084
opacityBox, 1084
propertyButton, 1084
retranslateUi, 1078
selectionStack, 1084
setupUi, 1078
verticalLayout, 1084
verticalSpacer, 1084
verticalSpacer_2, 1084
verticalSpacer_3, 1085
Ui_KbBindWidget, 1085
copyButton, 1089
gridLayout, 1089
gridLayout_2, 1089
horizontalLayout_2, 1089
horizontalSpacer, 1090
horizontalSpacer_2, 1090
keyWidget, 1090
label, 1090
label_2, 1090
line, 1090
line_2, 1090
rbWidget, 1090
resetButton, 1090
retranslateUi, 1087
selectLabel, 1091
setupUi, 1087
verticalLayout, 1091
verticalSpacer, 1091
widget, 1091
Ui_KbLightWidget, 1091
animButton, 1096
animWidget, 1097
bgButton, 1097
brightnessBox, 1097
gridLayout, 1097
gridLayout_2, 1097
gridLayout_4, 1097
horizontalLayout, 1097
horizontalLayout_2, 1097
horizontalLayout_3, 1097
horizontalSpacer_3, 1098
horizontalSpacer_4, 1098
horizontalSpacer_5, 1098
horizontalSpacer_6, 1098
keyWidget, 1098
label, 1098
label_2, 1098
label_3, 1098
line, 1098
line_2, 1099
retranslateUi, 1093
selLabel, 1099
setupUi, 1094
showAnimBox, 1099
verticalLayout, 1099
verticalSpacer_2, 1099
verticalSpacer_3, 1099
widget, 1099
Ui_KbProfileDialog, 1099
buttonBox, 1102
gridLayout, 1102
horizontalSpacer, 1103
horizontalSpacer_2, 1103
profileList, 1103
retranslateUi, 1101
setupUi, 1101
Ui_KbWidget, 1103
bindTab, 1109
bindWidget, 1109
devTab, 1109
fwLabel, 1109
fwUpdButton, 1109
fwUpdLabel, 1109
fwUpdLayout, 1110
gridLayout, 1110
gridLayout_2, 1110
gridLayout_3, 1110
gridLayout_4, 1110
gridLayout_5, 1110
gridLayout_7, 1110
horizontalSpacer_2, 1110
horizontalSpacer_3, 1110
hwSaveButton, 1111
kPerfTab, 1111
kPerfWidget, 1111
label_3, 1111
label_4, 1111
label_9, 1111
lightTab, 1111
lightWidget, 1111
mPerfTab, 1112
mPerfWidget, 1112
modesList, 1111
pollLabel, 1112
pollLabel2, 1112
profileBox, 1112
retranslateUi, 1105
serialLabel, 1112
setupUi, 1106
tabWidget, 1112
verticalSpacer, 1112
Ui_LayoutDialog, 1132
buttonBox, 1136
comboBox, 1136
gridLayout, 1136

horizontalSpacer, 1136
horizontalSpacer_2, 1136
label, 1136
label_2, 1137
retranslateUi, 1134
setupUi, 1134
verticalSpacer, 1137
verticalSpacer_2, 1137
verticalSpacer_3, 1137
verticalSpacer_4, 1137
Ui_MPerfWidget, 1145
aSnapBox, 1156
copyButton, 1156
eBox1, 1157
eBox2, 1157
eBox3, 1157
eBox4, 1157
eBox5, 1157
gridLayout, 1157
horizontalLayout, 1157
horizontalLayout_2, 1157
horizontalLayout_4, 1157
horizontalSpacer, 1157
horizontalSpacer_2, 1158
horizontalSpacer_3, 1158
horizontalSpacer_4, 1158
horizontalSpacer_5, 1158
horizontalSpacer_6, 1158
horizontalSpacer_7, 1158
iButton0, 1158
iButton1, 1158
iButton2, 1158
iButton3, 1159
iButton4, 1159
iButton5, 1159
iButtonO, 1159
indicBox, 1159
lHeightBox, 1161
label, 1159
label_11, 1159
label_12, 1159
label_13, 1159
label_14, 1160
label_15, 1160
label_2, 1160
label_25, 1160
label_3, 1160
label_4, 1160
label_5, 1160
label_6, 1160
label_7, 1160
label_8, 1161
line, 1161
line_2, 1161
line_3, 1161
retranslateUi, 1148
setupUi, 1149
spinBox, 1161
verticalLayout, 1161
verticalSpacer, 1161
verticalSpacer_2, 1161
verticalSpacer_3, 1162
xBox0, 1162
xBox1, 1162
xBox2, 1162
xBox3, 1162
xBox4, 1162
xBox5, 1162
xSlider0, 1162
xSlider1, 1162
xSlider2, 1163
xSlider3, 1163
xSlider4, 1163
xSlider5, 1163
xyBox, 1163
yBox0, 1163
yBox1, 1163
yBox2, 1163
yBox3, 1163
yBox4, 1164
yBox5, 1164
ySlider0, 1164
ySlider1, 1164
ySlider2, 1164
ySlider3, 1164
ySlider4, 1164
ySlider5, 1164
Ui_MainWindow, 1137
actionAbout, 1140
actionExit, 1140
centralWidget, 1140
gridLayout, 1140
retranslateUi, 1139
setupUi, 1139
tabWidget, 1141
Ui_ModeSelectDialog, 1141
buttonBox, 1144
gridLayout, 1144
label, 1144
modeList, 1144
retranslateUi, 1142
selAllButton, 1144
selNoneButton, 1145
setupUi, 1143
Ui_RebindWidget, 1165
actionTab, 1180
animBox, 1180
animButton, 1181
animKrBox, 1181
animOnceBox, 1181
animTab, 1181
applyButton, 1181
btnClearMacro, 1181
btnStartMacro, 1181
btnStopMacro, 1181
cancelButton, 1181

