

libyui-qt  
2.49.11

Generated by Doxygen 1.8.15



<b>1 Hierarchical Index</b>	<b>1</b>
1.1 Class Hierarchy	1
<b>2 Class Index</b>	<b>5</b>
2.1 Class List	5
<b>3 Class Documentation</b>	<b>9</b>
3.1 BusyBar Class Reference	9
3.1.1 Detailed Description	10
3.2 YQTimezoneSelectorPrivate::Location Struct Reference	10
3.2.1 Detailed Description	10
3.3 MyTextEdit Class Reference	11
3.3.1 Detailed Description	12
3.4 QY2CharValidator Class Reference	12
3.4.1 Detailed Description	13
3.4.2 Constructor & Destructor Documentation	13
3.4.2.1 QY2CharValidator()	13
3.4.3 Member Function Documentation	13
3.4.3.1 setValidChars()	14
3.4.3.2 validate()	14
3.5 QY2CheckListItem Class Reference	15
3.5.1 Detailed Description	16
3.5.2 Member Function Documentation	16
3.5.2.1 serial()	16
3.5.2.2 setBackgroundColor()	17
3.5.2.3 setTextColor()	17
3.5.2.4 toolTip()	17
3.5.2.5 updateData()	18
3.5.2.6 updateStatus()	18
3.6 QY2ComboTabWidget Class Reference	18
3.6.1 Detailed Description	20
3.6.2 Constructor & Destructor Documentation	20
3.6.2.1 QY2ComboTabWidget()	20
3.6.3 Member Function Documentation	21
3.6.3.1 addPage()	21
3.6.3.2 currentChanged	21
3.6.3.3 showPage	21
3.6.3.4 showPageIndex	22
3.7 QY2DiskUsageList Class Reference	22
3.7.1 Detailed Description	24

3.7.2 Constructor & Destructor Documentation	24
3.7.2.1 QY2DiskUsageList()	24
3.8 QY2DiskUsageListItem Class Reference	25
3.8.1 Detailed Description	27
3.8.2 Constructor & Destructor Documentation	27
3.8.2.1 QY2DiskUsageListItem()	27
3.8.3 Member Function Documentation	27
3.8.3.1 deviceName()	27
3.8.3.2 freeSize()	28
3.8.3.3 name()	28
3.8.3.4 operator<()	29
3.8.3.5 paintPercentageBar()	29
3.8.3.6 totalSize()	30
3.8.3.7 updateData()	30
3.8.3.8 updateStatus()	31
3.8.3.9 usedPercent()	31
3.8.3.10 usedSize()	32
3.9 QY2DiskUsagePercentageItem Class Reference	32
3.9.1 Detailed Description	33
3.10 QY2HelpDialog Class Reference	33
3.10.1 Detailed Description	34
3.11 QY2ListView Class Reference	35
3.11.1 Detailed Description	37
3.11.2 Member Function Documentation	37
3.11.2.1 columnWidthChanged	37
3.11.2.2 minimumSizeHint()	38
3.11.2.3 mouseDoubleClickEvent()	38
3.11.2.4 mousePressEvent()	38
3.11.2.5 mouseReleaseEvent()	39
3.11.2.6 selectSomething	39
3.11.2.7 tooltip()	39
3.11.2.8 updateItemData	40
3.11.2.9 updateItemStates	40
3.12 QY2ListViewItem Class Reference	41
3.12.1 Detailed Description	42
3.12.2 Member Function Documentation	42
3.12.2.1 operator<()	42
3.12.2.2 serial()	43
3.12.2.3 tooltip()	43

3.12.2.4 <code>updateData()</code> . . . . .	43
3.12.2.5 <code>updateStatus()</code> . . . . .	44
3.13 QY2RelNotesDialog Class Reference . . . . .	44
3.13.1 Detailed Description . . . . .	45
3.14 QY2StyleEditor Class Reference . . . . .	45
3.14.1 Detailed Description . . . . .	46
3.15 QY2Styler Class Reference . . . . .	46
3.15.1 Detailed Description . . . . .	48
3.15.2 Constructor & Destructor Documentation . . . . .	48
3.15.2.1 <code>QY2Styler()</code> . . . . .	49
3.15.3 Member Function Documentation . . . . .	49
3.15.3.1 <code>buildStyleSheet()</code> . . . . .	49
3.15.3.2 <code>buildStyleSheetFromFile()</code> . . . . .	50
3.15.3.3 <code>loadAlternateStyleSheet()</code> . . . . .	50
3.15.3.4 <code>loadDefaultStyleSheet()</code> . . . . .	51
3.15.3.5 <code>loadStyleSheet()</code> . . . . .	51
3.15.3.6 <code>processUrls()</code> . . . . .	52
3.15.3.7 <code>registerChildWidget()</code> . . . . .	52
3.15.3.8 <code>registerWidget()</code> . . . . .	53
3.15.3.9 <code>setAlternateStyleSheet()</code> . . . . .	53
3.15.3.10 <code>setDefaultStyleSheet()</code> . . . . .	54
3.15.3.11 <code>setStyleSheet()</code> . . . . .	54
3.15.3.12 <code>styleSheetExists()</code> . . . . .	55
3.15.3.13 <code>unregisterWidget()</code> . . . . .	55
3.16 YQWizard::Step Class Reference . . . . .	56
3.16.1 Detailed Description . . . . .	57
3.16.2 Constructor & Destructor Documentation . . . . .	57
3.16.2.1 <code>~Step()</code> . . . . .	57
3.17 YQWizard::StepHeading Class Reference . . . . .	58
3.17.1 Detailed Description . . . . .	59
3.18 YQWizard::TreeItem Class Reference . . . . .	59
3.18.1 Detailed Description . . . . .	60
3.19 YQAlignment Class Reference . . . . .	61
3.19.1 Detailed Description . . . . .	62
3.19.2 Constructor & Destructor Documentation . . . . .	62
3.19.2.1 <code>YQAlignment()</code> . . . . .	62
3.19.3 Member Function Documentation . . . . .	62
3.19.3.1 <code>moveChild()</code> . . . . .	63
3.19.3.2 <code>setBackgroundPixmap()</code> . . . . .	63

---

3.19.3.3	setEnabled()	63
3.19.3.4	setSize()	63
3.20	YQApplication Class Reference	64
3.20.1	Detailed Description	67
3.20.2	Constructor & Destructor Documentation	67
3.20.2.1	YQApplication()	67
3.20.3	Member Function Documentation	67
3.20.3.1	askForExistingDirectory()	68
3.20.3.2	askForExistingFile()	68
3.20.3.3	askForSaveFileName()	69
3.20.3.4	autoFonts()	70
3.20.3.5	beep()	70
3.20.3.6	busyCursor()	70
3.20.3.7	currentFont()	71
3.20.3.8	deviceUnits()	71
3.20.3.9	glyph()	72
3.20.3.10	headingFont()	72
3.20.3.11	layoutUnits()	73
3.20.3.12	makeScreenShot()	73
3.20.3.13	maybeLeftHandedUser()	74
3.20.3.14	normalCursor()	74
3.20.3.15	openContextMenu()	74
3.20.3.16	pickAutoFonts()	75
3.20.3.17	setApplicationIcon()	75
3.20.3.18	setApplicationTitle()	75
3.20.3.19	setLangFonts()	76
3.20.3.20	setLanguage()	76
3.21	YQBarGraph Class Reference	77
3.21.1	Detailed Description	79
3.21.2	Member Function Documentation	79
3.21.2.1	defaultTextColor()	79
3.21.2.2	doUpdate()	79
3.21.2.3	paintEvent()	79
3.21.2.4	preferredHeight()	80
3.21.2.5	preferredWidth()	80
3.21.2.6	setEnabled()	80
3.21.2.7	setSize()	80
3.22	YQBusyIndicator Class Reference	81
3.22.1	Detailed Description	82

---

3.22.2 Member Function Documentation	82
3.22.2.1 preferredHeight()	82
3.22.2.2 preferredWidth()	83
3.22.2.3 setAlive()	83
3.22.2.4 setEnabled()	83
3.22.2.5 setLabel()	84
3.22.2.6 setSize()	84
3.22.2.7 setTimeout()	84
3.22.2.8 timeout()	85
3.23 YQGroupBox Class Reference	85
3.23.1 Detailed Description	86
3.23.2 Member Function Documentation	86
3.23.2.1 moveChild()	86
3.23.2.2 setEnabled()	86
3.23.2.3 setSize()	87
3.24 YQCheckBox Class Reference	87
3.24.1 Detailed Description	88
3.24.2 Member Function Documentation	88
3.24.2.1 preferredHeight()	88
3.24.2.2 preferredWidth()	89
3.24.2.3 setEnabled()	89
3.24.2.4 setKeyboardFocus()	89
3.24.2.5 setLabel()	89
3.24.2.6 setSize()	90
3.24.2.7 setUseBoldFont()	90
3.24.2.8 setValue()	90
3.24.2.9 value()	91
3.25 YQCheckBoxFrame Class Reference	91
3.25.1 Detailed Description	92
3.25.2 Member Function Documentation	92
3.25.2.1 preferredHeight()	92
3.25.2.2 preferredWidth()	93
3.25.2.3 setEnabled()	93
3.25.2.4 setKeyboardFocus()	93
3.25.2.5 setLabel()	94
3.25.2.6 setSize()	94
3.25.2.7 setValue()	94
3.25.2.8 value()	95
3.26 YQComboBox Class Reference	95

---

3.26.1 Detailed Description	97
3.26.2 Member Function Documentation	97
3.26.2.1 addItem()	97
3.26.2.2 deleteAllItems()	97
3.26.2.3 preferredHeight()	98
3.26.2.4 preferredWidth()	98
3.26.2.5 setEnabled()	98
3.26.2.6 setInputMaxLength()	98
3.26.2.7 setLabel()	99
3.26.2.8 setSize()	99
3.26.2.9 setText()	99
3.26.2.10 setValidChars()	100
3.26.2.11 slotSelected	100
3.26.2.12 text()	101
3.26.2.13 textChanged	101
3.27 YQContextMenu Class Reference	102
3.27.1 Detailed Description	103
3.27.2 Member Function Documentation	103
3.27.2.1 preferredHeight()	103
3.27.2.2 preferredWidth()	104
3.27.2.3 rebuildMenuTree()	104
3.27.2.4 setSize()	104
3.28 YQDateField Class Reference	105
3.28.1 Detailed Description	106
3.28.2 Member Function Documentation	106
3.28.2.1 preferredHeight()	106
3.28.2.2 preferredWidth()	107
3.28.2.3 setEnabled()	107
3.28.2.4 setKeyboardFocus()	107
3.28.2.5 setLabel()	107
3.28.2.6 setSize()	108
3.28.2.7 setValue()	108
3.28.2.8 value()	108
3.29 YQDialog Class Reference	109
3.29.1 Detailed Description	111
3.29.2 Constructor & Destructor Documentation	112
3.29.2.1 YQDialog()	112
3.29.2.2 ~YQDialog()	113
3.29.3 Member Function Documentation	113

---



3.29.3.1 activate()	113
3.29.3.2 activateDefaultButton()	114
3.29.3.3 center()	114
3.29.3.4 ensureOnlyOneDefaultButton()	115
3.29.3.5 findWizard()	115
3.29.3.6 gettingFocus()	115
3.29.3.7 highlight()	116
3.29.3.8 keyPressEvent()	116
3.29.3.9 losingFocus()	117
3.29.3.10 openInternal()	117
3.29.3.11 pollEventInternal()	118
3.29.3.12 preferredHeight()	118
3.29.3.13 preferredWidth()	119
3.29.3.14 setDefaultButton()	119
3.29.3.15 setEnabled()	120
3.29.3.16 setSize()	120
3.29.3.17 waitForEventInternal()	120
3.29.3.18 wizardDefaultButton()	121
3.30 YQDownloadProgress Class Reference	121
3.30.1 Detailed Description	123
3.30.2 Member Function Documentation	123
3.30.2.1 preferredHeight()	123
3.30.2.2 preferredWidth()	123
3.30.2.3 setEnabled()	123
3.30.2.4 setExpectedSize()	124
3.30.2.5 setFilename()	124
3.30.2.6 setLabel()	124
3.30.2.7 setSize()	125
3.31 YQDumbTab Class Reference	125
3.31.1 Detailed Description	126
3.31.2 Member Function Documentation	126
3.31.2.1 addItem()	127
3.31.2.2 deleteAllItems()	127
3.31.2.3 deselectAllItems()	127
3.31.2.4 preferredHeight()	127
3.31.2.5 preferredWidth()	128
3.31.2.6 selectItem()	128
3.31.2.7 setEnabled()	128
3.31.2.8 setSize()	128

3.31.2.9 shortcutChanged()	129
3.32 YQEmpty Class Reference	129
3.32.1 Detailed Description	130
3.32.2 Member Function Documentation	130
3.32.2.1 setSize()	130
3.33 YQFrame Class Reference	131
3.33.1 Detailed Description	132
3.33.2 Member Function Documentation	132
3.33.2.1 preferredHeight()	132
3.33.2.2 preferredWidth()	132
3.33.2.3 setEnabled()	132
3.33.2.4 setSize()	133
3.34 YQGenericButton Class Reference	133
3.34.1 Detailed Description	135
3.34.2 Member Function Documentation	135
3.34.2.1 dialog()	135
3.34.2.2 eventFilter()	136
3.34.2.3 isShownAsDefault()	136
3.34.2.4 setEnabled()	136
3.34.2.5 setIcon()	137
3.34.2.6 setLabel()	137
3.34.2.7 setShortcut()	137
3.34.2.8 showAsDefault()	137
3.35 YQGraphPluginIf Class Reference	138
3.35.1 Detailed Description	138
3.36 YQGraphPluginStub Class Reference	138
3.36.1 Detailed Description	139
3.36.2 Member Function Documentation	139
3.36.2.1 createGraph()	139
3.37 YQImage Class Reference	140
3.37.1 Detailed Description	141
3.37.2 Constructor & Destructor Documentation	141
3.37.2.1 YQImage()	141
3.37.3 Member Function Documentation	142
3.37.3.1 preferredHeight()	142
3.37.3.2 preferredWidth()	142
3.37.3.3 setAutoScale()	142
3.37.3.4 setImage()	143
3.37.3.5 setSize()	143

3.38 YQInputField Class Reference	143
3.38.1 Detailed Description	145
3.38.2 Member Function Documentation	145
3.38.2.1 changed	145
3.38.2.2 preferredHeight()	146
3.38.2.3 preferredWidth()	146
3.38.2.4 setEnabled()	146
3.38.2.5 setInputMaxLength()	146
3.38.2.6 setKeyboardFocus()	147
3.38.2.7 setLabel()	147
3.38.2.8 setSize()	147
3.38.2.9 setValidChars()	148
3.38.2.10 setValue()	148
3.38.2.11 value()	149
3.39 YQIntField Class Reference	149
3.39.1 Detailed Description	150
3.39.2 Member Function Documentation	151
3.39.2.1 preferredHeight()	151
3.39.2.2 preferredWidth()	151
3.39.2.3 setEnabled()	151
3.39.2.4 setKeyboardFocus()	151
3.39.2.5 setLabel()	152
3.39.2.6 setSize()	152
3.39.2.7 setValueInternal()	152
3.39.2.8 value()	153
3.39.2.9 valueChangedSlot	153
3.40 YQLabel Class Reference	154
3.40.1 Detailed Description	155
3.40.2 Member Function Documentation	155
3.40.2.1 preferredHeight()	155
3.40.2.2 preferredWidth()	155
3.40.2.3 setEnabled()	155
3.40.2.4 setSize()	156
3.40.2.5 setText()	156
3.40.2.6 setUseBoldFont()	156
3.41 YQLayoutBox Class Reference	157
3.41.1 Detailed Description	158
3.41.2 Constructor & Destructor Documentation	158
3.41.2.1 YQLayoutBox()	158

3.41.3 Member Function Documentation	158
3.41.3.1 moveChild()	158
3.41.3.2 setEnabled()	159
3.41.3.3 setSize()	159
3.42 YQLogView Class Reference	160
3.42.1 Detailed Description	161
3.42.2 Member Function Documentation	161
3.42.2.1 displayLogText()	161
3.42.2.2 preferredHeight()	162
3.42.2.3 preferredWidth()	162
3.42.2.4 setEnabled()	162
3.42.2.5 setLabel()	162
3.42.2.6 setSize()	163
3.43 YQMainWinDock Class Reference	163
3.43.1 Detailed Description	165
3.43.2 Constructor & Destructor Documentation	165
3.43.2.1 YQMainWinDock()	165
3.43.3 Member Function Documentation	165
3.43.3.1 add()	166
3.43.3.2 closeEvent()	166
3.43.3.3 mainWinDock()	167
3.43.3.4 paintEvent()	167
3.43.3.5 remove()	167
3.43.3.6 resizeEvent()	168
3.43.3.7 show()	168
3.44 YQMenuButton Class Reference	169
3.44.1 Detailed Description	170
3.44.2 Member Function Documentation	170
3.44.2.1 preferredHeight()	170
3.44.2.2 preferredWidth()	171
3.44.2.3 rebuildMenuTree()	171
3.44.2.4 setEnabled()	171
3.44.2.5 setLabel()	172
3.44.2.6 setSize()	172
3.45 YQMultiLineEdit Class Reference	172
3.45.1 Detailed Description	174
3.45.2 Member Function Documentation	174
3.45.2.1 enforceMaxInputLength	174
3.45.2.2 preferredHeight()	174

3.45.2.3 preferredWidth()	175
3.45.2.4 setEnabled()	175
3.45.2.5 setInputMaxLength()	175
3.45.2.6 setLabel()	176
3.45.2.7 setSize()	176
3.45.2.8 setValue()	176
3.45.2.9 value()	177
3.46 YQMultiProgressMeter Class Reference	177
3.46.1 Detailed Description	179
3.46.2 Member Function Documentation	179
3.46.2.1 doUpdate()	179
3.46.2.2 drawSegment()	179
3.46.2.3 paintEvent()	180
3.46.2.4 preferredHeight()	181
3.46.2.5 preferredWidth()	182
3.46.2.6 setEnabled()	182
3.46.2.7 setMargin()	182
3.46.2.8 setSize()	183
3.46.2.9 setSpacing()	183
3.46.2.10 setTriThickness()	184
3.46.2.11 triThickness()	184
3.47 YQMultiSelectionBox Class Reference	185
3.47.1 Detailed Description	187
3.47.2 Member Function Documentation	187
3.47.2.1 addItem()	187
3.47.2.2 currentItem()	187
3.47.2.3 deleteAllItems()	188
3.47.2.4 deselectAllItems()	188
3.47.2.5 findItem()	188
3.47.2.6 preferredHeight()	189
3.47.2.7 preferredWidth()	189
3.47.2.8 selectItem()	189
3.47.2.9 setCurrentItem()	190
3.47.2.10 setEnabled()	190
3.47.2.11 setLabel()	191
3.47.2.12 setSize()	191
3.48 YQMultiSelectionBoxItem Class Reference	192
3.48.1 Detailed Description	193
3.49 YQOptionalWidgetFactory Class Reference	193