delayButtons, 1182
dpiBox, 1182
dpiButton, 1182
dpiCustLabel, 1182
dpiCustXBox, 1182
dpiCustYBox, 1182
dpiWarning, 1182
fnBox, 1182
fnButton, 1183
gridLayout, 1183
gridLayout_10, 1183
gridLayout_2, 1183
gridLayout_3, 1183
gridLayout_4, 1183
gridLayout_5, 1183
gridLayout_8, 1183
groupBox_2, 1183
groupBox_3, 1184
groupBox_4, 1184
horizontalLayout, 1184
horizontalLayout_2, 1184
horizontalLayout_3, 1184
horizontalSpacer, 1184
horizontalSpacer_10, 1184
horizontalSpacer_11, 1184
horizontalSpacer_12, 1184
horizontalSpacer_13, 1185
horizontalSpacer_14, 1185
horizontalSpacer_15, 1185
horizontalSpacer_16, 1185
horizontalSpacer_17, 1185
horizontalSpacer_18, 1185
horizontalSpacer_19, 1185
horizontalSpacer_2, 1185
horizontalSpacer_20, 1185
horizontalSpacer_21, 1186
horizontalSpacer_22, 1186
horizontalSpacer_23, 1186
horizontalSpacer_24, 1186
horizontalSpacer_25, 1186
horizontalSpacer_26, 1186
horizontalSpacer_3, 1186
horizontalSpacer_4, 1186
horizontalSpacer_5, 1186
horizontalSpacer_6, 1187
horizontalSpacer_7, 1187
horizontalSpacer_8, 1187
horizontalSpacer_9, 1187
keyTab, 1187
label, 1187
layoutWidget, 1187
layoutWidget1, 1187
layoutWidget2, 1187
layoutWidget_2, 1188
lbl_macro, 1188
lightBox, 1188
lightButton, 1188
lightWrapBox, 1188
lockBox, 1188
lockButton, 1188
macroTab, 1188
mb2Box, 1189
mb2Button, 1189
mbBox, 1189
mbButton, 1189
mediaBox, 1189
mediaButton, 1189
modBox, 1189
modButton, 1189
modeBox, 1189
modeButton, 1190
modeWrapBox, 1190
mouseTab, 1190
numBox, 1190
numButton, 1190
progTipLabel, 1191
programKpBox, 1190
programKpButton, 1190
programKpExtra, 1190
programKpModeBox, 1191
programKpSIBox, 1191
programKrBox, 1191
programKrButton, 1191
programKrExtra, 1191
programKrModeBox, 1191
programKrSIBox, 1191
programTab, 1191
pteMacroBox, 1192
pteMacroComment, 1192
pteMacroText, 1192
rb_delay_asTyped, 1192
rb_delay_default, 1192
rb_delay_no, 1192
resetButton, 1192
retranslateUi, 1168
setupUi, 1170
tabWidget, 1192
txtBuffer, 1193
typingBox, 1193
typingButton, 1193
unbindButton, 1193
verticalLayout, 1193
verticalLayout_2, 1193
verticalSpacer, 1193
verticalSpacer_10, 1193
verticalSpacer_11, 1193
verticalSpacer_2, 1194
verticalSpacer_3, 1194
verticalSpacer_4, 1194
verticalSpacer_5, 1194
verticalSpacer_6, 1194
verticalSpacer_7, 1194
verticalSpacer_8, 1194
wheelBox, 1194
wheelButton, 1194
Ui_SettingsWidget, 1195