3.49.1 Detailed Description	195
3.49.2 Constructor & Destructor Documentation	195
3.49.2.1 YQOptionalWidgetFactory()	195
3.50 YQPackageSelectorPluginIf Class Reference	195
3.50.1 Detailed Description	196
3.51 YQPackageSelectorPluginStub Class Reference	196
3.51.1 Detailed Description	197
3.51.2 Member Function Documentation	197
3.51.2.1 createPackageSelector()	197
3.52 YQPartitionSplitter Class Reference	198
3.52.1 Detailed Description	199
3.52.2 Member Function Documentation	199
3.52.2.1 preferredHeight()	199
3.52.2.2 preferredWidth()	200
3.52.2.3 setEnabled()	200
3.52.2.4 setSize()	200
3.52.2.5 setValue()	201
3.52.2.6 value()	201
3.53 YQProgressBar Class Reference	202
3.53.1 Detailed Description	203
3.53.2 Member Function Documentation	203
3.53.2.1 preferredHeight()	203
3.53.2.2 preferredWidth()	203
3.53.2.3 setEnabled()	204
3.53.2.4 setLabel()	204
3.53.2.5 setSize()	204
3.53.2.6 setValue()	205
3.54 YQPushButton Class Reference	205
3.54.1 Detailed Description	206
3.54.2 Member Function Documentation	207
3.54.2.1 hit	207
3.54.2.2 preferredHeight()	207
3.54.2.3 preferredWidth()	208
3.54.2.4 setSize()	208
3.55 YQRadioButton Class Reference	209
3.55.1 Detailed Description	210
3.55.2 Member Function Documentation	210
3.55.2.1 preferredHeight()	210
3.55.2.2 preferredWidth()	210

3.55.2.3	setEnabled()	211
3.55.2.4	setKeyboardFocus()	211
3.55.2.5	setLabel()	211
3.55.2.6	setSize()	211
3.55.2.7	setUseBoldFont()	212
3.55.2.8	setValue()	212
3.55.2.9	value()	212
3.56	YQRadioButtonGroup Class Reference	213
3.56.1	Detailed Description	214
3.56.2	Member Function Documentation	214
3.56.2.1	addRadioButton()	214
3.56.2.2	setEnabled()	214
3.56.2.3	setSize()	214
3.57	YQRawLineEdit Class Reference	215
3.57.1	Detailed Description	216
3.57.2	Member Function Documentation	216
3.57.2.1	x11Event()	216
3.58	YQReplacePoint Class Reference	217
3.58.1	Detailed Description	217
3.58.2	Member Function Documentation	218
3.58.2.1	setEnabled()	218
3.58.2.2	setSize()	218
3.58.2.3	showChild()	218
3.59	YQRichText Class Reference	219
3.59.1	Detailed Description	220
3.59.2	Member Function Documentation	220
3.59.2.1	eventFilter()	221
3.59.2.2	preferredHeight()	221
3.59.2.3	preferredWidth()	221
3.59.2.4	setAutoScrollDown()	222
3.59.2.5	setEnabled()	222
3.59.2.6	setKeyboardFocus()	222
3.59.2.7	setPlainTextMode()	222
3.59.2.8	setSize()	223
3.59.2.9	setValue()	223
3.60	YQSelectionBox Class Reference	223
3.60.1	Detailed Description	225
3.60.2	Member Function Documentation	225
3.60.2.1	addItem()	225

3.60.2.2 addItem()	226
3.60.2.3 deleteAllItems()	226
3.60.2.4 deselectAllItems()	226
3.60.2.5 eventFilter()	227
3.60.2.6 preferredHeight()	227
3.60.2.7 preferredWidth()	228
3.60.2.8 returnDelayed	228
3.60.2.9 returnImmediately	228
3.60.2.10 selectItem()	229
3.60.2.11 setEnabled()	229
3.60.2.12 setKeyboardFocus()	229
3.60.2.13 setLabel()	230
3.60.2.14 setSize()	230
3.60.2.15 slotSelectionChanged	231
3.61 YQSignalBlocker Class Reference	231
3.61.1 Detailed Description	231
3.61.2 Constructor & Destructor Documentation	232
3.61.2.1 YQSignalBlocker()	232
3.61.2.2 ~YQSignalBlocker()	232
3.62 YQSlider Class Reference	232
3.62.1 Detailed Description	234
3.62.2 Constructor & Destructor Documentation	234
3.62.2.1 YQSlider()	234
3.62.3 Member Function Documentation	235
3.62.3.1 preferredHeight()	235
3.62.3.2 preferredWidth()	235
3.62.3.3 setEnabled()	235
3.62.3.4 setLabel()	236
3.62.3.5 setSize()	236
3.62.3.6 setValueInternal()	236
3.62.3.7 value()	237
3.62.3.8 valueChangedSlot	237
3.63 YQSpacing Class Reference	238
3.63.1 Detailed Description	238
3.63.2 Member Function Documentation	239
3.63.2.1 setSize()	239
3.64 YQSquash Class Reference	239
3.64.1 Detailed Description	240
3.64.2 Member Function Documentation	240



3.64.2.1	setEnabled()	240
3.64.2.2	setSize()	241
3.65	YQTable Class Reference	241
3.65.1	Detailed Description	243
3.65.2	Member Function Documentation	243
3.65.2.1	addItem()	243
3.65.2.2	addItems()	244
3.65.2.3	cellChanged()	244
3.65.2.4	deleteAllItems()	245
3.65.2.5	deselectAllItems()	245
3.65.2.6	preferredHeight()	245
3.65.2.7	preferredWidth()	246
3.65.2.8	selectItem()	246
3.65.2.9	setEnabled()	246
3.65.2.10	setKeepSorting()	247
3.65.2.11	setKeyboardFocus()	247
3.65.2.12	setSize()	247
3.65.2.13	slotContextMenu	248
3.66	YQTableListViewItem Class Reference	248
3.66.1	Detailed Description	249
3.67	YQTextBrowser Class Reference	250
3.67.1	Detailed Description	251
3.67.2	Member Function Documentation	251
3.67.2.1	setSource	251
3.68	YQTimeField Class Reference	251
3.68.1	Detailed Description	253
3.68.2	Member Function Documentation	253
3.68.2.1	preferredHeight()	253
3.68.2.2	preferredWidth()	253
3.68.2.3	setEnabled()	253
3.68.2.4	setKeyboardFocus()	254
3.68.2.5	setLabel()	254
3.68.2.6	setSize()	254
3.68.2.7	setValue()	255
3.68.2.8	value()	255
3.69	YQTimezoneSelector Class Reference	255
3.69.1	Detailed Description	257
3.69.2	Member Function Documentation	257
3.69.2.1	preferredHeight()	257

---

3.69.2.2 preferredWidth()	257
3.69.2.3 setSize()	257
3.70 YQTimezoneSelectorPrivate Class Reference	258
3.70.1 Detailed Description	258
3.71 YQTree Class Reference	259
3.71.1 Detailed Description	261
3.71.2 Member Function Documentation	261
3.71.2.1 currentItem()	261
3.71.2.2 deleteAllItems()	261
3.71.2.3 deselectAllItems()	262
3.71.2.4 preferredHeight()	262
3.71.2.5 preferredWidth()	262
3.71.2.6 rebuildTree()	263
3.71.2.7 selectItem()	263
3.71.2.8 setEnabled()	264
3.71.2.9 setLabel()	264
3.71.2.10 setSize()	264
3.71.2.11 slotActivated	265
3.71.2.12 slotContextMenu	265
3.71.2.13 slotItemChanged	266
3.71.2.14 slotSelectionChanged	266
3.72 YQTreeItem Class Reference	267
3.72.1 Detailed Description	268
3.72.2 Member Function Documentation	268
3.72.2.1 key()	268
3.72.2.2 setOpen()	269
3.73 YQUI Class Reference	269
3.73.1 Detailed Description	273
3.73.2 Member Function Documentation	273
3.73.2.1 applicationTitle()	273
3.73.2.2 askConfigureLogging()	273
3.73.2.3 askSaveLogs()	274
3.73.2.4 blockEvents()	274
3.73.2.5 consumePendingEvent()	275
3.73.2.6 createOptionalWidgetFactory()	275
3.73.2.7 createWidgetFactory()	275
3.73.2.8 deleteNotify()	275
3.73.2.9 eventsBlocked()	276
3.73.2.10 idleLoop()	276

---

3.73.2.11 initUI()	277
3.73.2.12 makeScreenShot()	277
3.73.2.13 pendingEvent()	278
3.73.2.14 probeX11Display()	278
3.73.2.15 raiseFatalError()	278
3.73.2.16 runPkgSelection()	278
3.73.2.17 sendEvent()	279
3.73.2.18 uiThreadDestructor()	279
3.73.2.19 yqApp()	279
3.74 YQUISignalReceiver Class Reference	280
3.74.1 Detailed Description	280
3.75 YQWidgetCaption Class Reference	281
3.75.1 Detailed Description	282
3.75.2 Constructor & Destructor Documentation	282
3.75.2.1 YQWidgetCaption()	282
3.75.3 Member Function Documentation	282
3.75.3.1 setText()	282
3.76 YQWidgetFactory Class Reference	283
3.76.1 Detailed Description	284
3.76.2 Constructor & Destructor Documentation	285
3.76.2.1 YQWidgetFactory()	285
3.77 YQWizard Class Reference	285
3.77.1 Detailed Description	291
3.77.2 Member Function Documentation	291
3.77.2.1 addMenu()	291
3.77.2.2 addMenuEntry()	291
3.77.2.3 addMenuSeparator()	292
3.77.2.4 addStep()	292
3.77.2.5 addStepHeading()	292
3.77.2.6 addSubMenu()	293
3.77.2.7 addTreeItem()	293
3.77.2.8 backButton()	294
3.77.2.9 connectNotify()	294
3.77.2.10 copySteps()	294
3.77.2.11 currentStep()	295
3.77.2.12 currentTreeSelection()	295
3.77.2.13 debugLabel()	295
3.77.2.14 deleteMenus()	295
3.77.2.15 deleteSteps()	296

3.77.2.16 deleteTreeItems()	296
3.77.2.17 direction()	296
3.77.2.18 disconnectNotify()	297
3.77.2.19 eventFilter()	297
3.77.2.20 findStep()	297
3.77.2.21 findTreeItem()	298
3.77.2.22 hideReleaseNotesButton()	298
3.77.2.23 nextClicked	298
3.77.2.24 preferredHeight()	299
3.77.2.25 preferredWidth()	299
3.77.2.26 retranslateInternalButtons()	299
3.77.2.27 selectTreeItem()	300
3.77.2.28 sendTreeEvent	300
3.77.2.29 setButtonLabel()	301
3.77.2.30 setCurrentStep()	301
3.77.2.31 setDialogHeading()	302
3.77.2.32 setDialogIcon()	302
3.77.2.33 setDialogTitle()	302
3.77.2.34 setHelpText()	303
3.77.2.35 setSize()	303
3.77.2.36 showHelp	303
3.77.2.37 showHotkeys	304
3.77.2.38 showReleaseNotesButton()	304
3.77.2.39 showSteps	304
3.77.2.40 showTree	305
3.77.2.41 stepsList()	305
3.77.2.42 treeSelectionChanged	305
3.77.2.43 updateSteps()	306
3.77.2.44 workArea()	306
3.78 YQWizardButton Class Reference	307
3.78.1 Detailed Description	308
3.78.2 Member Function Documentation	308
3.78.2.1 preferredHeight()	308
3.78.2.2 preferredWidth()	309
3.78.2.3 setSize()	309
3.78.2.4 show()	309

# Chapter 1

## Hierarchical Index

### 1.1 Class Hierarchy

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

YQTimezoneSelectorPrivate::Location . . . . .	10
QCheckBox	
YQCheckBox . . . . .	87
QDialog	
QY2HelpDialog . . . . .	33
QY2RelNotesDialog . . . . .	44
QY2StyleEditor . . . . .	45
QFrame	
BusyBar . . . . .	9
YQBarGraph . . . . .	77
YQBusyIndicator . . . . .	81
YQComboBox . . . . .	95
YQDateField . . . . .	105
YQDownloadProgress . . . . .	121
YQInputField . . . . .	143
YQIntField . . . . .	149
YQLogView . . . . .	160
YQMultiLineEdit . . . . .	172
YQMultiSelectionBox . . . . .	185
YQProgressBar . . . . .	202
YQRichText . . . . .	219
YQSelectionBox . . . . .	223
YQSlider . . . . .	232
YQTable . . . . .	241
YQTimeField . . . . .	251
YQTimezoneSelector . . . . .	255
YQTree . . . . .	259
QGroupBox	
YQCheckBoxFrame . . . . .	91
YQFrame . . . . .	131
QItemDelegate	

QY2DiskUsagePercentageltem . . . . .	32
QLabel	
YQImage . . . . .	140
YQLabel . . . . .	154
YQWidgetCaption . . . . .	281
QLineEdit	
YQRawLineEdit . . . . .	215
QObject	
QY2Styler . . . . .	46
YQApplication . . . . .	64
YQContextMenu . . . . .	102
YQUISignalReceiver . . . . .	280
QRadioButton	
YQRadioButton . . . . .	209
QSplitter	
YQWizard . . . . .	285
QTextBrowser	
YQTextBrowser . . . . .	250
QTextEdit	
MyTextEdit . . . . .	11
QTreeWidget	
QY2ListView . . . . .	35
QY2DiskUsageList . . . . .	22
QTreeWidgetItem	
QY2ListViewItem . . . . .	41
QY2CheckListItem . . . . .	15
QY2DiskUsageListItem . . . . .	25
YQTableListViewItem . . . . .	248
YQWizard::TreeItem . . . . .	59
YQMultiSelectionBoxItem . . . . .	192
YQTreeItem . . . . .	267
QValidator	
QY2CharValidator . . . . .	12
QWidget	
QY2ComboTabWidget . . . . .	18
YQAlignment . . . . .	61
YQButtonBox . . . . .	85
YQDialog . . . . .	109
YQDumbTab . . . . .	125
YQEmpty . . . . .	129
YQGenericButton . . . . .	133
YQPushButton . . . . .	205
YQWizardButton . . . . .	307
YQLayoutBox . . . . .	157
YQMainWinDock . . . . .	163
YQMenuButton . . . . .	169
YQMultiProgressMeter . . . . .	177
YQPartitionSplitter . . . . .	198
YQRadioButtonGroup . . . . .	213
YQReplacePoint . . . . .	217
YQSpacing . . . . .	238
YQSquash . . . . .	239
YQWizard::Step . . . . .	56
YQWizard::StepHeading . . . . .	58

YAlignment	
YQAlignment	61
YApplication	
YQApplication	64
YBarGraph	
YQBarGraph	77
YBusyIndicator	
YQBusyIndicator	81
YButtonBox	
YQButtonBox	85
YCheckBox	
YQCheckBox	87
YCheckBoxFrame	
YQCheckBoxFrame	91
YComboBox	
YQComboBox	95
YContextMenu	
YQContextMenu	102
YDateField	
YQDateField	105
YDialog	
YQDialog	109
YDownloadProgress	
YQDownloadProgress	121
YDumbTab	
YQDumbTab	125
YEmpty	
YQEmpty	129
YFrame	
YQFrame	131
YGraphPlugin	
YQGraphPluginStub	138
YImage	
YQImage	140
YInputField	
YQInputField	143
YIntField	
YQIntField	149
YLabel	
YQLabel	154
YLayoutBox	
YQLayoutBox	157
YLogView	
YQLogView	160
YMenuButton	
YQMenuButton	169
YMultiLineEdit	
YQMultiLineEdit	172
YMultiProgressMeter	
YQMultiProgressMeter	177
YMultiSelectionBox	
YQMultiSelectionBox	185
YOptionalWidgetFactory	
YQOptionalWidgetFactory	193

YPackageSelectorPlugin	
YQPackageSelectorPluginStub . . . . .	196
YPartitionSplitter	
YQPartitionSplitter . . . . .	198
YProgressBar	
YQProgressBar . . . . .	202
YPushButton	
YQGenericButton . . . . .	133
YQGraphPluginIf . . . . .	138
YQPackageSelectorPluginIf . . . . .	195
YQSignalBlocker . . . . .	231
YQTimezoneSelectorPrivate . . . . .	258
YRadioButton	
YQRadioButton . . . . .	209
YRadioButtonGroup	
YQRadioButtonGroup . . . . .	213
YReplacePoint	
YQReplacePoint . . . . .	217
YRichText	
YQRichText . . . . .	219
YSelectionBox	
YQSelectionBox . . . . .	223
YSlider	
YQSlider . . . . .	232
YSpacing	
YQSpacing . . . . .	238
YSquash	
YQSquash . . . . .	239
YTable	
YQTable . . . . .	241
YTimeField	
YQTimeField . . . . .	251
YTimezoneSelector	
YQTimezoneSelector . . . . .	255
YTree	
YQTree . . . . .	259
YUI	
YQUI . . . . .	269
YWidgetFactory	
YQWidgetFactory . . . . .	283
YWizard	
YQWizard . . . . .	285



## Chapter 2

# Class Index

### 2.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

<a href="#">BusyBar</a>	9
<a href="#">YQTimezoneSelectorPrivate::Location</a>	10
<a href="#">MyTextEdit</a>	11
<a href="#">QY2CharValidator</a>	12
<a href="#">QY2CheckListItem</a>	
Enhanced QCheckListItem	15
<a href="#">QY2ComboTabWidget</a>	
This widget is very much like a QTabWidget, but it uses a combo box above the tab pages rather than a ( scrolled ) one-line row of tabs	18
<a href="#">QY2DiskUsageList</a>	
Generic scrollable list of disk usage for any number of partitions	22
<a href="#">QY2DiskUsageListItem</a>	
Abstract base class for one partition ( mount point ) to display in a <a href="#">QY2DiskUsageList</a>	25
<a href="#">QY2DiskUsagePercentageltem</a>	32
<a href="#">QY2HelpDialog</a>	33
<a href="#">QY2ListView</a>	
Enhanced QTreeWidget	35
<a href="#">QY2ListViewItem</a>	
Enhanced QTreeWidgetItem	41
<a href="#">QY2RelNotesDialog</a>	44
<a href="#">QY2StyleEditor</a>	
Stylesheet Editor Dialog	45
<a href="#">QY2Styler</a>	46
<a href="#">YQWizard::Step</a>	
Helper class to represent a wizard step internally	56
<a href="#">YQWizard::StepHeading</a>	
Helper class to represent a wizard step heading internally	58
<a href="#">YQWizard::Treeltem</a>	
Helper class for wizard tree item	59
<a href="#">YQAlignment</a>	61
<a href="#">YQApplication</a>	64

YQBarGraph	77
YQBusyIndicator	81
YQButtonBox	85
YQCheckBox	87
YQCheckBoxFrame	91
YQComboBox	95
YQContextMenu	102
YQDateField	105
YQDialog	109
YQDownloadProgress	121
YQDumbTab	125
YQEmpty	129
YQFrame	131
YQGenericButton	
Abstract base class for push button and similar widgets - all that can become a YQDialog's "default button"	133
YQGraphPluginIf	138
YQGraphPluginStub	138
YQImage	140
YQInputField	143
YQIntField	149
YQLabel	154
YQLayoutBox	157
YQLogView	160
YQMainWinDock	
Container window for YQDialogs of type YMainWindowDialog:	163
YQMenuButton	169
YQMultiLineEdit	
MultiLineEdit - an input area for multi-line text	172
YQMultiProgressMeter	177
YQMultiSelectionBox	185
YQMultiSelectionBoxItem	192
YQOptionalWidgetFactory	
Widget factory for optional ("special") widgets	193
YQPackageSelectorPluginIf	195
YQPackageSelectorPluginStub	196
YQPartitionSplitter	198
YQProgressBar	202
YQPushButton	205
YQRadioButton	209
YQRadioButtonGroup	213
YQRawLineEdit	
Helper class that can obtain the CapsLock status, too	215
YQReplacePoint	217
YQRichText	219
YQSelectionBox	223
YQSignalBlocker	
Helper class to block Qt signals for QWidgets or QObjects as long as this object exists	231
YQSlider	232
YQSpacing	238
YQSquash	239
YQTable	241
YQTableListViewItem	
Visual representation of a YTableItem	248

<a href="#">YQTextBrowser</a>	
Helper class - needed to have the benefits of both QVBox as the base class for <a href="#">YQRichText</a> so uniform margins can be implemented (outside, not inside the scroll window as QTextBrowser normally does it) and QTextBrowser's features (enable hyperlinks) without removing the text each time the user clicks	250
<a href="#">YQTimeField</a>	251
<a href="#">YQTimezoneSelector</a>	255
<a href="#">YQTimezoneSelectorPrivate</a>	258
<a href="#">YQTree</a>	259
<a href="#">YQTreeItem</a>	267
<a href="#">YQUI</a>	269
<a href="#">YQUISignalReceiver</a>	
Helper class that acts as a Qt signal receiver for <a href="#">YQUI</a>	280
<a href="#">YQWidgetCaption</a>	
Helper class for captions (labels) above a widget: Takes care of hiding itself when its text is empty and showing when its text becomes non-empty	281
<a href="#">YQWidgetFactory</a>	
Concrete widget factory for mandatory widgets	283
<a href="#">YQWizard</a>	285
<a href="#">YQWizardButton</a>	307

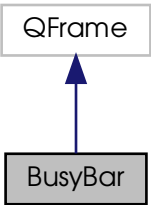


# Chapter 3

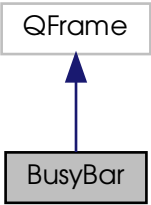
## Class Documentation

### 3.1 BusyBar Class Reference

Inheritance diagram for BusyBar:



Collaboration diagram for BusyBar:



## Public Member Functions

- **BusyBar** (QWidget \*parent)
- void **run** ()  
*start moving bar animation*
- void **stop** ()  
*stop moving bar animation*

## Protected Member Functions

- void **paintEvent** (QPaintEvent \*event)

### 3.1.1 Detailed Description

Definition at line 134 of file [YQBusyIndicator.h](#).