altBox, 1202
altLabel, 1202
autoFWBox, 1202
capsBox, 1202
capsLabel, 1202
ctrlBox, 1203
ctrlLabel, 1203
devicesLabel, 1203
extraButton, 1203
gridLayout, 1203
gridLayout_3, 1203
gridLayout_5, 1203
horizontalLayout, 1203
horizontalLayout_2, 1203
horizontalSpacer, 1203
horizontalSpacer_10, 1204
horizontalSpacer_2, 1204
horizontalSpacer_7, 1204
horizontalSpacer_9, 1204
label, 1204
label_2, 1204
label_3, 1204
label_5, 1204
label_6, 1204
label_8, 1205
layoutBox, 1205
line, 1205
line_2, 1205
line_3, 1205
loginItemBox, 1205
pushButton, 1205
retranslateUi, 1197
setupUi, 1198
shiftBox, 1205
shiftLabel, 1205
versionLabel, 1206
verticalSpacer, 1206
verticalSpacer_10, 1206
verticalSpacer_2, 1206
verticalSpacer_3, 1206
verticalSpacer_4, 1206
verticalSpacer_5, 1206
verticalSpacer_6, 1206
verticalSpacer_7, 1206
verticalSpacer_8, 1207
winBox, 1207
winLabel, 1207
uiUpdated
KPerfWidget, 710
unbindButton
Ui::RebindWidget, 963
Ui_RebindWidget, 1193
uncompressedSize
QuaZipFileInfo, 903
QuaZipFileInfo64, 910
QuaZipFilePrivate, 915
QuaZipNewInfo, 924
unz64_file_pos_s, 1443
unz64_s, 1389
unz64_file_pos
unzip.h, 1446
unz64local_CheckCurrentFileCoherencyHeader
unzip.c, 1392
unz64local_DosDateToTmuDate
unzip.c, 1394
unz64local_GetCurrentFileInfoInternal
unzip.c, 1395
unz64local_SearchCentralDir
unzip.c, 1402
unz64local_SearchCentralDir64
unzip.c, 1403
unz64local_getByte
unzip.c, 1395
unz64local_getLong
unzip.c, 1399
unz64local_getLong64
unzip.c, 1400
unz64local_getShort
unzip.c, 1401
unz_file_info64_internal_s, 1388
unz_file_info64_s, 1440
unz_file_info_s, 1441
unz_file_pos_s, 1443
unz_global_info64_s, 1438
unz_global_info_s, 1439
unz_copyright
unzip.c, 1435
unz_file_info
unzip.h, 1446
unz_file_info64
unzip.h, 1446
unz_file_info64_internal
unzip.c, 1392
unz_file_pos
unzip.h, 1446
unz_global_info
unzip.h, 1446
unz_global_info64
unzip.h, 1446
unzClearFlags
unzip.c, 1405
unzip.h, 1447
unzClose
unzip.c, 1406
unzCloseCurrentFile
unzip.c, 1407
unzFile
unzip.h, 1446
unzGetCurrentFileInfo
unzip.c, 1408
unzGetCurrentFileInfo64
unzip.c, 1409
unzGetCurrentFileZStreamPos64
unzip.c, 1410
unzGetFilePos
unzip.c, 1410

unzip.h, 1447
unzGetFilePos64
unzip.c, 1411
unzip.h, 1448
unzGetGlobalComment
unzip.c, 1411
unzGetGlobalInfo
unzip.c, 1412
unzGetGlobalInfo64
unzip.c, 1412
unzGetLocalExtrafield
unzip.c, 1413
unzGetOffset
unzip.c, 1413
unzip.h, 1448
unzGetOffset64
unzip.c, 1414
unzip.h, 1449
unzGoToFilePos
unzip.c, 1414
unzip.h, 1449
unzGoToFilePos64
unzip.c, 1415
unzip.h, 1450
unzGoToFirstFile
unzip.c, 1416
unzGoToNextFile
unzip.c, 1417
unzLocateFile
unzip.c, 1418
unzOpen
unzip.c, 1419
unzOpen2
unzip.c, 1419
unzOpen2_64
unzip.c, 1420
unzOpen64
unzip.c, 1421
unzOpenCurrentFile
unzip.c, 1421
unzOpenCurrentFile2
unzip.c, 1422
unzOpenCurrentFile3
unzip.c, 1422
unzOpenCurrentFilePassword
unzip.c, 1425
unzOpenInternal
unzip.c, 1426
unzip.h, 1451
unzReadCurrentFile
unzip.c, 1429
unzSetFlags
unzip.c, 1432
unzip.h, 1454
unzSetOffset
unzip.c, 1433
unzip.h, 1455
unzSetOffset64
unzip.c, 1433
unzip.h, 1455
unzStringFileNameCompare
unzip.c, 1434
unzeof
unzip.c, 1408
unzip.c
_Z_OF, 1392
ALLOC, 1390
BUFREADCOMMENT, 1391
local, 1391
SIZECENTRALDIRITEM, 1391
SIZEZIPLOCALHEADER, 1391
strcmpcasenosensitive_internal, 1392
TRYFREE, 1391
UNZ_BUFSIZE, 1391
UNZ_MAXFILENAMEINZIP, 1391
unz64local_CheckCurrentFileCoherencyHeader, 1392
unz64local_DosDateToTmuDate, 1394
unz64local_GetCurrentFileInfoInternal, 1395
unz64local_SearchCentralDir, 1402
unz64local_SearchCentralDir64, 1403
unz64local_getByte, 1395
unz64local_getLong, 1399
unz64local_getLong64, 1400
unz64local_getShort, 1401
unz_copyright, 1435
unz_file_info64_internal, 1392
unzClearFlags, 1405
unzClose, 1406
unzCloseCurrentFile, 1407
unzGetCurrentFileInfo, 1408
unzGetCurrentFileInfo64, 1409
unzGetCurrentFileZStreamPos64, 1410
unzGetFilePos, 1410
unzGetFilePos64, 1411
unzGetGlobalComment, 1411
unzGetGlobalInfo, 1412
unzGetGlobalInfo64, 1412
unzGetLocalExtrafield, 1413
unzGetOffset, 1413
unzGetOffset64, 1414
unzGoToFilePos, 1414
unzGoToFilePos64, 1415
unzGoToFirstFile, 1416
unzGoToNextFile, 1417
unzLocateFile, 1418
unzOpen, 1419
unzOpen2, 1419
unzOpen2_64, 1420
unzOpen64, 1421
unzOpenCurrentFile, 1421
unzOpenCurrentFile2, 1422
unzOpenCurrentFile3, 1422
unzOpenCurrentFilePassword, 1425
unzOpenInternal, 1426
unzReadCurrentFile, 1429

unzSetFlags, 1432
unzSetOffset, 1433
unzSetOffset64, 1433
unzStringFileNameCompare, 1434
unzeof, 1408
unztell, 1434
unztell64, 1435
z_crc_t, 1392
unzip.h
_Z_OF, 1446, 1447
tm_unz, 1446
UNZ_AUTO_CLOSE, 1444
UNZ_BADZIPFILE, 1444
UNZ_CRCERROR, 1444
UNZ_DEFAULT_FLAGS, 1444
UNZ_EOF, 1445
UNZ_ERRNO, 1445
UNZ_INTERNALERROR, 1445
UNZ_OK, 1445
UNZ_PARAMERROR, 1445
unz64_file_pos, 1446
unz_file_info, 1446
unz_file_info64, 1446
unz_file_pos, 1446
unz_global_info, 1446
unz_global_info64, 1446
unzClearFlags, 1447
unzFile, 1446
unzGetFilePos, 1447
unzGetFilePos64, 1448
unzGetOffset, 1448
unzGetOffset64, 1449
unzGoToFilePos, 1449
unzGoToFilePos64, 1450
unzOpenInternal, 1451
unzSetFlags, 1454
unzSetOffset, 1455
unzSetOffset64, 1455
Z_BZIP2ED, 1445
unztell
unzip.c, 1434
unztell64
unzip.c, 1435
update
KbBind, 387
KbPerf, 529
QuaAdler32, 806
QuaChecksum32, 808
QuaCrc32, 812
update_keys
crypt.h, 1337
updateBind
KbBindWidget, 406
updateFwButton
KbWidget, 584
updateImage
ColorButton, 147
GradientButton, 200
updateLayout
Kb, 300
updateLight
KbLightWidget, 470
updateMapper
AnimSettingDialog, 117
KPerfWidget, 712
updateModifiers
SettingsWidget, 1016
updateParam
AnimSettingDialog, 112
updateParams
KbAnim, 330
updatePresets
GradientDialog, 220
updateProfileList
KbWidget, 585
updateSelDisplay
KbBindWidget, 407
updateStops
AnimSettingDialog, 115
updateVersion
MainWindow, 745
updated
KbBind, 389
KbLight, 448
KbMode, 497
url
KbFirmware::FW, 180
urlParam
animscript.cpp, 1215
urldecode
ckb-anim.h, 1239
UsbId, 1207
guid, 1212
guidString, 1209
hwModified, 1212
hwModifiedString, 1210
modified, 1212
modifiedString, 1210, 1211
newGuid, 1211
newModified, 1211
UsbId, 1209
UsbId, 1209
usbModel
Kb, 305
usbSerial
Kb, 306
usize
QuaZipFile, 898