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

- /usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQBusyIndicator.h
- /usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQBusyIndicator.cc

## 3.2 YQTimezoneSelectorPrivate::Location Struct Reference

### Public Member Functions

- bool **operator<** (const [Location](#) &l2) const

### Public Attributes

- QString **country**
- double **latitude**
- double **longitude**
- QString **zone**
- QString **comment**
- QString **tip**
- QPoint **pix\_pos**

### 3.2.1 Detailed Description

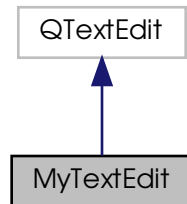
Definition at line 58 of file [YQTimezoneSelector.cc](#).

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

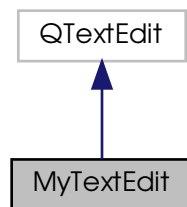
- /usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQTimezoneSelector.cc

### 3.3 MyTextEdit Class Reference

Inheritance diagram for MyTextEdit:



Collaboration diagram for MyTextEdit:



#### Signals

- void **resized** ()

#### Public Member Functions

- **MyTextEdit** (QWidget \*parent)

#### Protected Member Functions

- void **resizeEvent** (QResizeEvent \*event)

### 3.3.1 Detailed Description

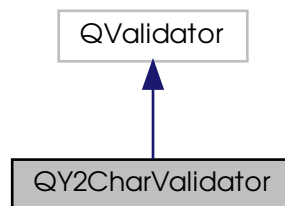
Definition at line 126 of file [YQLogView.h](#).

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

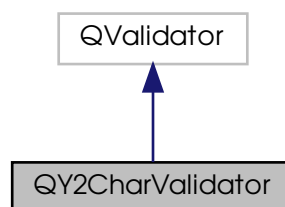
- [/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQLogView.h](#)

## 3.4 QY2CharValidator Class Reference

Inheritance diagram for QY2CharValidator:



Collaboration diagram for QY2CharValidator:





## Public Member Functions

- [QY2CharValidator](#) (const QString &initialValidChars, QObject \*parent=0, const char \*name=0)  
*Constructor.*
- [~QY2CharValidator](#) ()  
*Destructor.*
- virtual State [validate](#) (QString &input, int &pos) const  
*Check user input.*
- void [setValidChars](#) (const QString &newValidChars)  
*Set the valid input characters.*
- QString [validChars](#) () const  
*Retrieve the valid input characters.*

## Protected Attributes

- QString **\_validChars**

### 3.4.1 Detailed Description

Definition at line 34 of file [QY2CharValidator.h](#).

### 3.4.2 Constructor & Destructor Documentation

#### 3.4.2.1 QY2CharValidator()

```
QY2CharValidator::QY2CharValidator (
    const QString & initialValidChars,
    QObject * parent = 0,
    const char * name = 0 )
```

Constructor.

Create a validator object that checks for input characters being any one of 'validChars'. If this is empty, any input is accepted.

Definition at line 34 of file [QY2CharValidator.cc](#).

### 3.4.3 Member Function Documentation

### 3.4.3.1 setValidChars()

```
void QY2CharValidator::setValidChars (
    const QString & newValidChars ) [inline]
```

Set the valid input characters.

If 'newValidChars' is empty, any input is accepted.

Definition at line 66 of file [QY2CharValidator.h](#).

### 3.4.3.2 validate()

```
QValidator::State QY2CharValidator::validate (
    QString & input,
    int & pos ) const [virtual]
```

Check user input.

Reimplemented from QValidator.

Definition at line 51 of file [QY2CharValidator.cc](#).

Here is the call graph for this function:



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

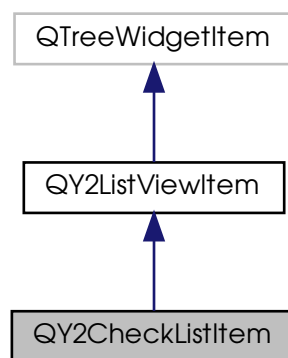
- `/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/QY2CharValidator.h`
- `/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/QY2CharValidator.cc`

## 3.5 QY2CheckListItem Class Reference

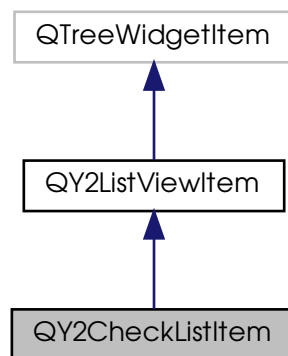
Enhanced QCheckListItem.

```
#include <QY2ListView.h>
```

Inheritance diagram for QY2CheckListItem:



Collaboration diagram for QY2CheckListItem:



## Public Member Functions

- [QY2CheckListItem](#) ([QY2ListView](#) \*parentListView, const QString &text)  
*Constructor for toplevel items.*
- [QY2CheckListItem](#) (QTreeWidgetItem \*parentItem, const QString &text)  
*Constructor for deeper level items.*
- virtual [~QY2CheckListItem](#) ()  
*Destructor.*
- virtual void [updateStatus](#) ()  
*Update this item's status.*
- virtual void [updateData](#) ()  
*Update this item's data completely.*
- int [serial](#) () const  
*Return this item's serial number.*
- void [setTextColor](#) (const QColor &col)  
*Set the text foreground color for all columns.*
- void [setBackgroundColor](#) (const QColor &col)  
*Set the text background color for all columns.*
- virtual QString [toolTip](#) (int column)  
*Returns a tool tip text for a specific column of this item.*

## Protected Attributes

- int [\\_serial](#)

### 3.5.1 Detailed Description

Enhanced QCheckListItem.

Definition at line 309 of file [QY2ListView.h](#).

### 3.5.2 Member Function Documentation

#### 3.5.2.1 serial()

```
int QY2CheckListItem::serial ( ) const [inline]
```

Return this item's serial number.

Useful for comparison functions that order by insertion sequence.

Definition at line 351 of file [QY2ListView.h](#).

### 3.5.2.2 setBackgroundColor()

```
void QY2CheckListItem::setBackgroundColor (
    const QColor & col ) [inline]
```

Set the text background color for all columns.

For more specific purposes reimplement paintCell().

Definition at line 364 of file [QY2ListView.h](#).

### 3.5.2.3 setTextColor()

```
void QY2CheckListItem::setTextColor (
    const QColor & col ) [inline]
```

Set the text foreground color for all columns.

For more specific purposes reimplement paintCell().

Definition at line 357 of file [QY2ListView.h](#).

### 3.5.2.4 toolTip()

```
virtual QString QY2CheckListItem::toolTip (
    int column ) [inline], [virtual]
```

Returns a tool tip text for a specific column of this item.

'column' is -1 if the mouse pointer is in the tree indentation area.

This default implementation does nothing.

Reimplemented from [QY2ListViewItem](#).

Definition at line 373 of file [QY2ListView.h](#).

#### 3.5.2.5 `updateData()`

```
virtual void QY2CheckListItem::updateData ( ) [inline], [virtual]
```

Update this item's data completely.

Triggered by `QY2ListView::updateAllItemData()`. Derived classes should overwrite this. This default implementation does nothing.

Reimplemented from [QY2ListViewItem](#).

Definition at line [345](#) of file [QY2ListView.h](#).

#### 3.5.2.6 `updateStatus()`

```
virtual void QY2CheckListItem::updateStatus ( ) [inline], [virtual]
```

Update this item's status.

Triggered by `QY2ListView::updateAllItemStates()`. Derived classes should overwrite this. This default implementation does nothing.

Reimplemented from [QY2ListViewItem](#).

Definition at line [337](#) of file [QY2ListView.h](#).

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

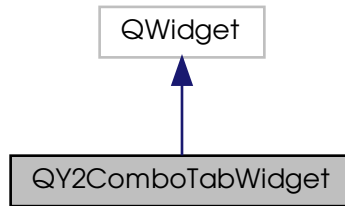
- `/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/QY2ListView.h`
- `/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/QY2ListView.cc`

## 3.6 QY2ComboTabWidget Class Reference

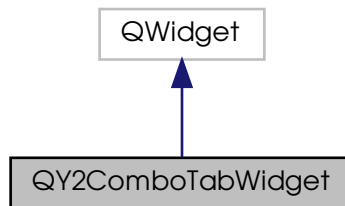
This widget is very much like a `QTabWidget`, but it uses a combo box above the tab pages rather than a ( scrolled ) one-line row of tabs.

```
#include <QY2ComboTabWidget.h>
```

Inheritance diagram for QY2ComboTabWidget:



Collaboration diagram for QY2ComboTabWidget:



### Public Slots

- void [showPage](#) (QWidget \*page)  
*Show a page.*

### Signals

- void [currentChanged](#) (QWidget \*newCurrentPage)  
*Emitted when the current page changes.*

### Public Member Functions

- [QY2ComboTabWidget](#) (const QString &combo\_box\_label, QWidget \*parent=0, const char \*name=0)  
*Constructor.*
- virtual [~QY2ComboTabWidget](#) ()  
*Destructor.*
- void [addPage](#) (const QString &page\_label, QWidget \*page)  
*Add a page.*

## Protected Slots

- void [showPageIndex](#) (int index)  
*Show a page identified by its index.*

## Protected Attributes

- QComboBox \* **combo\_box**
- QLabel \* **combo\_label**
- QStackedWidget \* **widget\_stack**
- QHash< int, QWidget \* > **pages**

### 3.6.1 Detailed Description

This widget is very much like a QTabWidget, but it uses a combo box above the tab pages rather than a ( scrolled ) one-line row of tabs.

Definition at line 45 of file [QY2ComboTabWidget.h](#).

### 3.6.2 Constructor & Destructor Documentation

#### 3.6.2.1 QY2ComboTabWidget()

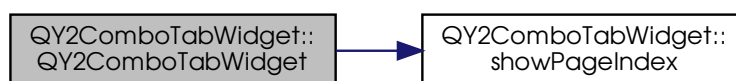
```
QY2ComboTabWidget::QY2ComboTabWidget (
    const QString & combo_box_label,
    QWidget * parent = 0,
    const char * name = 0 )
```

Constructor.

'combo\_box\_label' is the user-visible label of the combo box that is used to switch between the different tab pages.

Definition at line 47 of file [QY2ComboTabWidget.cc](#).

Here is the call graph for this function:





### 3.6.3 Member Function Documentation

#### 3.6.3.1 addPage()

```
void QY2ComboTabWidget::addPage (
    const QString & page_label,
    QWidget * page )
```

Add a page.

'page\_label' will be the user-visible combo box entry for that page.

Definition at line 93 of file [QY2ComboTabWidget.cc](#).

#### 3.6.3.2 currentChanged

```
void QY2ComboTabWidget::currentChanged (
    QWidget * newCurrentPage ) [signal]
```

Emitted when the current page changes.

NOT emitted initially for the very first page that is shown.

#### 3.6.3.3 showPage

```
void QY2ComboTabWidget::showPage (
    QWidget * page ) [slot]
```

Show a page.

Updates the combo box contents accordingly. This is an expensive operation: All combo box items are searched for the item that corresponds to this page.

Definition at line 123 of file [QY2ComboTabWidget.cc](#).

### 3.6.3.4 showPageIndex

```
void QY2ComboTabWidget::showPageIndex (
    int index ) [protected], [slot]
```

Show a page identified by its index.

Does NOT update the combo box contents.

Definition at line 105 of file [QY2ComboTabWidget.cc](#).

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

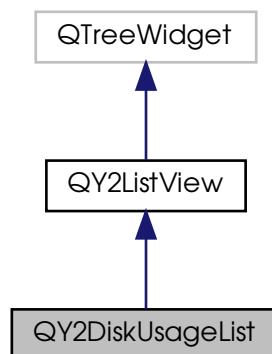
- /usr/src/RPM/BUILD/libyui-qt-2.49.11/src/QY2ComboTabWidget.h
- /usr/src/RPM/BUILD/libyui-qt-2.49.11/src/QY2ComboTabWidget.cc

## 3.7 QY2DiskUsageList Class Reference

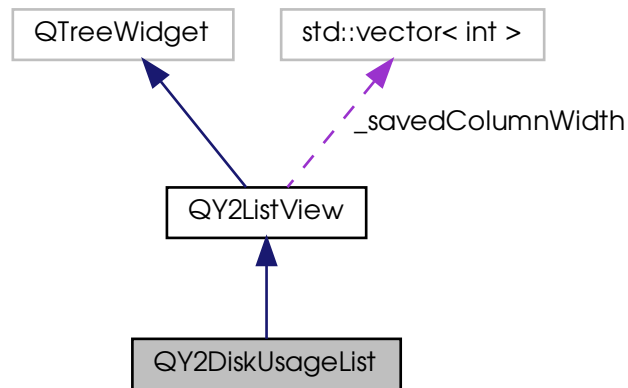
Generic scrollable list of disk usage for any number of partitions.

```
#include <QY2DiskUsageList.h>
```

Inheritance diagram for QY2DiskUsageList:



Collaboration diagram for QY2DiskUsageList:



### Public Member Functions

- [QY2DiskUsageList](#) (QWidget \*parent, bool addStdColumns=true)  
*Constructor.*
- virtual [~QY2DiskUsageList](#) ()  
*Destructor.*
- int **nameCol** () const
- int **percentageBarCol** () const
- int **usedSizeCol** () const
- int **freeSizeCol** () const
- int **totalSizeCol** () const
- int **deviceNameCol** () const
- virtual void **drawRow** (QPainter \*painter, const QStyleOptionViewItem &option, const QModelIndex &index) const
- QTreeWidgetItem \* **itemFromIndex** (const QModelIndex &index) const

### Protected Attributes

- int **\_nameCol**
- int **\_percentageBarCol**
- int **\_usedSizeCol**
- int **\_freeSizeCol**
- int **\_totalSizeCol**
- int **\_deviceNameCol**

## Additional Inherited Members

### 3.7.1 Detailed Description

Generic scrollable list of disk usage for any number of partitions.

Definition at line 42 of file [QY2DiskUsageList.h](#).

### 3.7.2 Constructor & Destructor Documentation

#### 3.7.2.1 QY2DiskUsageList()

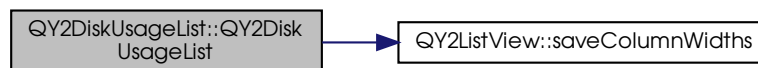
```
QY2DiskUsageList::QY2DiskUsageList (
    QWidget * parent,
    bool addStdColumns = true )
```

Constructor.

Adds a standard set of list columns if 'addStdColumns' is 'true'. Otherwise the caller is responsible for adding any columns.

Definition at line 160 of file [QY2DiskUsageList.cc](#).