VERSIONMADEBY
zip.c, 1464
value
CkbSettings, 135
KeyAction, 627
QuaAdler32, 806
QuaChecksum32, 808
QuaCrc32, 812

verifyFw
fwupgradedialog.cpp, 1252
version
AnimScript, 80
versionCreated
QuaZipFileInfo, 903
QuaZipFileInfo64, 910
versionForBoard
KbFirmware, 419
versionLabel
Ui::SettingsWidget, 1029
Ui_SettingsWidget, 1206
versionNeeded
QuaZipFileInfo, 903
QuaZipFileInfo64, 910
versionUpdated
KbManager, 483
verticalLayout
Ui::KbAnimWidget, 342
Ui::KbBindWidget, 397
Ui::KbLightWidget, 458
Ui::MPerfWidget, 772
Ui::RebindWidget, 963
Ui_KbAnimWidget, 1084
Ui_KbBindWidget, 1091
Ui_KbLightWidget, 1099
Ui_MPerfWidget, 1161
Ui_RebindWidget, 1193
verticalLayout_2
Ui::RebindWidget, 963
Ui_RebindWidget, 1193
verticalSpacer
Ui::AnimSettingDialog, 96
Ui::ExtraSettingsWidget, 169
Ui::GradientDialog, 209
Ui::KbAnimWidget, 342
Ui::KbBindWidget, 397
Ui::KbWidget, 596
Ui::KPerfWidget, 699
Ui::LayoutDialog, 717
Ui::MPerfWidget, 772
Ui::RebindWidget, 964
Ui::SettingsWidget, 1029
Ui_AnimSettingDialog, 1050
Ui_ExtraSettingsWidget, 1061
Ui_GradientDialog, 1075
Ui_KbAnimWidget, 1084
Ui_KbBindWidget, 1091
Ui_KbWidget, 1112
Ui_KPerfWidget, 1132
Ui_LayoutDialog, 1137
Ui_MPerfWidget, 1161
Ui_RebindWidget, 1193
Ui_SettingsWidget, 1206
verticalSpacer_10
Ui::ExtraSettingsWidget, 169
Ui::RebindWidget, 964
Ui::SettingsWidget, 1029
Ui_ExtraSettingsWidget, 1061
Ui_RebindWidget, 1193
Ui_SettingsWidget, 1206
verticalSpacer_11
Ui::ExtraSettingsWidget, 169
Ui::RebindWidget, 964
Ui_ExtraSettingsWidget, 1061
Ui_RebindWidget, 1193
verticalSpacer_2
Ui::AnimAddDialog, 52
Ui::AnimSettingDialog, 96
Ui::ExtraSettingsWidget, 169
Ui::GradientDialog, 209
Ui::KbAnimWidget, 342
Ui::KbLightWidget, 458
Ui::KPerfWidget, 699
Ui::LayoutDialog, 717
Ui::MPerfWidget, 772
Ui::RebindWidget, 964
Ui::SettingsWidget, 1029
Ui_AnimAddDialog, 1037
Ui_AnimSettingDialog, 1050
Ui_ExtraSettingsWidget, 1061
Ui_GradientDialog, 1076
Ui_KbAnimWidget, 1084
Ui_KbLightWidget, 1099
Ui_KPerfWidget, 1132
Ui_LayoutDialog, 1137
Ui_MPerfWidget, 1161
Ui_RebindWidget, 1194
Ui_SettingsWidget, 1206
verticalSpacer_3
Ui::AnimAddDialog, 52
Ui::AnimSettingDialog, 97
Ui::ExtraSettingsWidget, 169
Ui::KbAnimWidget, 342
Ui::KbLightWidget, 458
Ui::KPerfWidget, 699
Ui::LayoutDialog, 717
Ui::MPerfWidget, 773
Ui::RebindWidget, 964
Ui::SettingsWidget, 1029
Ui_AnimAddDialog, 1037
Ui_AnimSettingDialog, 1050
Ui_ExtraSettingsWidget, 1061
Ui_KbAnimWidget, 1085
Ui_KbLightWidget, 1099
Ui_KPerfWidget, 1132
Ui_LayoutDialog, 1137
Ui_MPerfWidget, 1162
Ui_RebindWidget, 1194
Ui_SettingsWidget, 1206
verticalSpacer_4
Ui::ExtraSettingsWidget, 169
Ui::KPerfWidget, 699
Ui::LayoutDialog, 717
Ui::RebindWidget, 964
Ui::SettingsWidget, 1029