Here is the call graph for this function:



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

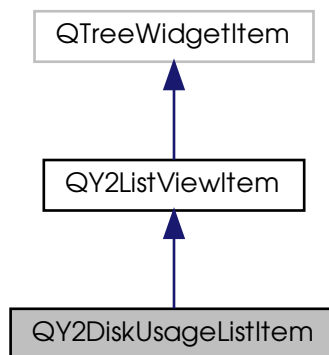
- /usr/src/RPM/BUILD/libyui-qt-2.49.11/src/QY2DiskUsageList.h
- /usr/src/RPM/BUILD/libyui-qt-2.49.11/src/QY2DiskUsageList.cc

## 3.8 QY2DiskUsageListItem Class Reference

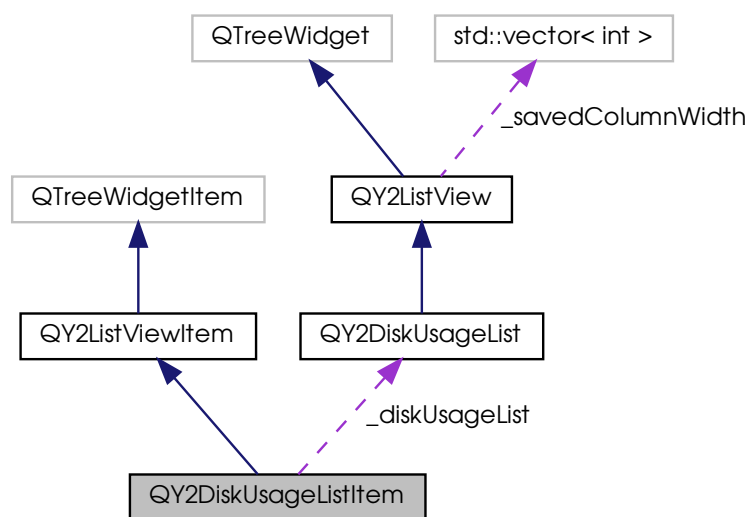
Abstract base class for one partition ( mount point ) to display in a [QY2DiskUsageList](#).

```
#include <QY2DiskUsageList.h>
```

Inheritance diagram for QY2DiskUsageListItem:



Collaboration diagram for QY2DiskUsageListItem:



## Public Member Functions

- virtual FSize [usedSize](#) () const =0  
*The currently used size of this partition.*
- virtual FSize [totalSize](#) () const =0  
*The total size of this partition.*
- virtual FSize [freeSize](#) () const  
*The current free size of this partition.*
- virtual int [usedPercent](#) () const  
*The currently used percentage ( 0..100 ) of this partition.*
- virtual QString [name](#) () const =0  
*The name to display for this partition.*
- virtual QString [deviceName](#) () const  
*The device name of this partition.*
- virtual void [updateStatus](#) ()  
*Update this item's status ( here: the numeric fields ).*
- virtual void [updateData](#) ()  
*Update this item's data completely.*
- void [setText](#) (int column, const QString &text)  
*Re-declare ordinary [setText\(\)](#) method so the compiler doesn't get confused which one to use.*
- void [setText](#) (int column, const FSize &size)  
*Set a column text via FSize.*
- virtual bool [operator<](#) (const QTreeWidgetItem &other) const  
*Comparison function used for sorting the list.*
- int [nameCol](#) () const
- int [percentageBarCol](#) () const
- int [usedSizeCol](#) () const
- int [freeSizeCol](#) () const
- int [totalSizeCol](#) () const
- int [deviceNameCol](#) () const
- void [paintPercentageBar](#) (QPainter \*painter, QStyleOptionViewItem option, const QColor &fillColor)  
*Paint a percentage bar into a QListViewItem cell.*

## Protected Member Functions

- [QY2DiskUsageListItem](#) ([QY2DiskUsageList](#) \*parent)  
*Constructor.*
- virtual [~QY2DiskUsageListItem](#) ()  
*Destructor.*
- void [init](#) (bool allFields)  
*( Re- ) initialize fields - all displayed fields ( if 'allFields' is 'true' ) or only the varying fields ( used, free, percentage ).*

## Protected Attributes

- [QY2DiskUsageList](#) \* [\\_diskUsageList](#)

### 3.8.1 Detailed Description

Abstract base class for one partition ( mount point ) to display in a [QY2DiskUsageList](#).

This class contains pure virtuals, so it cannot be used directly.

Definition at line 96 of file [QY2DiskUsageList.h](#).

### 3.8.2 Constructor & Destructor Documentation

#### 3.8.2.1 QY2DiskUsageListItem()

```
QY2DiskUsageListItem::QY2DiskUsageListItem (
    QY2DiskUsageList * parent ) [protected]
```

Constructor.

Call [updateData\(\)](#) after the constructor for the initial display update. Unfortunately, this cannot be done automatically in the constructor since it uses virtual methods that are not available yet at this point.

Definition at line 219 of file [QY2DiskUsageList.cc](#).

### 3.8.3 Member Function Documentation

#### 3.8.3.1 deviceName()

```
virtual QString QY2DiskUsageListItem::deviceName ( ) const [inline], [virtual]
```

The device name of this partition.

Derived classes may choose to reimplement this method. This default implementation returns an empty `std::string`.

Definition at line 166 of file [QY2DiskUsageList.h](#).

### 3.8.3.2 freeSize()

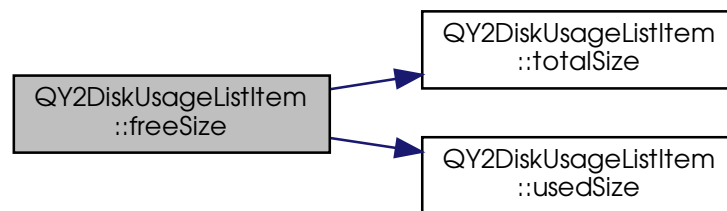
```
FSize QY2DiskUsageListItem::freeSize ( ) const [virtual]
```

The current free size of this partition.

Derived classes can choose reimplement this if it is less expensive than calculating this value each time from [usedSize\(\)](#) and [totalSize\(\)](#) which is the default implementation.

Definition at line 269 of file [QY2DiskUsageList.cc](#).

Here is the call graph for this function:



### 3.8.3.3 name()

```
virtual QString QY2DiskUsageListItem::name ( ) const [pure virtual]
```

The name to display for this partition.

It makes most sense to use the mount point here ( but this is not a requirement ). This is what will be displayed in the "Name" column.

Derived classes need to implement this method.



## 3.8.3.4 operator&lt;()

```
bool QY2DiskUsageListItem::operator< (
    const QTreeWidgetItem & otherListViewItem ) const [virtual]
```

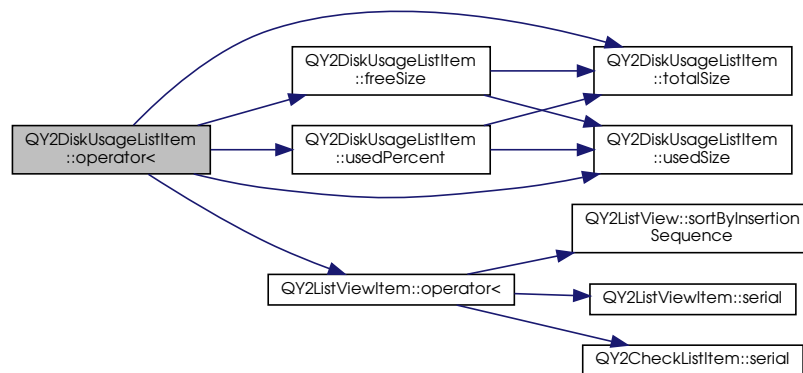
Comparison function used for sorting the list.

Reimplemented from QTreeWidgetItem

Reimplemented from [QY2ListViewItem](#).

Definition at line 309 of file [QY2DiskUsageList.cc](#).

Here is the call graph for this function:



## 3.8.3.5 paintPercentageBar()

```
void QY2DiskUsageListItem::paintPercentageBar (
    QPainter * painter,
    QStyleOptionViewItem option,
    const QColor & fillColor )
```

Paint a percentage bar into a QListViewItem cell.

Stolen from KDirStat::KDirTreeView with the author's permission.

'width' is the width of the entire cell. 'indent' is the number of pixels to indent the bar.

Stolen from KDirStat::KDirTreeView with the author's permission.

Definition at line 342 of file [QY2DiskUsageList.cc](#).

Here is the call graph for this function:



### 3.8.3.6 totalSize()

```
virtual FSize QY2DiskUsageListItem::totalSize ( ) const [pure virtual]
```

The total size of this partition.

Derived classes need to implement this method.

### 3.8.3.7 updateData()

```
void QY2DiskUsageListItem::updateData ( ) [virtual]
```

Update this item's data completely.

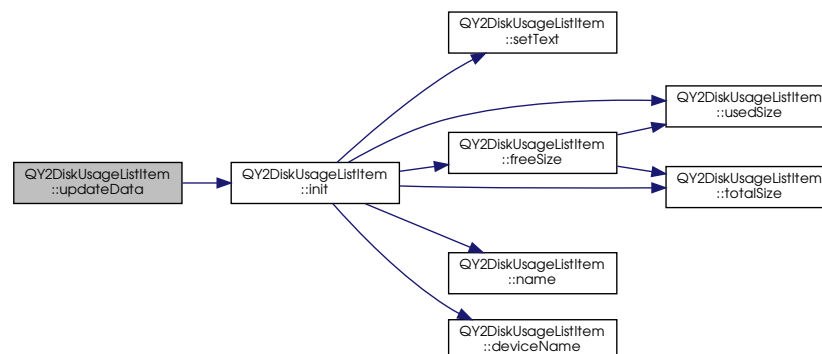
Triggered by `QY2ListView::updateAllItemData()`.

Reimplemented from [QY2ListViewItem](#).

Reimplemented from [QY2ListViewItem](#).

Definition at line 295 of file [QY2DiskUsageList.cc](#).

Here is the call graph for this function:



## 3.8.3.8 updateStatus()

```
void QY2DiskUsageListItem::updateStatus ( ) [virtual]
```

Update this item's status ( here: the numeric fields ).

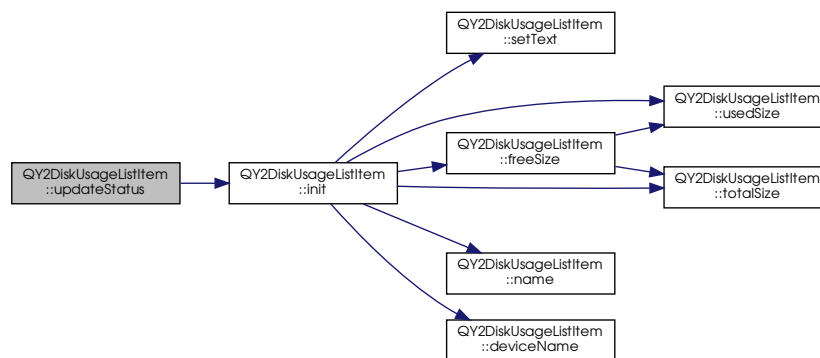
Triggered by `QY2ListView::updateAllItemStates()`.

Reimplemented from [QY2ListViewItem](#).

Reimplemented from [QY2ListViewItem](#).

Definition at line 288 of file [QY2DiskUsageList.cc](#).

Here is the call graph for this function:



## 3.8.3.9 usedPercent()

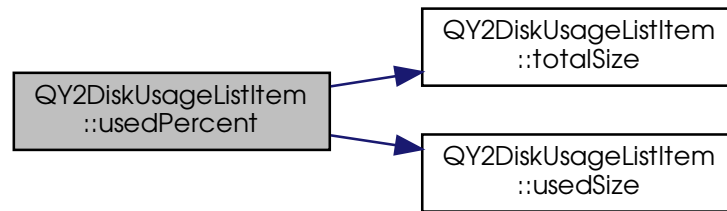
```
int QY2DiskUsageListItem::usedPercent ( ) const [virtual]
```

The currently used percentage ( 0..100 ) of this partition.

Derived classes can choose reimplement this if it is less expensive than calculating this value each time from [usedSize\(\)](#) and [totalSize\(\)](#) which is the default implementation.

Definition at line 276 of file [QY2DiskUsageList.cc](#).

Here is the call graph for this function:



### 3.8.3.10 usedSize()

```
virtual FSize QY2DiskUsageListItem::usedSize ( ) const [pure virtual]
```

The currently used size of this partition.

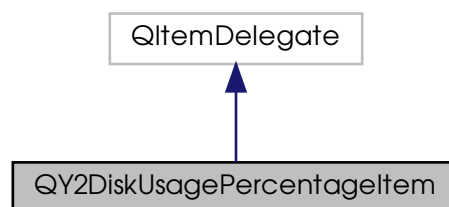
Derived classes need to implement this method.

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

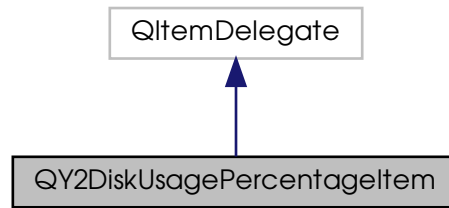
- `/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/QY2DiskUsageList.h`
- `/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/QY2DiskUsageList.cc`

## 3.9 QY2DiskUsagePercentageltem Class Reference

Inheritance diagram for QY2DiskUsagePercentageltem:



Collaboration diagram for QY2DiskUsagePercentageltem:



### Public Member Functions

- **QY2DiskUsagePercentageltem** ([QY2DiskUsageList](#) \*parent)
- virtual void **paint** (QPainter \*painter, const QStyleOptionViewItem &option, const QModelIndex &index) const

### 3.9.1 Detailed Description

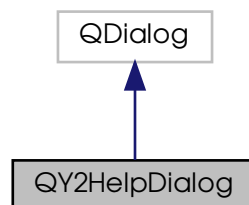
Definition at line 132 of file [QY2DiskUsageList.cc](#).

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

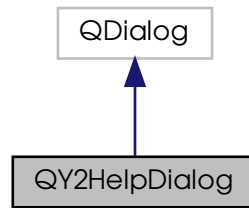
- `/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/QY2DiskUsageList.cc`

## 3.10 QY2HelpDialog Class Reference

Inheritance diagram for QY2HelpDialog:



Collaboration diagram for QY2HelpDialog:



### Public Slots

- void **searchStringChanged** (QString)

### Public Member Functions

- **QY2HelpDialog** (const QString &helpText, QWidget \*parent)
- void **setHelpText** (const QString &helpText)
- void **retranslate** ()
- QColor **getSearchResultForeground** ()
- QColor **getSearchResultBackground** ()
- void **setSearchResultForeground** (QColor pen)
- void **setSearchResultBackground** (QColor pen)

### Properties

- QColor **searchResultForeground**
- QColor **searchResultBackground**

### 3.10.1 Detailed Description

Definition at line 36 of file [QY2HelpDialog.h](#).

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

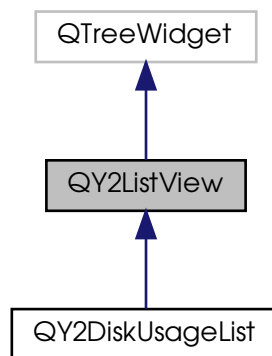
- /usr/src/RPM/BUILD/libyui-qt-2.49.11/src/QY2HelpDialog.h
- /usr/src/RPM/BUILD/libyui-qt-2.49.11/src/QY2HelpDialog.cc

## 3.11 QY2ListView Class Reference

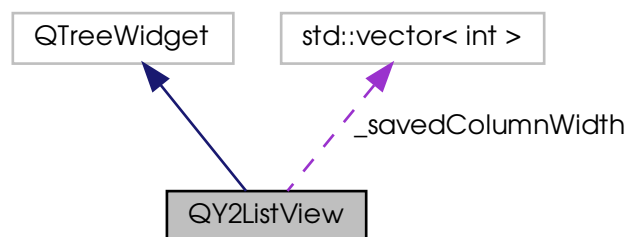
Enhanced QTreeWidget.

```
#include <QY2ListView.h>
```

Inheritance diagram for QY2ListView:



Collaboration diagram for QY2ListView:



### Public Slots

- virtual void [selectSomething](#) ()  
*Select a list entry (if there is any).*
- virtual void [clear](#) ()

*Reimplemented from Q3ListView: Adjust header sizes after clearing contents.*

- void [updateItemStates](#) ()  
*Update the status display of all list entries: Call [QY2ListViewItem::updateStatus\(\)](#) for each item.*
- void [updateItemData](#) ()  
*Update the status display of all list entries: Call [QY2ListViewItem::updateData\(\)](#) for each item.*
- void [saveColumnWidths](#) ()  
*Save the current column widths.*
- void [restoreColumnWidths](#) ()  
*Restore the column widths to what was saved previously with [saveColumnWidths\(\)](#).*

## Signals

- void [columnClicked](#) (int button, QTreeWidgetItem \*item, int col, const QPoint &pos)  
*Emitted for mouse clicks on an item.*
- void [columnDoubleClicked](#) (int button, QTreeWidgetItem \*item, int col, const QPoint &pos)  
*Emitted for mouse double clicks on an item.*

## Public Member Functions

- [QY2ListView](#) (QWidget \*parent)  
*Constructor.*
- virtual [~QY2ListView](#) ()  
*Destructor.*
- virtual QString [toolTip](#) (QTreeWidgetItem \*item, int column)  
*Returns a tool tip text for a specific column of a list item.*
- bool [sortByInsertionSequence](#) () const  
*Returns 'true' if the sort order should always be the item insertion order, 'false' if the user can change the sort order by clicking on a column header.*
- virtual void [setSortByInsertionSequence](#) (bool [sortByInsertionSequence](#))  
*Enforce sorting by item insertion order (true) or let user change sorting by clicking on a column header (false).*
- int [nextSerial](#) ()  
*Returns the next free serial number for items that want to be ordered in insertion sequence.*
- virtual QSize [minimumSizeHint](#) () const  
*Returns the minimum size required for this widget.*
- virtual bool [eventFilter](#) (QObject \*obj, QEvent \*event)  
*Event filter - inherited from QWidget.*

## Protected Slots

- void [columnWidthChanged](#) (int col, int oldSize, int newSize)  
*Internal: Handle manual column resize.*
- void [treeExpanded](#) (QTreeWidgetItem \*listViewItem)  
*Internal notification that a tree item has been expanded.*
- void [treeCollapsed](#) (QTreeWidgetItem \*listViewItem)  
*Internal notification that a tree item has been collapsed.*



## Protected Member Functions

- virtual void [mousePressEvent](#) (QMouseEvent \*e)  
*Handle mouse clicks.*
- virtual void [mouseReleaseEvent](#) (QMouseEvent \*)  
*Handle mouse clicks.*
- virtual void [mouseDoubleClickEvent](#) (QMouseEvent \*)  
*Handle mouse clicks.*

## Protected Attributes

- QTreeWidgetItem \* **\_mousePressedItem**
- int **\_mousePressedCol**
- Qt::MouseButton **\_mousePressedButton**
- std::vector< int > **\_savedColumnWidth**
- bool **\_sortByInsertionSequence**
- int **\_nextSerial**
- QY2ListViewToolTip \* **\_toolTip**
- bool **\_mouseButton1PressedInHeader**
- bool **\_finalSizeChangeExpected**

### 3.11.1 Detailed Description

Enhanced QTreeWidgetItem.

Definition at line 47 of file [QY2ListView.h](#).

### 3.11.2 Member Function Documentation

#### 3.11.2.1 columnWidthChanged

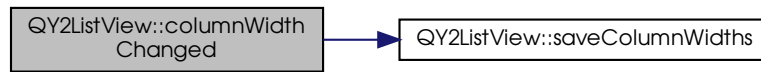
```
void QY2ListView::columnWidthChanged (  
    int col,  
    int oldSize,  
    int newSize ) [protected], [slot]
```

Internal: Handle manual column resize.

Save the user's preferred sizes so they don't get overwritten each time the list is cleared and filled with new contents.

Definition at line 289 of file [QY2ListView.cc](#).

Here is the call graph for this function:



### 3.11.2.2 `minimumSizeHint()`

```
QSize QY2ListView::minimumSizeHint ( ) const [virtual]
```

Returns the minimum size required for this widget.

Inherited from `QWidget`.

Definition at line [348](#) of file [QY2ListView.cc](#).

### 3.11.2.3 `mouseDoubleClickEvent()`

```
void QY2ListView::mouseDoubleClickEvent (
    QMouseEvent * ev ) [protected], [virtual]
```

Handle mouse clicks.

Reimplemented from `QScrollView`.

Definition at line [267](#) of file [QY2ListView.cc](#).

### 3.11.2.4 `mousePressEvent()`

```
void QY2ListView::mousePressEvent (
    QMouseEvent * e ) [protected], [virtual]
```

Handle mouse clicks.

Reimplemented from `QScrollView`.

Definition at line [212](#) of file [QY2ListView.cc](#).

### 3.11.2.5 mouseReleaseEvent()

```
void QY2ListView::mouseReleaseEvent (
    QMouseEvent * ev )    [protected], [virtual]
```

Handle mouse clicks.