Ui_ExtraSettingsWidget, 1061
Ui_KPerfWidget, 1132
Ui_LayoutDialog, 1137
Ui_RebindWidget, 1194
Ui_SettingsWidget, 1206
verticalSpacer_5
Ui::ExtraSettingsWidget, 169
Ui::RebindWidget, 964
Ui::SettingsWidget, 1029
Ui_ExtraSettingsWidget, 1062
Ui_RebindWidget, 1194
Ui_SettingsWidget, 1206
verticalSpacer_6
Ui::ExtraSettingsWidget, 169
Ui::RebindWidget, 964
Ui::SettingsWidget, 1029
Ui_ExtraSettingsWidget, 1062
Ui_RebindWidget, 1194
Ui_SettingsWidget, 1206
verticalSpacer_7
Ui::ExtraSettingsWidget, 170
Ui::RebindWidget, 964
Ui::SettingsWidget, 1030
Ui_ExtraSettingsWidget, 1062
Ui_RebindWidget, 1194
Ui_SettingsWidget, 1206
verticalSpacer_8
Ui::ExtraSettingsWidget, 170
Ui::RebindWidget, 965
Ui::SettingsWidget, 1030
Ui_ExtraSettingsWidget, 1062
Ui_RebindWidget, 1194
Ui_SettingsWidget, 1207
verticalSpacer_9
Ui::ExtraSettingsWidget, 170
Ui_ExtraSettingsWidget, 1062

wheelBox
Ui::RebindWidget, 965
Ui_RebindWidget, 1194
wheelButton
Ui::RebindWidget, 965
Ui_RebindWidget, 1194
wheelKeys
RebindWidget, 1001
widget
Ui::GradientDialog, 209
Ui::KbBindWidget, 397
Ui::KbLightWidget, 459
Ui_GradientDialog, 1076
Ui_KbBindWidget, 1091
Ui_KbLightWidget, 1099
width
Key, 600
KeyMap, 655
winBox
Ui::SettingsWidget, 1030
Ui_SettingsWidget, 1207
winLabel
Ui::SettingsWidget, 1030
Ui_SettingsWidget, 1207
winLock
KbBind, 389, 390
Write_EndOfCentralDirectoryRecord
zip.c, 1472
Write_GlobalComment
zip.c, 1473
Write_LocalFileHeader
zip.c, 1474
Write_Zip64EndOfCentralDirectoryLocator
zip.c, 1476
Write_Zip64EndOfCentralDirectoryRecord
zip.c, 1477
writeCache
CkbSettings, 137
writeData
QuaGzipFile, 818
QuaZIODevice, 826
QuaZipFile, 899
writePos
QuaZipFilePrivate, 915
writeProfileHeader
Kb, 301

x
Key, 600
xBox0
Ui::MPerfWidget, 773
Ui_MPerfWidget, 1162
xBox1
Ui::MPerfWidget, 773
Ui_MPerfWidget, 1162
xBox2
Ui::MPerfWidget, 773
Ui_MPerfWidget, 1162
xBox3
Ui::MPerfWidget, 773
Ui_MPerfWidget, 1162
xBox4
Ui::MPerfWidget, 773
Ui_MPerfWidget, 1162
xBox5
Ui::MPerfWidget, 773
Ui_MPerfWidget, 1162
xSlider0
Ui::MPerfWidget, 773
Ui_MPerfWidget, 1162
xSlider1
Ui::MPerfWidget, 773
Ui_MPerfWidget, 1162
xSlider2
Ui::MPerfWidget, 774
Ui_MPerfWidget, 1163
xSlider3
Ui::MPerfWidget, 774
Ui_MPerfWidget, 1163
xSlider4
Ui::MPerfWidget, 774

Ui_MPerfWidget, 1163
xSlider5
Ui::MPerfWidget, 774
Ui_MPerfWidget, 1163
xyBox
Ui::MPerfWidget, 774
Ui_MPerfWidget, 1163
xyLinkPath
mperfwidget.cpp, 1332

y
Key, 600
YES
KbFirmware, 412
yBox0
Ui::MPerfWidget, 774
Ui_MPerfWidget, 1163
yBox1
Ui::MPerfWidget, 774
Ui_MPerfWidget, 1163
yBox2
Ui::MPerfWidget, 774
Ui_MPerfWidget, 1163
yBox3
Ui::MPerfWidget, 774
Ui_MPerfWidget, 1163
yBox4
Ui::MPerfWidget, 775
Ui_MPerfWidget, 1164
yBox5
Ui::MPerfWidget, 775
Ui_MPerfWidget, 1164
ySlider0
Ui::MPerfWidget, 775
Ui_MPerfWidget, 1164
ySlider1
Ui::MPerfWidget, 775
Ui_MPerfWidget, 1164
ySlider2
Ui::MPerfWidget, 775
Ui_MPerfWidget, 1164
ySlider3
Ui::MPerfWidget, 775
Ui_MPerfWidget, 1164
ySlider4
Ui::MPerfWidget, 775
Ui_MPerfWidget, 1164
ySlider5
Ui::MPerfWidget, 775
Ui_MPerfWidget, 1164
year
AnimScript, 80

Z_BUFSIZE
zip.c, 1464
Z_BZIP2ED
unzip.h, 1445
zip.h, 1517
Z_MAXFILENAMEINZIP
zip.c, 1464
z_crc_t
unzip.c, 1392
zip.c, 1465
ZCALLBACK
ioapi.h, 1342
ZCLOSE64
ioapi.h, 1342
ZERROR64
ioapi.h, 1342
ZFAKECLOSE64
ioapi.h, 1342
ZIP64ENDHEADERMAGIC
zip.c, 1464
ZIP64ENDLOCHEADERMAGIC
zip.c, 1465
ZIP_AUTO_CLOSE
zip.h, 1517
ZIP_BADZIPFILE
zip.h, 1517
ZIP_DEFAULT_FLAGS
zip.h, 1517
ZIP_EOF
zip.h, 1517
ZIP_ERRNO
zip.h, 1517
ZIP_INTERNALERROR
zip.h, 1517
ZIP_OK
zip.h, 1517
ZIP_PARAMERROR
zip.h, 1518
ZIP_SEQUENTIAL
zip.h, 1518
ZOPEN64
ioapi.h, 1343
ZPOS64_T
ioapi.h, 1344
ZREAD64
ioapi.h, 1343
ZSEEK64
ioapi.h, 1343
ZTELL64
ioapi.h, 1344
ZWRITE64
ioapi.h, 1344
zdecode
crypt.h, 1336
zencode
crypt.h, 1336
zip
QuaZipFilePrivate, 915
zip.c
_Z_OF, 1465
ALLOC, 1462
add_data_in_datablock, 1465
allocate_new_datablock, 1466
BUFREADCOMMENT, 1463