Reimplemented from QScrollView.

Definition at line 237 of file [QY2ListView.cc](#).

### 3.11.2.6 selectSomething

```
void QY2ListView::selectSomething ( )    [virtual], [slot]
```

Select a list entry (if there is any).

Usually this will be the first list entry, but don't rely on that - this might change without notice. Emits signal `selectionChanged()`.

Definition at line 82 of file [QY2ListView.cc](#).

### 3.11.2.7 toolTip()

```
QString QY2ListView::toolTip (
    QTreeWidgetItem * item,
    int column )    [virtual]
```

Returns a tool tip text for a specific column of a list item.

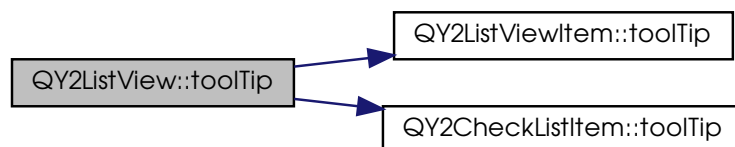
'column' is -1 if the mouse pointer is in the tree indentation area.

This default implementation tries to call `QY2ListViewItem::toolTip( column )` or `QY2CheckListItem::toolTip( column )`, respectively if 'item' is a subclass of either.

Derived classes may handle this differently.

Definition at line 144 of file [QY2ListView.cc](#).

Here is the call graph for this function:



### 3.11.2.8 updateItemData

```
void QY2ListView::updateItemData ( ) [slot]
```

Update the status display of all list entries: Call [QY2ListViewItem::updateData\(\)](#) for each item.

This is an expensive operation.

Definition at line 127 of file [QY2ListView.cc](#).

Here is the call graph for this function:



### 3.11.2.9 updateItemStates

```
void QY2ListView::updateItemStates ( ) [slot]
```

Update the status display of all list entries: Call [QY2ListViewItem::updateStatus\(\)](#) for each item.

This is an expensive operation.

Definition at line 110 of file [QY2ListView.cc](#).

Here is the call graph for this function:



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

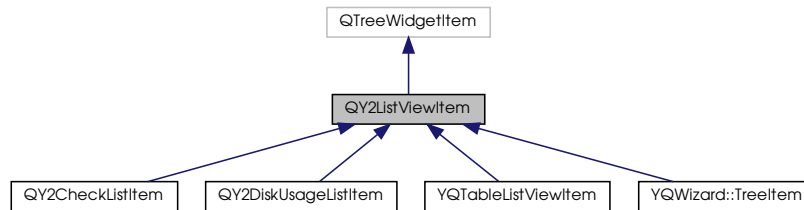
- `/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/QY2ListView.h`
- `/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/QY2ListView.cc`

## 3.12 QY2ListViewItem Class Reference

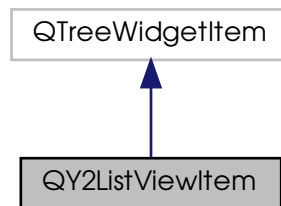
Enhanced QTreeWidgetItem.

```
#include <QY2ListView.h>
```

Inheritance diagram for QY2ListViewItem:



Collaboration diagram for QY2ListViewItem:



### Public Member Functions

- [QY2ListViewItem](#) ([QY2ListView](#) \*parentListView, const QString &text=QString::null)  
*Constructor for toplevel items.*
- [QY2ListViewItem](#) (QTreeWidgetItem \*parentItem, const QString &text=QString::null)  
*Constructor for deeper level items.*
- virtual [~QY2ListViewItem](#) ()  
*Destructor.*
- virtual void [updateStatus](#) ()  
*Update this item's status.*
- virtual void [updateData](#) ()  
*Update this item's data completely.*

- virtual bool `operator<` (const QTreeWidgetItem &other) const  
*Comparison function used for sorting the list.*
- int `serial` () const  
*Return this item's serial number.*
- virtual QString `toolTip` (int column)  
*Returns a tool tip text for a specific column of this item.*

### Protected Attributes

- int `_serial`
- QColor `_textColor`
- QColor `_backgroundColor`

### 3.12.1 Detailed Description

Enhanced QTreeWidgetItem.

Definition at line 233 of file [QY2ListView.h](#).

### 3.12.2 Member Function Documentation

#### 3.12.2.1 `operator<()`

```
bool QY2ListViewItem::operator< (
    const QTreeWidgetItem & other ) const [virtual]
```

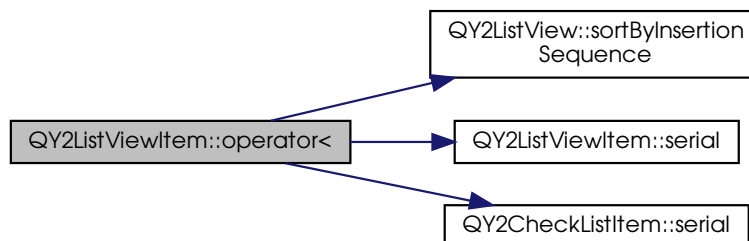
Comparison function used for sorting the list.

Reimplemented from QTreeWidgetItem

Reimplemented in [QY2DiskUsageListItem](#).

Definition at line 396 of file [QY2ListView.cc](#).

Here is the call graph for this function:



#### 3.12.2.2 serial()

```
int QY2ListViewItem::serial ( ) const [inline]
```

Return this item's serial number.

Useful for comparison functions that order by insertion sequence.

Definition at line 281 of file [QY2ListView.h](#).

#### 3.12.2.3 toolTip()

```
virtual QString QY2ListViewItem::toolTip (
    int column ) [inline], [virtual]
```

Returns a tool tip text for a specific column of this item.

'column' is -1 if the mouse pointer is in the tree indentation area.

This default implementation does nothing.

Reimplemented in [QY2CheckListItem](#).

Definition at line 289 of file [QY2ListView.h](#).

#### 3.12.2.4 updateData()

```
virtual void QY2ListViewItem::updateData ( ) [inline], [virtual]
```

Update this item's data completely.

Triggered by [QY2ListView::updateAllItemData\(\)](#). Derived classes should overwrite this. This default implementation does nothing.

Reimplemented in [QY2CheckListItem](#), and [QY2DiskUsageListItem](#).

Definition at line 269 of file [QY2ListView.h](#).

### 3.12.2.5 `updateStatus()`

```
virtual void QY2ListViewItem::updateStatus ( ) [inline], [virtual]
```

Update this item's status.

Triggered by `QY2ListView::updateAllItemStates()`. Derived classes should overwrite this. This default implementation does nothing.

Reimplemented in [QY2CheckListItem](#), and [QY2DiskUsageListItem](#).

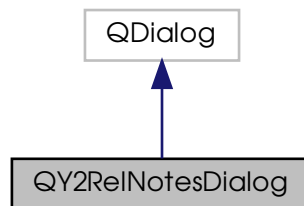
Definition at line 261 of file [QY2ListView.h](#).

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

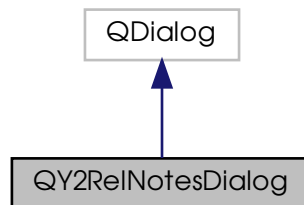
- `/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/QY2ListView.h`
- `/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/QY2ListView.cc`

## 3.13 QY2RelNotesDialog Class Reference

Inheritance diagram for QY2RelNotesDialog:



Collaboration diagram for QY2RelNotesDialog:





### Public Slots

- void **tabChanged** (int index)

### Public Member Functions

- **QY2RelNotesDialog** (QWidget \*parent)
- void **setRelNotes** (const std::map< std::string, std::string > &relnotes)
- void **retranslate** ()

### Protected Attributes

- QVBoxLayout \* **vboxLayout**
- QTabBar \* **tabBar**
- QTextBrowser \* **textBrowser**
- QHBoxLayout \* **hboxLayout**
- QPushButton \* **closeButton**

#### 3.13.1 Detailed Description

Definition at line 39 of file [QY2RelNotesDialog.h](#).

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

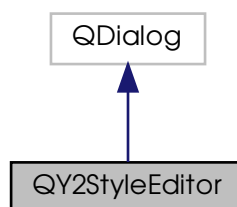
- /usr/src/RPM/BUILD/libyui-qt-2.49.11/src/QY2RelNotesDialog.h
- /usr/src/RPM/BUILD/libyui-qt-2.49.11/src/QY2RelNotesDialog.cc

## 3.14 QY2StyleEditor Class Reference

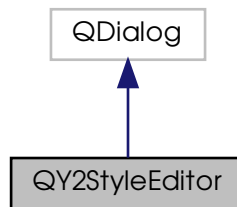
Stylesheet Editor Dialog.

```
#include <QY2StyleEditor.h>
```

Inheritance diagram for QY2StyleEditor:



Collaboration diagram for QY2StyleEditor:



### Public Member Functions

- [QY2StyleEditor](#) (QWidget \*parent)  
*Constructor.*
- [~QY2StyleEditor](#) ()  
*Destructor.*

### 3.14.1 Detailed Description

Stylesheet Editor Dialog.

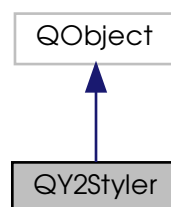
Definition at line 38 of file [QY2StyleEditor.h](#).

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

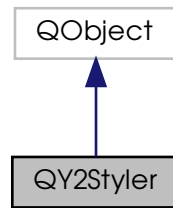
- /usr/src/RPM/BUILD/libyui-qt-2.49.11/src/QY2StyleEditor.h
- /usr/src/RPM/BUILD/libyui-qt-2.49.11/src/QY2StyleEditor.cc

## 3.15 QY2Styler Class Reference

Inheritance diagram for QY2Styler:



Collaboration diagram for QY2Styler:



## Public Member Functions

- bool [styleSheetExists](#) (const QString &file)  
*Determines if an style sheet exists.*
- bool [loadStyleSheet](#) (const QString &file)  
*Loads and apply a style sheet from a file.*
- void [setStyleSheet](#) (const QString &text)  
*Applies a style sheet from a string.*
- bool [loadDefaultStyleSheet](#) ()  
*Loads the default stylesheet.*
- bool [loadAlternateStyleSheet](#) ()  
*Loads the alternate stylesheet.*
- QString [themeDir](#) () const  
*Returns the path to the style sheets directory.*
- void [registerWidget](#) (QWidget \*widget)  
*Registers a widget and applies the style sheet.*
- void [unregisterWidget](#) (QWidget \*widget)  
*Unregisters a widget.*
- void [registerChildWidget](#) (QWidget \*parent, QWidget \*widget)  
*Registers a child widget.*
- QString [textStyle](#) () const
- void [setDefaultStyleSheet](#) (const QString &styleSheet)  
*Set style sheet for the default theme.*
- void [setAlternateStyleSheet](#) (const QString &styleSheet)  
*Set style sheet for the alternate theme.*
- void [toggleAlternateStyleSheet](#) ()  
*Toggle between default/alternate style sheets.*
- bool [usingAlternateStyleSheet](#) ()  
*Determines if the alternate style is being used.*
- bool [updateRendering](#) (QWidget \*wid)

## Static Public Member Functions

- static [QY2Styler](#) \* **styler** ()

## Protected Member Functions

- [QY2Styler](#) (QObject \*parent, const QString &defaultStyleSheet="", const QString &alternateStyleSheet="")

*Constructor.*

- void **renderParent** (QWidget \*wid)
- QImage **getScaled** (const QString name, const QSize &size)
- void [processUrls](#) (QString &text)

*Search and replace some self-defined macros in the style sheet.*

- const QString [buildStyleSheet](#) (QString content)

*Build a stylesheet from a string.*

- const QString [buildStyleSheet](#) (QString content, QStringList &alreadyImportedFileNames)

*Build a stylesheet from a string.*

- const QString [buildStyleSheetFromFile](#) (const QString &filename, QStringList &alreadyImportedFileNames)

*Build a stylesheet from a file.*

- bool **eventFilter** (QObject \*obj, QEvent \*ev)

## Protected Attributes

- QString **\_currentStyleSheet**
- QString **\_defaultStyleSheet** = DEFAULT\_STYLE\_SHEET
- QString **\_alternateStyleSheet** = HIGH\_CONTRAST\_STYLE\_SHEET
- bool **\_usingAlternateStyleSheet** = false

### 3.15.1 Detailed Description

Definition at line 38 of file [QY2Styler.h](#).

### 3.15.2 Constructor & Destructor Documentation

## 3.15.2.1 QY2Styler()

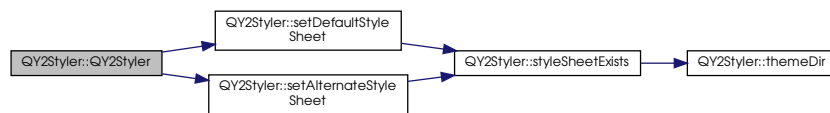
```
QY2Styler::QY2Styler (
    QObject * parent,
    const QString & defaultStyleSheet = "",
    const QString & alternateStyleSheet = "" ) [protected]
```

Constructor.

Use the static `styler()` function instead to return the singleton for this class.

Definition at line 55 of file [QY2Styler.cc](#).

Here is the call graph for this function:



## 3.15.3 Member Function Documentation

## 3.15.3.1 buildStyleSheet()

```
const QString QY2Styler::buildStyleSheet (
    QString content,
    QStringList & alreadyImportedFilenames ) [protected]
```

Build a stylesheet from a string.

Receives a list of already imported files.

Definition at line 152 of file [QY2Styler.cc](#).

Here is the call graph for this function:



### 3.15.3.2 buildStyleSheetFromFile()

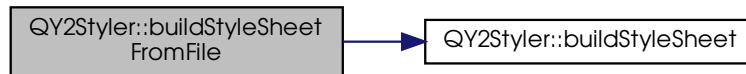
```
const QString QY2Styler::buildStyleSheetFromFile (
    const QString & filename,
    QStringList & alreadyImportedFileNames ) [protected]
```

Build a stylesheet from a file.

Receives a list of already imported files.

Definition at line 165 of file [QY2Styler.cc](#).

Here is the call graph for this function:



### 3.15.3.3 loadAlternateStyleSheet()

```
bool QY2Styler::loadAlternateStyleSheet ( )
```

Loads the alternate stylesheet.

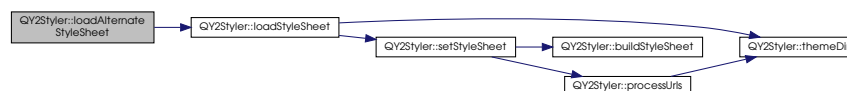
The alternate stylesheet is determined by the environment variable Y2ALTSTYLE. If this variable is not set, the `HIG_COLOR_STYLE_SHEET` style sheet will be used.

#### Returns

true if the stylesheet was loaded; false otherwise.

Definition at line 120 of file [QY2Styler.cc](#).

Here is the call graph for this function:



## 3.15.3.4 loadDefaultStyleSheet()

```
bool QY2Styler::loadDefaultStyleSheet ( )
```

Loads the default stylesheet.

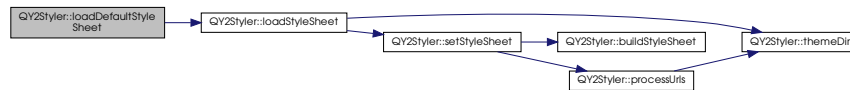
The default stylesheet is determined by the environment variable Y2STYLE. If this variable is not set, the DEFAULT\_STYLE\_SHEET style sheet will be used.

## Returns

true if the stylesheet was loaded; false otherwise.

Definition at line 113 of file [QY2Styler.cc](#).

Here is the call graph for this function:



## 3.15.3.5 loadStyleSheet()

```
bool QY2Styler::loadStyleSheet (
    const QString & file )
```

Loads and apply a style sheet from a file.

## Parameters

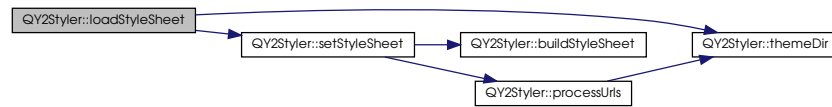
<i>file</i>	Filename. It should live in the <a href="#">themeDir()</a> directory.
-------------	---

## Returns

true if the file was found (and applied); false otherwise.

Definition at line 127 of file [QY2Styler.cc](#).

Here is the call graph for this function:



### 3.15.3.6 processUrls()

```
void QY2Styler::processUrls (
    QString & text ) [protected]
```

Search and replace some self-defined macros in the style sheet.

Among other things, expands the file name inside url( filename.png ) in the style sheet with the full path.

Definition at line [201](#) of file [QY2Styler.cc](#).

Here is the call graph for this function:



### 3.15.3.7 registerChildWidget()

```
void QY2Styler::registerChildWidget (
    QWidget * parent,
    QWidget * widget )
```

Registers a child widget.

#### Parameters

<i>parent</i>	Parent widget.
<i>widget</i>	Widget to register.



Definition at line 284 of file [QY2Styler.cc](#).

### 3.15.3.8 registerWidget()

```
void QY2Styler::registerWidget (
    QWidget * widget )
```

Registers a widget and applies the style sheet.

#### Parameters

<i>widget</i>	Widget to register.
---------------	---------------------

Definition at line 268 of file [QY2Styler.cc](#).

### 3.15.3.9 setAlternateStyleSheet()

```
void QY2Styler::setAlternateStyleSheet (
    const QString & styleSheet )
```

Set style sheet for the alternate theme.

If the style sheet does not exists, it won't be changed.

#### Parameters

<i>styleSheet</i>	Style sheet file name
-------------------	-----------------------

Definition at line 105 of file [QY2Styler.cc](#).

Here is the call graph for this function:



### 3.15.3.10 setDefaultStyleSheet()

```
void QY2Styler::setDefaultStyleSheet (
    const QString & styleSheet )
```

Set style sheet for the default theme.

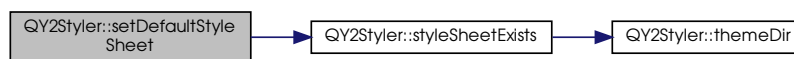
If the style sheet does not exists, it won't be changed.

#### Parameters

<i>styleSheet</i>	Style sheet file name
-------------------	-----------------------

Definition at line 97 of file [QY2Styler.cc](#).

Here is the call graph for this function:



### 3.15.3.11 setStyleSheet()

```
void QY2Styler::setStyleSheet (
    const QString & text )
```

Applies a style sheet from a string.

#### Parameters

<i>text</i>	Style sheet content.
-------------	----------------------

Definition at line 177 of file [QY2Styler.cc](#).

Here is the call graph for this function:



#### 3.15.3.12 `styleSheetExists()`

```
bool QY2Styler::styleSheetExists (
    const QString & file )
```

Determines if an style sheet exists.

##### Parameters

<i>file</i>	Filename. It should live in the <a href="#">themeDir()</a> directory.
-------------	---

##### Returns

true if the file was found; false otherwise.

Definition at line 91 of file [QY2Styler.cc](#).

Here is the call graph for this function:



#### 3.15.3.13 `unregisterWidget()`

```
void QY2Styler::unregisterWidget (
    QWidget * widget )
```

Unregisters a widget.