CENTRALHEADERMAGIC, 1463
DESCRIPTORHEADERMAGIC, 1463
ENDHEADERMAGIC, 1463
free_datablock, 1467
free_linkedlist, 1467
init_linkedlist, 1468
LOCALHEADERMAGIC, 1463
linkedlist_data, 1465
linkedlist_datablock_internal, 1465
LoadCentralDirectoryRecord, 1469
local, 1463
MAKEULONG64, 1463
SEEK_CUR, 1464
SEEK_END, 1464
SEEK_SET, 1464
SIZECENTRALHEADER, 1464
SIZEDATA_INDATABLOCK, 1464
TRYFREE, 1464
VERSIONMADEBY, 1464
Write_EndOfCentralDirectoryRecord, 1472
Write_GlobalComment, 1473
Write_LocalFileHeader, 1474
Write_Zip64EndOfCentralDirectoryLocator, 1476
Write_Zip64EndOfCentralDirectoryRecord, 1477
Z_BUFSIZE, 1464
Z_MAXFILENAMEINZIP, 1464
z_crc_t, 1465
ZIP64ENDHEADERMAGIC, 1464
ZIP64ENDLOCHEADERMAGIC, 1465
zip64FlushWriteBuffer, 1478
zip64local_SearchCentralDir, 1484
zip64local_SearchCentralDir64, 1485
zip64local_TmzDateToDosDate, 1487
zip64local_getByte, 1479
zip64local_getLong, 1480
zip64local_getLong64, 1481
zip64local_getShort, 1482
zip64local_putValue, 1483
zip64local_putValue_inmemory, 1484
zip_copyright, 1512
zipClearFlags, 1488
zipClose, 1488
zipCloseFileInZip, 1490
zipCloseFileInZipRaw, 1491
zipCloseFileInZipRaw64, 1491
zipOpen, 1496
zipOpen2, 1496
zipOpen2_64, 1497
zipOpen3, 1498
zipOpen64, 1500
zipOpenNewFileInZip, 1501
zipOpenNewFileInZip2, 1501
zipOpenNewFileInZip2_64, 1502
zipOpenNewFileInZip3, 1502
zipOpenNewFileInZip3_64, 1503
zipOpenNewFileInZip4, 1503
zipOpenNewFileInZip4_64, 1504
zipOpenNewFileInZip64, 1508
zipRemoveExtraInfoBlock, 1508
zipSetFlags, 1509
zipWriteInFileInZip, 1510
zip.h
_Z_OF, 1518, 1519
DEF_MEM_LEVEL, 1517
tm_zip, 1518
Z_BZIP2ED, 1517
ZIP_AUTO_CLOSE, 1517
ZIP_BADZIPFILE, 1517
ZIP_DEFAULT_FLAGS, 1517
ZIP_EOF, 1517
ZIP_ERRNO, 1517
ZIP_INTERNALERROR, 1517
ZIP_OK, 1517
ZIP_PARAMERROR, 1518
ZIP_SEQUENTIAL, 1518
zipClearFlags, 1519
zipFile, 1518
zipOpen3, 1520
zipSetFlags, 1522
zipcharpc, 1518
zip64
QuaZipPrivate, 933
zip64_internal, 1462
zip64FlushWriteBuffer
zip.c, 1478
zip64local_SearchCentralDir
zip.c, 1484
zip64local_SearchCentralDir64
zip.c, 1485
zip64local_TmzDateToDosDate
zip.c, 1487
zip64local_getByte
zip.c, 1479
zip64local_getLong
zip.c, 1480
zip64local_getLong64
zip.c, 1481
zip64local_getShort
zip.c, 1482
zip64local_putValue
zip.c, 1483
zip64local_putValue_inmemory
zip.c, 1484
zip_fileinfo, 1515
zip_copyright
zip.c, 1512
zipClearFlags
zip.c, 1488
zip.h, 1519
zipClose
zip.c, 1488
zipCloseFileInZip
zip.c, 1490
zipCloseFileInZipRaw
zip.c, 1491
zipCloseFileInZipRaw64

zip.c, 1491
zipError
QuaZipFilePrivate, 915
QuaZipPrivate, 933
zipFile
zip.h, 1518
zipName
QuaZipPrivate, 933
zipOpen
zip.c, 1496
zipOpen2
zip.c, 1496
zipOpen2_64
zip.c, 1497
zipOpen3
zip.c, 1498
zip.h, 1520
zipOpen64
zip.c, 1500
zipOpenNewFileInZip
zip.c, 1501
zipOpenNewFileInZip2
zip.c, 1501
zipOpenNewFileInZip2_64
zip.c, 1502
zipOpenNewFileInZip3
zip.c, 1502
zipOpenNewFileInZip3_64
zip.c, 1503
zipOpenNewFileInZip4
zip.c, 1503
zipOpenNewFileInZip4_64
zip.c, 1504
zipOpenNewFileInZip64
zip.c, 1508
zipRemoveExtraInfoBlock
zip.c, 1508
zipSetFlags
zip.c, 1509
zip.h, 1522
zipWriteInFileInZip
zip.c, 1510
zipcharpc
zip.h, 1518
zlib_filefunc64_32_def_s, 1341
zlib_filefunc64_def_s, 1340
zlib_filefunc_def_s, 1340
zlib_filefunc64_32_def
ioapi.h, 1344
zlib_filefunc64_def
ioapi.h, 1344
zlib_filefunc_def
ioapi.h, 1344