**Parameters**

<i>widget</i>	Widget to unregister.
---------------	-----------------------

Definition at line 277 of file [QY2Styler.cc](#).

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

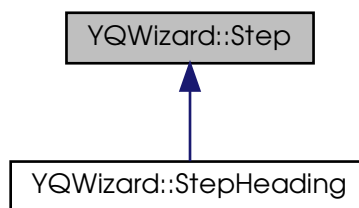
- /usr/src/RPM/BUILD/libyui-qt-2.49.11/src/QY2Styler.h
- /usr/src/RPM/BUILD/libyui-qt-2.49.11/src/QY2Styler.cc

### 3.16 YQWizard::Step Class Reference

Helper class to represent a wizard step internally.

```
#include <YQWizard.h>
```

Inheritance diagram for YQWizard::Step:

**Public Types**

- enum **Status** { **Unset**, **Todo**, **Current**, **Done** }

## Public Member Functions

- **Step** (const QString &name="", const QString &id="")
- virtual [~Step](#) ()  
*Destructor.*
- virtual bool **isHeading** () const
- QString **name** () const
- QLabel \* **statusLabel** () const
- QLabel \* **nameLabel** () const
- bool **isEnabled** () const
- const QStringList & **id** () const
- void **addID** (const QString &id)
- virtual bool **hasID** (const QString &id)
- void **setStatusLabel** (QLabel \*label)
- void **setNameLabel** (QLabel \*label)
- void **setEnabled** (bool enabled)
- void **deleteLabels** ()
- void [setStatus](#) (Status s)

*Set text color and status icon for one wizard step.*

## Protected Attributes

- QString **\_name**
- QLabel \* **\_statusLabel**
- QLabel \* **\_nameLabel**
- bool **\_enabled**
- QStringList **\_idList**
- Status **\_status**

### 3.16.1 Detailed Description

Helper class to represent a wizard step internally.

Definition at line 625 of file [YQWizard.h](#).

### 3.16.2 Constructor & Destructor Documentation

#### 3.16.2.1 ~Step()

```
YQWizard::Step::~Step ( ) [virtual]
```

Destructor.

Intentionally not deleting the widgets.

Definition at line 1449 of file [YQWizard.cc](#).

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

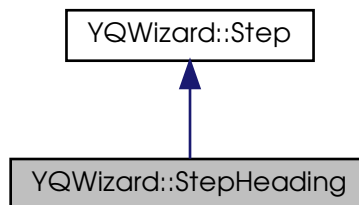
- /usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQWizard.h
- /usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQWizard.cc

### 3.17 YQWizard::StepHeading Class Reference

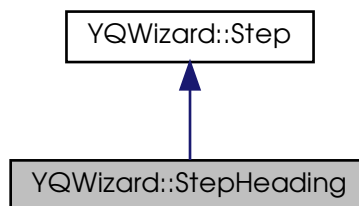
Helper class to represent a wizard step heading internally.

```
#include <YQWizard.h>
```

Inheritance diagram for YQWizard::StepHeading:



Collaboration diagram for YQWizard::StepHeading:



#### Public Member Functions

- **StepHeading** (const QString &name="")
- virtual bool **isHeading** () const
- virtual bool **hasID** (const QString &id)

## Additional Inherited Members

### 3.17.1 Detailed Description

Helper class to represent a wizard step heading internally.

Definition at line 683 of file [YQWizard.h](#).

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

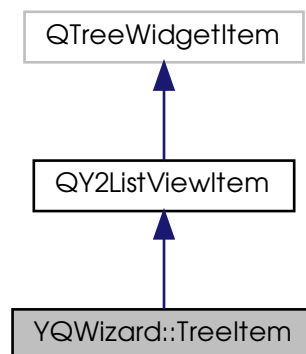
- `/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQWizard.h`

## 3.18 YQWizard::Treeltem Class Reference

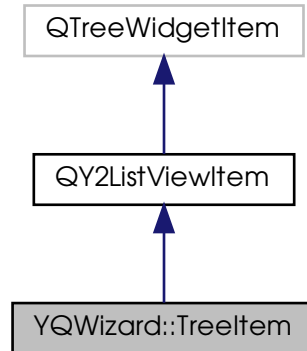
Helper class for wizard tree item.

```
#include <YQWizard.h>
```

Inheritance diagram for YQWizard::Treeltem:



Collaboration diagram for YQWizard::Treeltem:



## Public Member Functions

- **Treeltem** ([QY2ListView](#) \*parent, const QString &text, const QString &id)
- **Treeltem** ([YQWizard::Treeltem](#) \*parent, const QString &text, const QString &id)
- virtual QString **text** (int index) const
- QString **text** () const
- QString **id** () const

## Additional Inherited Members

### 3.18.1 Detailed Description

Helper class for wizard tree item.

Definition at line [703](#) of file [YQWizard.h](#).

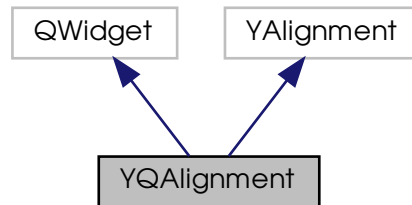
The documentation for this class was generated from the following file:

- /usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQWizard.h

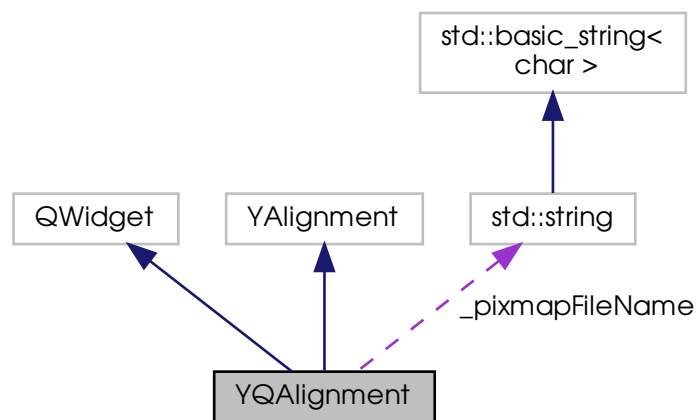


## 3.19 YQAlignment Class Reference

Inheritance diagram for YQAlignment:



Collaboration diagram for YQAlignment:



### Public Member Functions

- [YQAlignment](#) (YWidget \*parent, YAlignmentType horAlign, YAlignmentType vertAlign)  
*Constructor.*
- [YQAlignment](#) (YWidget \*yParent, QWidget \*qParent, YAlignmentType horAlign, YAlignmentType vertAlign)  
*Constructor for cases where the YWidget parent's widgetRep() isn't the QWidget parent, for example in compound widgets like [YQWizard](#) where it makes sense to pass a child of the [YQWizard](#) (the client area) as the QWidget parent.*
- virtual void [moveChild](#) (YWidget \*child, int newX, int newY)

*Move the child widget to a new position.*

- virtual void [setEnabled](#) (bool enabled)

*Enable or disable this widget and its child.*

- virtual void [setSize](#) (int newWidth, int newHeight)

*Set the new size of the widget.*

- virtual void [setBackgroundPixmap](#) (const std::string &pixmapFileName)

*Set the background pixmap.*

## Protected Member Functions

- virtual void **paintEvent** (QPaintEvent \*event)

## Protected Attributes

- std::string **\_pixmapFileName**

### 3.19.1 Detailed Description

Definition at line 35 of file [YQAlignment.h](#).

### 3.19.2 Constructor & Destructor Documentation

#### 3.19.2.1 YQAlignment()

```
YQAlignment::YQAlignment (
    YWidget * yParent,
    QWidget * qParent,
    YAlignmentType horAlign,
    YAlignmentType vertAlign )
```

Constructor for cases where the YWidget parent's widgetRep() isn't the QWidget parent, for example in compound widgets like [YQWizard](#) where it makes sense to pass a child of the [YQWizard](#) (the client area) as the QWidget parent.

Use this only if you know very well what you are doing.

Definition at line 45 of file [YQAlignment.cc](#).

### 3.19.3 Member Function Documentation

#### 3.19.3.1 moveChild()

```
void YQAlignment::moveChild (
    YWidget * child,
    int newX,
    int newY ) [virtual]
```

Move the child widget to a new position.

Implemented from YAlignment.

Definition at line 63 of file [YQAlignment.cc](#).

#### 3.19.3.2 setBackgroundPixmap()

```
void YQAlignment::setBackgroundPixmap (
    const std::string & pixmapFileName ) [virtual]
```

Set the background pixmap.

Reimplemented from YAlignment.

Definition at line 84 of file [YQAlignment.cc](#).

#### 3.19.3.3 setEnabled()

```
void YQAlignment::setEnabled (
    bool enabled ) [virtual]
```

Enable or disable this widget and its child.

Reimplemented from YWidget.

Definition at line 56 of file [YQAlignment.cc](#).

#### 3.19.3.4 setSize()

```
void YQAlignment::setSize (
    int newWidth,
    int newHeight ) [virtual]
```

Set the new size of the widget.

Reimplemented from YWidget.

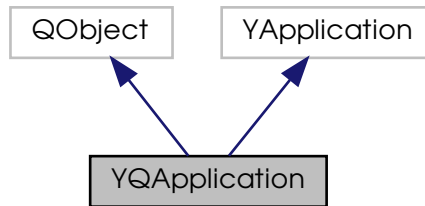
Definition at line 70 of file [YQAlignment.cc](#).

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

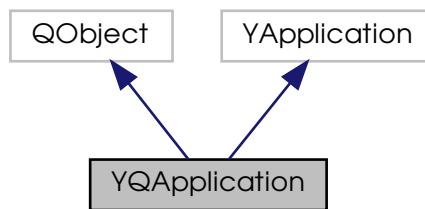
- /usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQAlignment.h
- /usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQAlignment.cc

### 3.20 YQApplication Class Reference

Inheritance diagram for YQApplication:



Collaboration diagram for YQApplication:



#### Public Member Functions

- virtual void [setLanguage](#) (const std::string &language, const std::string &encoding=std::string())  
Set language and encoding for the locale environment (\$LANG).
- void [loadPredefinedQtTranslations](#) ()  
Load translations for Qt's predefined dialogs like file selection box etc.
- void [setLayoutDirection](#) (const std::string &language)  
Set the layout direction (left-to-right or right-to-left) from 'language'.
- void [setLangFonts](#) (const std::string &language, const std::string &encoding=std::string())  
Set fonts according to the specified language and encoding.
- const QFont & [currentFont](#) ()  
Returns the application's default font.
- const QFont & [boldFont](#) ()

- Returns the application's default bold font.*

  - const QFont & **headingFont** ()

*Returns the application's heading font.*
- void **deleteFonts** ()

*Delete the fonts so they will be reloaded upon their next usage.*
- void **pickAutoFonts** ()

*Determine good fonts based on defaultsize geometry and set `_auto_normal_font_size` and `_auto_heading_font_size` accordingly.*
- bool **autoFonts** () const

*Returns 'true' if the UI automatically picks fonts, disregarding Qt standard settings.*
- void **setAutoFonts** (bool useAutoFonts)

*Set whether or not fonts should automatically be picked.*
- virtual std::string **glyph** (const std::string &glyphSymbolName)

*Return a std::string for a named glyph.*
- virtual std::string **askForExistingDirectory** (const std::string &startDir, const std::string &headline)

*Open a directory selection box and prompt the user for an existing directory.*
- virtual std::string **askForExistingFile** (const std::string &startWith, const std::string &filter, const std::string &headline)

*Open a file selection box and prompt the user for an existing file.*
- virtual std::string **askForSaveFileName** (const std::string &startWith, const std::string &filter, const std::string &headline)

*Open a file selection box and prompt the user for a file to save data to.*
- virtual bool **openContextMenu** (const YItemCollection &itemCollection)

*Open a context menu for a widget.*
- void **maybeLeftHandedUser** ()

*A mouse click with the wrong mouse button was detected - e.g., a right click on a push button.*
- virtual int **deviceUnits** (YUIDimension dim, float **layoutUnits**)

*Convert logical layout spacing units into device dependent units.*
- virtual float **layoutUnits** (YUIDimension dim, int **deviceUnits**)

*Convert device dependent units into logical layout spacing units.*
- virtual void **busyCursor** ()

*Change the (mouse) cursor to indicate busy status.*
- virtual void **normalCursor** ()

*Change the (mouse) cursor back from busy status to normal.*
- virtual void **makeScreenShot** (const std::string &fileName)

*Make a screen shot and save it to the specified file.*
- virtual void **beep** ()

*Beep.*
- virtual QPoint **contextMenuPos** ()

*Return position of the context menu (in gloabl coordinates)*
- virtual void **setContextMenuPos** (QPoint **contextMenuPos**)

*Sets the position of the context menu (in gloabl coordinates)*
- virtual int **displayWidth** ()
- virtual int **displayHeight** ()
- virtual int **displayDepth** ()
- virtual long **displayColors** ()
- virtual int **defaultWidth** ()
- virtual int **defaultHeight** ()

- virtual bool **isTextMode** ()
- virtual bool **hasImageSupport** ()
- virtual bool **hasIconSupport** ()
- virtual bool **hasAnimationSupport** ()
- virtual bool **hasFullUtf8Support** ()
- virtual bool **richTextSupportsTable** ()
- virtual bool **hasWizardDialogSupport** ()
- virtual bool **leftHandedMouse** ()
- virtual void **setApplicationTitle** (const std::string &title)  
*Set the application title.*
- virtual void **setApplicationIcon** (const std::string &icon)  
*Set the application icon.*

### Static Public Member Functions

- static QString **askForSaveFileName** (const QString &startWith, const QString &filter, const QString &headline)  
*Lower-level version that works with QStrings and does not change the mouse cursor.*
- static **YQPackageSelectorPluginStub** \* **packageSelectorPlugin** ()  
*Return the package selector plugin singleton or creates it (including loading the plugin lib) if it does not exist yet.*
- static **YQGraphPluginStub** \* **graphPlugin** ()  
*Return the package selector plugin singleton or creates it (including loading the plugin lib) if it does not exist yet.*

### Protected Member Functions

- **YQApplication** ()  
*Constructor.*
- virtual **~YQApplication** ()  
*Destructor.*
- QString **fontKey** (const QString &lang)  
*Constructs a key for the language specific font file: "font[lang]" for font[de\_DE] = "Sans Serif" font[zh] = "ChineseSpecial, something" font[ja\_JP] = "JapaneseSpecial, something" font = "Sans Serif".*

### Protected Attributes

- QFont \* **\_currentFont**
- QFont \* **\_headingFont**
- QFont \* **\_boldFont**
- QString **\_fontFamily**  
*Font family or list of font families to use ("Sans Serif" etc.)*
- QSettings \* **\_langFonts**  
*Language-specific font settings.*
- QTranslator \* **\_qtTranslations**  
*Translator for the predefined Qt dialogs.*
- bool **\_autoFonts**
- int **\_autoNormalFontSize**
- int **\_autoHeadingFontSize**
- bool **\_leftHandedMouse**
- bool **\_askedForLeftHandedMouse**
- QPoint **\_contextMenuPos**
- QMenu \* **\_contextMenu**

## Friends

- class **YQUI**

### 3.20.1 Detailed Description

Definition at line 43 of file [YQApplication.h](#).

### 3.20.2 Constructor & Destructor Documentation

#### 3.20.2.1 YQApplication()

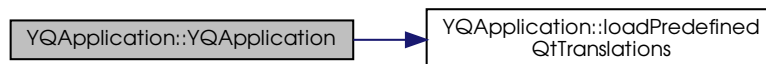
```
YQApplication::YQApplication ( ) [protected]
```

Constructor.

Use `YUI::app()` to get the singleton for this class.

Definition at line 62 of file [YQApplication.cc](#).

Here is the call graph for this function:



### 3.20.3 Member Function Documentation

### 3.20.3.1 askForExistingDirectory()

```
string YQApplication::askForExistingDirectory (
    const std::string & startDir,
    const std::string & headline ) [virtual]
```

Open a directory selection box and prompt the user for an existing directory.

'startDir' is the initial directory that is displayed.

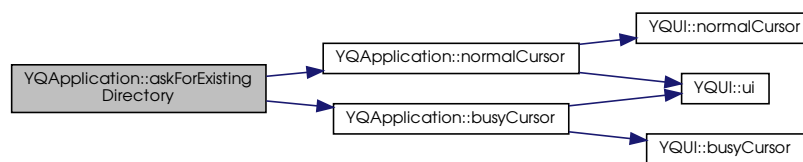
'headline' is an explanatory text for the directory selection box. Graphical UIs may omit that if no window manager is running.

Returns the selected directory name or an empty std::string if the user canceled the operation.

Implemented from YApplication.

Definition at line 478 of file [YQApplication.cc](#).

Here is the call graph for this function:



### 3.20.3.2 askForExistingFile()

```
string YQApplication::askForExistingFile (
    const std::string & startWith,
    const std::string & filter,
    const std::string & headline ) [virtual]
```

Open a file selection box and prompt the user for an existing file.

'startWith' is the initial directory or file.

'filter' is one or more blank-separated file patterns, e.g. "\*.png \*.jpg"

'headline' is an explanatory text for the file selection box. Graphical UIs may omit that if no window manager is running.

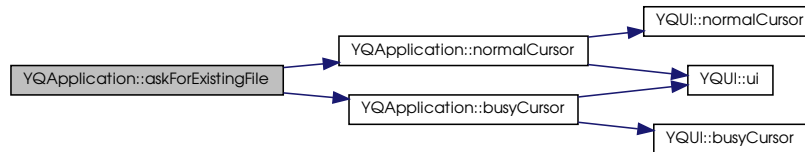
Returns the selected file name or an empty std::string if the user canceled the operation.

Implemented from YApplication.



Definition at line 495 of file [YQApplication.cc](#).

Here is the call graph for this function:



### 3.20.3.3 askForSaveFileName()

```

string YQApplication::askForSaveFileName (
    const std::string & startWith,
    const std::string & filter,
    const std::string & headline ) [virtual]
  
```

Open a file selection box and prompt the user for a file to save data to.

Automatically asks for confirmation if the user selects an existing file.

'startWith' is the initial directory or file.

'filter' is one or more blank-separated file patterns, e.g. "\*.png \*.jpg"

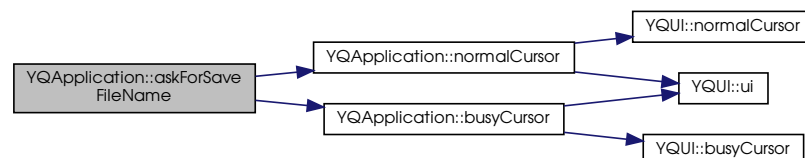
'headline' is an explanatory text for the file selection box. Graphical UIs may omit that if no window manager is running.

Returns the selected file name or an empty `std::string` if the user canceled the operation.

Implemented from `YApplication`.

Definition at line 521 of file [YQApplication.cc](#).

Here is the call graph for this function:



### 3.20.3.4 autoFonts()

```
bool YQApplication::autoFonts ( ) const [inline]
```

Returns 'true' if the UI automatically picks fonts, disregarding Qt standard settings.

This makes sense during system installation system where the display DPI cannot reliably be retrieved and thus Qt uses random font sizes based on that random DPI.

Definition at line 138 of file [YQApplication.h](#).

### 3.20.3.5 beep()

```
void YQApplication::beep ( ) [virtual]
```

Beep.

Reimplemented from YApplication.

Definition at line 699 of file [YQApplication.cc](#).

### 3.20.3.6 busyCursor()

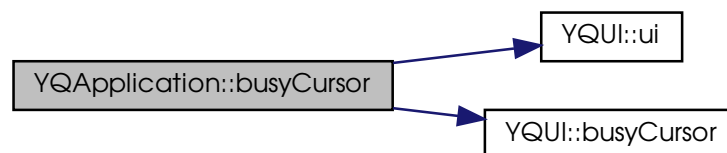
```
void YQApplication::busyCursor ( ) [virtual]
```

Change the (mouse) cursor to indicate busy status.

Reimplemented from YApplication.

Definition at line 705 of file [YQApplication.cc](#).

Here is the call graph for this function:



### 3.20.3.7 currentFont()

```
const QFont & YQApplication::currentFont ( )
```

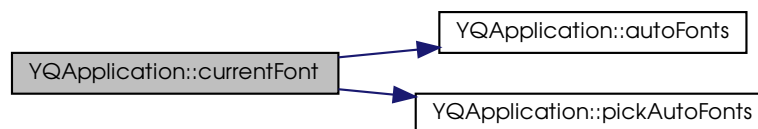
Returns the application's default font.

Brute force approach to make sure we'll really get a complete Unicode font: Explicitly load the one font that we made sure to contain all required characters, including Latin1, Latin2, Japanese, Korean, and the characters used for glyphs.

There are many fonts that claim to be Unicode, but most of them contain just a sorry excuse for a complete Unicode character set. Qt can't know how complete a font is, so it chooses one that might be better in otherf aspects, but lacks necessary characters.

Definition at line 294 of file [YQApplication.cc](#).

Here is the call graph for this function:



### 3.20.3.8 deviceUnits()

```
int YQApplication::deviceUnits (
    YUIDimension dim,
    float layoutUnits ) [virtual]
```

Convert logical layout spacing units into device dependent units.

A default size dialog is assumed to be 80x25 layout spacing units.

Reimplemented from `YApplication`.

Definition at line 679 of file [YQApplication.cc](#).

Here is the call graph for this function:



### 3.20.3.9 glyph()

```
string YQApplication::glyph (
    const std::string & glyphSymbolName ) [virtual]
```

Return a std::string for a named glyph.

Using this is discouraged in new applications.

Reimplemented from YApplication.

Definition at line 457 of file [YQApplication.cc](#).

### 3.20.3.10 headingFont()

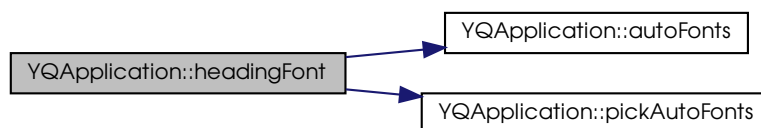
```
const QFont & YQApplication::headingFont ( )
```

Returns the application's heading font.

Brute force load the heading font - see [currentFont\(\)](#) above for more.

Definition at line 349 of file [YQApplication.cc](#).

Here is the call graph for this function:



## 3.20.3.11 layoutUnits()

```
float YQApplication::layoutUnits (
    YUIDimension dim,
    int deviceUnits ) [virtual]
```

Convert device dependent units into logical layout spacing units.

A default size dialog is assumed to be 80x25 layout spacing units.

Reimplemented from YApplication.

Definition at line 688 of file [YQApplication.cc](#).

Here is the call graph for this function:



## 3.20.3.12 makeScreenShot()

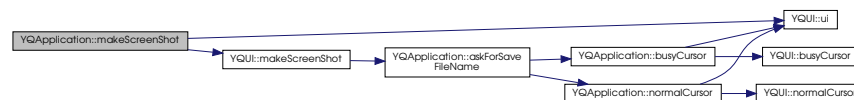
```
void YQApplication::makeScreenShot (
    const std::string & fileName ) [virtual]
```

Make a screen shot and save it to the specified file.

Reimplemented from YApplication.

Definition at line 717 of file [YQApplication.cc](#).

Here is the call graph for this function:



### 3.20.3.13 maybeLeftHandedUser()

```
void YQApplication::maybeLeftHandedUser ( )
```

A mouse click with the wrong mouse button was detected - e.g., a right click on a push button.

The user might be left-handed, but his mouse might not (yet) be configured for left-handed use - e.g., during installation. Ask him if he would like his mouse temporarily configured as a left-handed mouse.

This status can be queried with `YQApplication::leftHandedMouse()`.

Definition at line 629 of file [YQApplication.cc](#).

### 3.20.3.14 normalCursor()

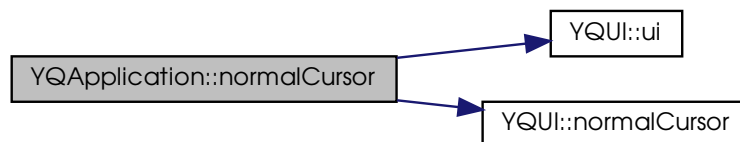
```
void YQApplication::normalCursor ( ) [virtual]
```

Change the (mouse) cursor back from busy status to normal.

Reimplemented from `YApplication`.

Definition at line 711 of file [YQApplication.cc](#).

Here is the call graph for this function:



### 3.20.3.15 openContextMenu()

```
bool YQApplication::openContextMenu (
    const YItemCollection & itemCollection ) [virtual]
```

Open a context menu for a widget.

'itemCollection' describes the menu structure

Returns true on success (otherwise false).

Implemented from `YApplication`.

Definition at line 537 of file [YQApplication.cc](#).

### 3.20.3.16 pickAutoFonts()

```
void YQApplication::pickAutoFonts ( )
```

Determine good fonts based on defaultsize geometry and set `_auto_normal_font_size` and `_auto_heading_font_size` accordingly.

Caches the values, so it's safe to call this repeatedly.

Definition at line 400 of file [YQApplication.cc](#).

### 3.20.3.17 setApplicationIcon()

```
void YQApplication::setApplicationIcon (
    const std::string & icon ) [virtual]
```

Set the application icon.

Reimplemented from YApplication.

Definition at line 772 of file [YQApplication.cc](#).

### 3.20.3.18 setApplicationTitle()

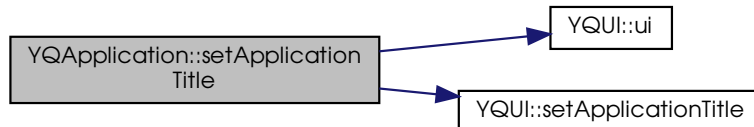
```
void YQApplication::setApplicationTitle (
    const std::string & title ) [virtual]
```

Set the application title.

Reimplemented from YApplication.

Definition at line 764 of file [YQApplication.cc](#).

Here is the call graph for this function:



### 3.20.3.19 setLangFonts()

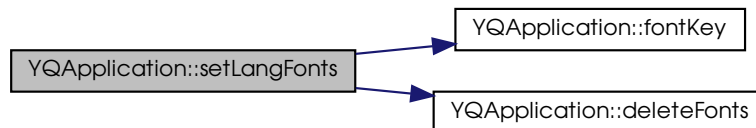
```
void YQApplication::setLangFonts (
    const std::string & language,
    const std::string & encoding = std::string() )
```

Set fonts according to the specified language and encoding.

This is most important for some Asian languages that have overlaps in the Unicode table, like Japanese vs. Chinese.

Definition at line [204](#) of file [YQApplication.cc](#).

Here is the call graph for this function:



### 3.20.3.20 setLanguage()

```
void YQApplication::setLanguage (
    const std::string & language,
    const std::string & encoding = std::string() ) [virtual]
```

Set language and encoding for the locale environment (\$LANG).

Load UI-internal translations (e.g. for predefined dialogs like file selection) and fonts.

'language' is the ISO short code ("de\_DE", "en\_US", ...).

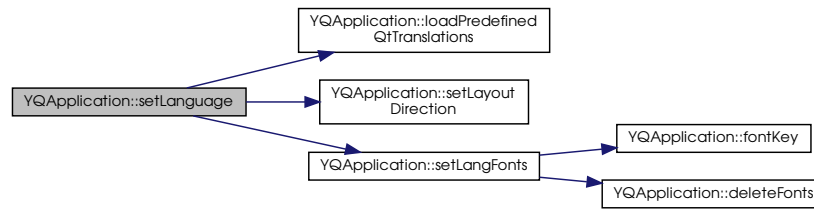
'encoding' an (optional) encoding ("utf8", ...) that will be appended if present.

Reimplemented from YApplication.

Definition at line [100](#) of file [YQApplication.cc](#).



Here is the call graph for this function:

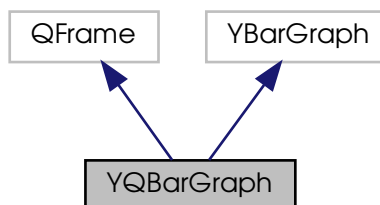


The documentation for this class was generated from the following files:

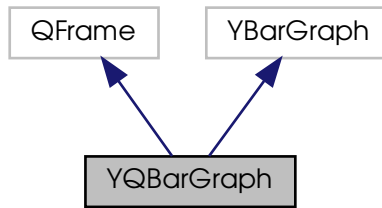
- `/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQApplication.h`
- `/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQApplication.cc`

## 3.21 YQBarGraph Class Reference

Inheritance diagram for YQBarGraph:



Collaboration diagram for YQBarGraph:



### Public Member Functions

- [YQBarGraph](#) (YWidget \*parent)  
*Constructor.*
- virtual [~YQBarGraph](#) ()  
*Destructor.*
- virtual void [doUpdate](#) ()  
*Perform a visual update on the screen.*
- virtual void [setEnabled](#) (bool enabled)  
*Set enabled/disabled state.*
- virtual int [preferredWidth](#) ()  
*Preferred width of the widget.*
- virtual int [preferredHeight](#) ()  
*Preferred height of the widget.*
- virtual void [setSize](#) (int newWidth, int newHeight)  
*Set the new size of the widget.*

### Protected Member Functions

- virtual bool **event** (QEvent \*event)
- virtual void [paintEvent](#) (QPaintEvent \*painter)  
*Draw the contents.*
- YColor [defaultSegmentColor](#) (unsigned index)  
*Return one from a set of default segment background colors.*
- YColor [defaultTextColor](#) (unsigned index)  
*Return one from a set of default text colors.*

### Properties

- QString **BackgroundColors**
- QString **ForegroundColors**

### 3.21.1 Detailed Description

Definition at line 39 of file [YQBarGraph.h](#).

### 3.21.2 Member Function Documentation

#### 3.21.2.1 defaultTextColor()

```
YColor YQBarGraph::defaultTextColor (
    unsigned index ) [protected]
```

Return one from a set of default text colors.

This text color is guaranteed to contrast with the defaultSegmentColor with the same index.

Definition at line 221 of file [YQBarGraph.cc](#).

#### 3.21.2.2 doUpdate()

```
void YQBarGraph::doUpdate ( ) [virtual]
```

Perform a visual update on the screen.

Implemented from YBarGraph.

Definition at line 68 of file [YQBarGraph.cc](#).

#### 3.21.2.3 paintEvent()

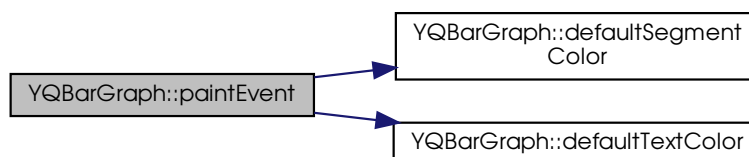
```
void YQBarGraph::paintEvent (
    QPaintEvent * painter ) [protected], [virtual]
```

Draw the contents.

Reimplemented from QFrame.

Definition at line 93 of file [YQBarGraph.cc](#).

Here is the call graph for this function:



#### 3.21.2.4 preferredHeight()

```
int YQBarGraph::preferredHeight ( ) [virtual]
```

Preferred height of the widget.

Reimplemented from YWidget.

Definition at line 290 of file [YQBarGraph.cc](#).

#### 3.21.2.5 preferredWidth()

```
int YQBarGraph::preferredWidth ( ) [virtual]
```

Preferred width of the widget.

Reimplemented from YWidget.

Definition at line 264 of file [YQBarGraph.cc](#).

#### 3.21.2.6 setEnabled()

```
void YQBarGraph::setEnabled (
    bool enabled ) [virtual]
```

Set enabled/disabled state.

Reimplemented from YWidget.

Definition at line 256 of file [YQBarGraph.cc](#).

#### 3.21.2.7 setSize()

```
void YQBarGraph::setSize (
    int newWidth,
    int newHeight ) [virtual]
```

Set the new size of the widget.

Reimplemented from YWidget.

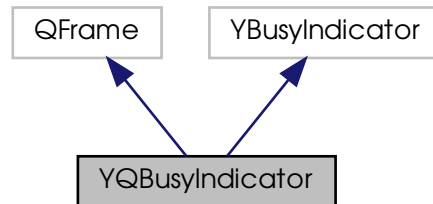
Definition at line 316 of file [YQBarGraph.cc](#).

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

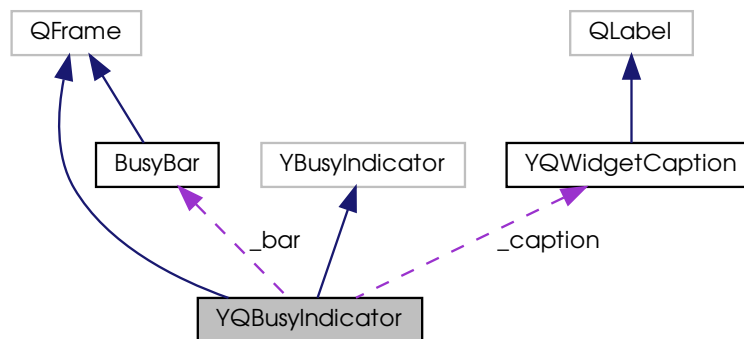
- [/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQBarGraph.h](#)
- [/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQBarGraph.cc](#)

## 3.22 YQBusyIndicator Class Reference

Inheritance diagram for YQBusyIndicator:



Collaboration diagram for YQBusyIndicator:



### Public Member Functions

- [YQBusyIndicator](#) (YWidget \*parent, const std::string &label, int timeout=1000)  
*Constructor.*
- virtual [~YQBusyIndicator](#) ()  
*Destructor.*
- virtual void [setLabel](#) (const std::string &label)  
*Set the label (the caption above the progress bar).*
- virtual void [setTimeout](#) (int newTimeout)  
*Set the timeout is ms after that the widget shows 'stalled' when no new tick is received.*

- `int timeout () const`  
*Return the timeout is ms after that the widget shows 'stalled' when no new tick is received.*
- `virtual void setAlive (bool newAlive)`  
*Send a keep alive message.*
- `virtual void setEnabled (bool enabled)`  
*Set enabled/disabled state.*
- `virtual int preferredWidth ()`  
*Preferred width of the widget.*
- `virtual int preferredHeight ()`  
*Preferred height of the widget.*
- `virtual void setSize (int newWidth, int newHeight)`  
*Set the new size of the widget.*
- `virtual bool setKeyboardFocus ()`  
*Accept the keyboard focus.*

## Protected Attributes

- `YQWidgetCaption * _caption`
- `BusyBar * _bar`
- `QTimer * _timer`
- `int _timeout`

### 3.22.1 Detailed Description

Definition at line 39 of file [YQBusyIndicator.h](#).

### 3.22.2 Member Function Documentation

#### 3.22.2.1 preferredHeight()

```
int YQBusyIndicator::preferredHeight ( ) [virtual]
```

Preferred height of the widget.

Reimplemented from YWidget.

Definition at line 210 of file [YQBusyIndicator.cc](#).

### 3.22.2.2 preferredWidth()

```
int YQBusyIndicator::preferredWidth ( ) [virtual]
```

Preferred width of the widget.

Reimplemented from YWidget.

Definition at line 201 of file [YQBusyIndicator.cc](#).

### 3.22.2.3 setAlive()

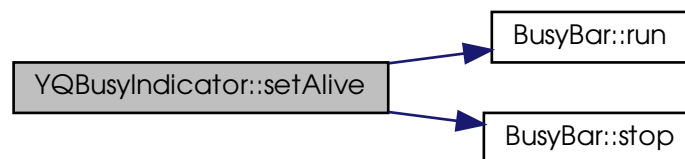
```
void YQBusyIndicator::setAlive (
    bool newAlive ) [virtual]
```

Send a keep alive message.

Reimplemented from YBusyIndicator.

Definition at line 163 of file [YQBusyIndicator.cc](#).

Here is the call graph for this function:



### 3.22.2.4 setEnabled()

```
void YQBusyIndicator::setEnabled (
    bool enabled ) [virtual]
```

Set enabled/disabled state.

Reimplemented from YWidget.

Definition at line 193 of file [YQBusyIndicator.cc](#).

### 3.22.2.5 setLabel()

```
void YQBusyIndicator::setLabel (
    const std::string & label ) [virtual]
```

Set the label (the caption above the progress bar).

Reimplemented from YBusyIndicator.

Definition at line 156 of file [YQBusyIndicator.cc](#).

Here is the call graph for this function:



### 3.22.2.6 setSize()

```
void YQBusyIndicator::setSize (
    int newWidth,
    int newHeight ) [virtual]
```

Set the new size of the widget.

Reimplemented from YWidget.

Definition at line 216 of file [YQBusyIndicator.cc](#).

### 3.22.2.7 setTimeout()

```
void YQBusyIndicator::setTimeout (
    int newTimeout ) [virtual]
```

Set the timeout is ms after that the widget shows 'stalled' when no new tick is received.

Reimplemented from YBusyIndicator.

Definition at line 186 of file [YQBusyIndicator.cc](#).



### 3.22.2.8 timeout()

```
int YQBusyIndicator::timeout ( ) const [inline]
```

Return the timeout is ms after that the widget shows 'stalled' when no new tick is received.

Reimplemented from YBusyIndicator.

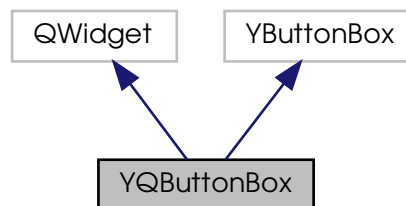
Definition at line 76 of file [YQBusyIndicator.h](#).

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

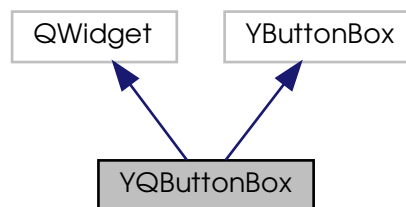
- /usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQBusyIndicator.h
- /usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQBusyIndicator.cc

## 3.23 YQButtonBox Class Reference

Inheritance diagram for YQButtonBox:



Collaboration diagram for YQButtonBox:



## Public Member Functions

- [YQButtonBox](#) (YWidget \*parent)  
*Constructor.*
- virtual [~YQButtonBox](#) ()  
*Destructor.*
- virtual void [setEnabled](#) (bool enabled)  
*Set enabled/disabled state.*
- virtual void [setSize](#) (int newWidth, int newHeight)  
*Set the new size of the widget.*
- virtual void [moveChild](#) (YWidget \*child, int newX, int newY)  
*Move a child widget to a new position.*

### 3.23.1 Detailed Description

Definition at line 33 of file [YQButtonBox.h](#).

### 3.23.2 Member Function Documentation

#### 3.23.2.1 moveChild()

```
void YQButtonBox::moveChild (  
    YWidget * child,  
    int newX,  
    int newY ) [virtual]
```

Move a child widget to a new position.

Reimplemented from YButtonBox.

Definition at line 59 of file [YQButtonBox.cc](#).

#### 3.23.2.2 setEnabled()

```
void YQButtonBox::setEnabled (  
    bool enabled ) [virtual]
```

Set enabled/disabled state.

Reimplemented from YWidget.

Definition at line 45 of file [YQButtonBox.cc](#).

### 3.23.2.3 setSize()

```
void YQButtonBox::setSize (
    int newWidth,
    int newHeight ) [virtual]
```

Set the new size of the widget.

Reimplemented from YWidget.

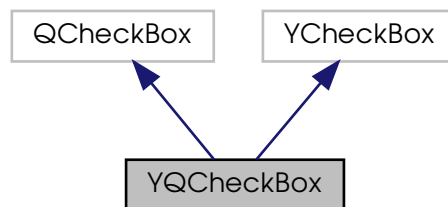
Definition at line 52 of file [YQButtonBox.cc](#).

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

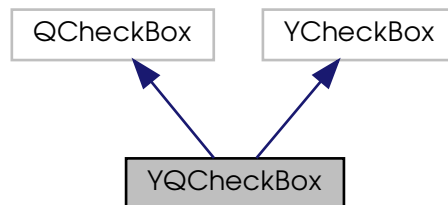
- /usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQButtonBox.h
- /usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQButtonBox.cc

## 3.24 YQCheckBox Class Reference

Inheritance diagram for YQCheckBox:



Collaboration diagram for YQCheckBox:



## Public Member Functions

- [YQCheckBox](#) (YWidget \*parent, const std::string &label, bool checked)  
*Constructor.*
- virtual [~YQCheckBox](#) ()  
*Destructor.*
- virtual YCheckBoxState [value](#) ()  
*Get the current value:*
- virtual void [setValue](#) (YCheckBoxState state)  
*Set the CheckBox value (on/off/don't care).*
- virtual void [setLabel](#) (const std::string &label)  
*Change the label (the text) on the RadioButton.*
- virtual void [setUseBoldFont](#) (bool bold=true)  
*Use a bold font.*
- virtual void [setEnabled](#) (bool enabled)  
*Set enabled / disabled state.*
- virtual int [preferredWidth](#) ()  
*Preferred width of the widget.*
- virtual int [preferredHeight](#) ()  
*Preferred height of the widget.*
- virtual void [setSize](#) (int newWidth, int newHeight)  
*Set the new size of the widget.*
- virtual bool [setKeyboardFocus](#) ()  
*Accept the keyboard focus.*

### 3.24.1 Detailed Description

Definition at line 32 of file [YQCheckBox.h](#).

### 3.24.2 Member Function Documentation

#### 3.24.2.1 preferredHeight()

```
int YQCheckBox::preferredHeight ( ) [virtual]
```

Preferred height of the widget.

Reimplemented from YWidget.

Definition at line 129 of file [YQCheckBox.cc](#).

#### 3.24.2.2 preferredWidth()

```
int YQCheckBox::preferredWidth ( ) [virtual]
```

Preferred width of the widget.

Reimplemented from YWidget.

Definition at line 123 of file [YQCheckBox.cc](#).

#### 3.24.2.3 setEnabled()

```
void YQCheckBox::setEnabled (
    bool enabled ) [virtual]
```

Set enabled / disabled state.

Reimplemented from YWidget.

Definition at line 116 of file [YQCheckBox.cc](#).

#### 3.24.2.4 setKeyboardFocus()

```
bool YQCheckBox::setKeyboardFocus ( ) [virtual]
```

Accept the keyboard focus.

Reimplemented from YWidget.

Definition at line 141 of file [YQCheckBox.cc](#).

#### 3.24.2.5 setLabel()

```
void YQCheckBox::setLabel (
    const std::string & label ) [virtual]
```

Change the label (the text) on the RadioButton.

Reimplemented from YRadioButton.

Definition at line 99 of file [YQCheckBox.cc](#).

### 3.24.2.6 setSize()

```
void YQCheckBox::setSize (
    int newWidth,
    int newHeight ) [virtual]
```

Set the new size of the widget.

Reimplemented from YWidget.

Definition at line 135 of file [YQCheckBox.cc](#).

### 3.24.2.7 setUseBoldFont()

```
void YQCheckBox::setUseBoldFont (
    bool bold = true ) [virtual]
```

Use a bold font.

Reimplemented from YRadioButton.

Definition at line 106 of file [YQCheckBox.cc](#).

Here is the call graph for this function:



### 3.24.2.8 setValue()

```
void YQCheckBox::setValue (
    YCheckBoxState state ) [virtual]
```

Set the CheckBox value (on/off/don't care).

Reimplemented from YCheckBox.

Definition at line 77 of file [YQCheckBox.cc](#).

### 3.24.2.9 value()

```
YCheckBoxState YQCheckBox::value ( ) [virtual]
```

Get the current value:

YCheckBox\_on CheckBox is checked YCheckBox\_off CheckBox is unchecked

YCheckBox\_dont\_care tri-state: CheckBox is greyed out, neither checked nor unchecked

Reimplemented from YCheckBox.

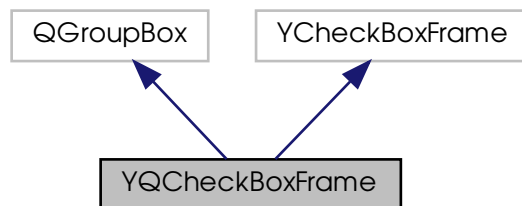
Definition at line 63 of file [YQCheckBox.cc](#).

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

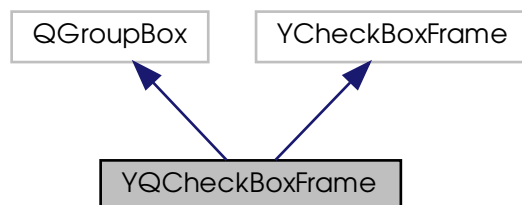
- /usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQCheckBox.h
- /usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQCheckBox.cc

## 3.25 YQCheckBoxFrame Class Reference

Inheritance diagram for YQCheckBoxFrame:



Collaboration diagram for YQCheckBoxFrame:



## Public Member Functions

- [YQCheckBoxFrame](#) (YWidget \*parent, const std::string &label, bool checked)  
*Constructor.*
- virtual void [setLabel](#) (const std::string &label)  
*Change the label text on the CheckBoxFrame.*
- virtual void [setValue](#) (bool isChecked)  
*Check or uncheck the CheckBoxFrame's check box.*
- virtual bool [value](#) ()  
*Get the status of the CheckBoxFrame's check box.*
- virtual void [setEnabled](#) (bool enabled)  
*Set enabled / disabled state.*
- virtual int [preferredWidth](#) ()  
*Preferred width of the widget.*
- virtual int [preferredHeight](#) ()  
*Preferred height of the widget.*
- virtual void [setSize](#) (int newWidth, int newHeight)  
*Set the new size of the widget.*
- virtual bool [setKeyboardFocus](#) ()  
*Accept the keyboard focus.*

## Protected Member Functions

- virtual void [childEvent](#) (QChildEvent \*)  
*Reimplemented from QGroupBox to prevent QGroupBox from disabling children according to the check box status as the children are inserted.*
- virtual bool [event](#) (QEvent \*e)

### 3.25.1 Detailed Description

Definition at line 36 of file [YQCheckBoxFrame.h](#).

### 3.25.2 Member Function Documentation

#### 3.25.2.1 preferredHeight()

```
int YQCheckBoxFrame::preferredHeight ( ) [virtual]
```

Preferred height of the widget.

Reimplemented from YWidget.

Definition at line 177 of file [YQCheckBoxFrame.cc](#).



### 3.25.2.2 preferredWidth()

```
int YQCheckBoxFrame::preferredWidth ( ) [virtual]
```

Preferred width of the widget.

Reimplemented from YWidget.

Definition at line 167 of file [YQCheckBoxFrame.cc](#).

### 3.25.2.3 setEnabled()

```
void YQCheckBoxFrame::setEnabled (
    bool enabled ) [virtual]
```

Set enabled / disabled state.

Reimplemented from YWidget.

Definition at line 79 of file [YQCheckBoxFrame.cc](#).

Here is the call graph for this function:



### 3.25.2.4 setKeyboardFocus()

```
bool YQCheckBoxFrame::setKeyboardFocus ( ) [virtual]
```

Accept the keyboard focus.

Reimplemented from YWidget.

Definition at line 187 of file [YQCheckBoxFrame.cc](#).

### 3.25.2.5 setLabel()

```
void YQCheckBoxFrame::setLabel (
    const std::string & label ) [virtual]
```

Change the label text on the CheckBoxFrame.

Reimplemented from YCheckBoxFrame.

Definition at line 59 of file [YQCheckBoxFrame.cc](#).

### 3.25.2.6 setSize()

```
void YQCheckBoxFrame::setSize (
    int newWidth,
    int newHeight ) [virtual]
```

Set the new size of the widget.

Reimplemented from YWidget.

Definition at line 148 of file [YQCheckBoxFrame.cc](#).

### 3.25.2.7 setValue()

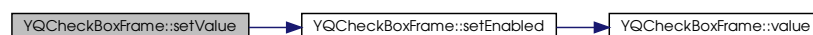
```
void YQCheckBoxFrame::setValue (
    bool isChecked ) [virtual]
```

Check or uncheck the CheckBoxFrame's check box.

Reimplemented from YCheckBoxFrame.

Definition at line 72 of file [YQCheckBoxFrame.cc](#).

Here is the call graph for this function:



### 3.25.2.8 value()

```
bool YQCheckBoxFrame::value ( ) [virtual]
```

Get the status of the CheckBoxFrame's check box.

Reimplemented from YCheckBoxFrame.

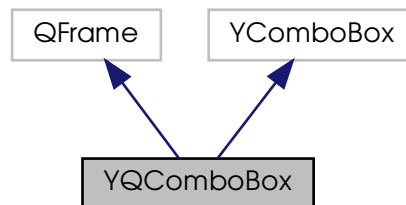
Definition at line 66 of file [YQCheckBoxFrame.cc](#).

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

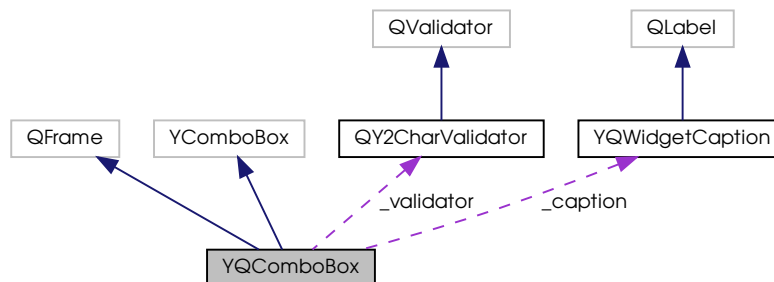
- /usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQCheckBoxFrame.h
- /usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQCheckBoxFrame.cc

## 3.26 YQComboBox Class Reference

Inheritance diagram for YQComboBox:



Collaboration diagram for YQComboBox:



## Public Member Functions

- [YQComboBox](#) (YWidget \*parent, const std::string &label, bool editable)  
*Constructor.*
- [~YQComboBox](#) ()  
*Destructor.*
- virtual void [addItem](#) (YItem \*item)  
*Add one item.*
- virtual void [deleteAllItems](#) ()  
*Delete all items.*
- virtual void [setLabel](#) (const std::string &label)  
*Change the label text.*
- virtual void [setValidChars](#) (const std::string &validChars)  
*Change the valid input characters.*
- virtual void [setInputMaxLength](#) (int numberOfChars)  
*Specify the amount of characters which can be inserted.*
- bool [isValidText](#) (const QString &txt) const  
*Returns 'true' if the given text is valid according to the current setting of ValidChars.*
- virtual void [setEnabled](#) (bool enabled)  
*Set enabled / disabled state.*
- virtual int [preferredWidth](#) ()  
*Preferred width of the widget.*
- virtual int [preferredHeight](#) ()  
*Preferred height of the widget.*
- virtual void [setSize](#) (int newWidth, int newHeight)  
*Set the new size of the widget.*
- virtual bool [setKeyboardFocus](#) ()  
*Accept the keyboard focus.*

## Protected Slots

- void [slotSelected](#) (int i)  
*Tells the ui that an item has been selected.*
- void [textChanged](#) (QString)  
*Tells the ui that the user has edited the text ( if the 'editable' option is set ).*

## Protected Member Functions

- virtual std::string [text](#) ()  
*Return this ComboBox's current value as text.*
- virtual void [setText](#) (const std::string &newText)  
*Set this ComboBox's current value as text.*

## Protected Attributes

- [YQWidgetCaption](#) \* `_caption`
- [QComboBox](#) \* `_qt_comboBox`
- [QY2CharValidator](#) \* `_validator`

### 3.26.1 Detailed Description

Definition at line 37 of file [YQComboBox.h](#).

### 3.26.2 Member Function Documentation

#### 3.26.2.1 addItem()

```
void YQComboBox::addItem (
    YItem * item ) [virtual]
```

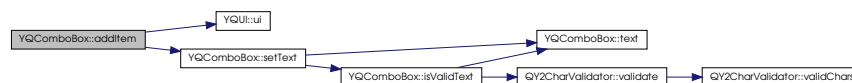
Add one item.

This widget assumes ownership of the item object and will delete it in its destructor.

Reimplemented from [YComboBox](#).

Definition at line 117 of file [YQComboBox.cc](#).

Here is the call graph for this function:



#### 3.26.2.2 deleteAllItems()

```
void YQComboBox::deleteAllItems ( ) [virtual]
```

Delete all items.

Reimplemented from [YSelectionWidget](#).

Definition at line 140 of file [YQComboBox.cc](#).

### 3.26.2.3 preferredHeight()

```
int YQComboBox::preferredHeight ( ) [virtual]
```

Preferred height of the widget.

Reimplemented from YWidget.

Definition at line [235](#) of file [YQComboBox.cc](#).

### 3.26.2.4 preferredWidth()

```
int YQComboBox::preferredWidth ( ) [virtual]
```

Preferred width of the widget.

Reimplemented from YWidget.

Definition at line [229](#) of file [YQComboBox.cc](#).

### 3.26.2.5 setEnabled()

```
void YQComboBox::setEnabled (
    bool enabled ) [virtual]
```

Set enabled / disabled state.

Reimplemented from YWidget.

Definition at line [247](#) of file [YQComboBox.cc](#).

### 3.26.2.6 setInputMaxLength()

```
void YQComboBox::setInputMaxLength (
    int numberOfChars ) [virtual]
```

Specify the amount of characters which can be inserted.

Reimplemented from YComboBox.

Definition at line [222](#) of file [YQComboBox.cc](#).

## 3.26.2.7 setLabel()

```
void YQComboBox::setLabel (
    const std::string & label ) [virtual]
```

Change the label text.

Reimplemented from YSelectionWidget.

Definition at line 149 of file [YQComboBox.cc](#).

Here is the call graph for this function:



## 3.26.2.8 setSize()

```
void YQComboBox::setSize (
    int newWidth,
    int newHeight ) [virtual]
```

Set the new size of the widget.

Reimplemented from YWidget.

Definition at line 241 of file [YQComboBox.cc](#).

## 3.26.2.9 setText()

```
void YQComboBox::setText (
    const std::string & newText ) [protected], [virtual]
```

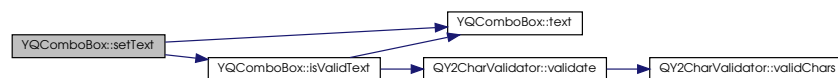
Set this ComboBox's current value as text.

Called internally whenever the content is to change programmatically. Don't call `setValue()` or `selectItem()` from here.

Implemented from YComboBox.

Definition at line 95 of file [YQComboBox.cc](#).

Here is the call graph for this function:



### 3.26.2.10 setValidChars()

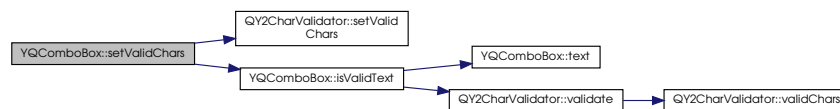
```
void YQComboBox::setValidChars (
    const std::string & validChars ) [virtual]
```

Change the valid input characters.

Reimplemented from YComboBox.

Definition at line 156 of file [YQComboBox.cc](#).

Here is the call graph for this function:



### 3.26.2.11 slotSelected

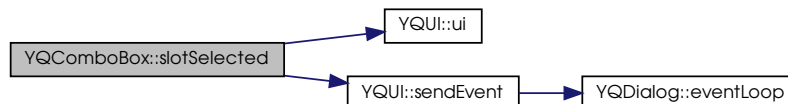
```
void YQComboBox::slotSelected (
    int i ) [protected], [slot]
```

Tells the ui that an item has been selected.

This is only interesting if the ``notify` option is set.

Definition at line 201 of file [YQComboBox.cc](#).

Here is the call graph for this function:





## 3.26.2.12 text()

```
string YQComboBox::text ( ) [protected], [virtual]
```

Return this ComboBox's current value as text.

Called internally from value(), selectedItem() and related.

Implemented from YComboBox.

Definition at line 89 of file [YQComboBox.cc](#).

## 3.26.2.13 textChanged

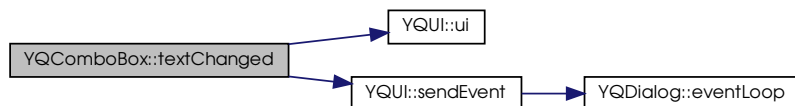
```
void YQComboBox::textChanged (
    QString ) [protected], [slot]
```

Tells the ui that the user has edited the text ( if the 'editable' option is set ).

This is only interesting if the `notify` option is set.

Definition at line 215 of file [YQComboBox.cc](#).

Here is the call graph for this function:

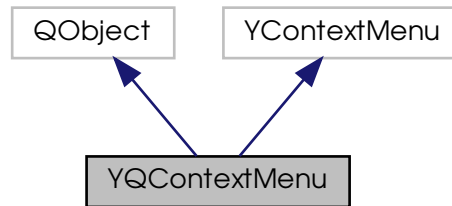


The documentation for this class was generated from the following files:

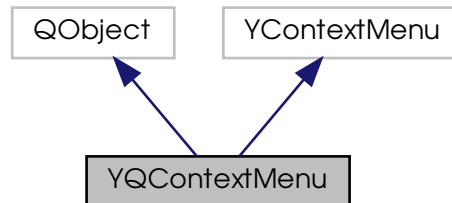
- `/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQComboBox.h`
- `/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQComboBox.cc`

### 3.27 YQContextMenu Class Reference

Inheritance diagram for YQContextMenu:



Collaboration diagram for YQContextMenu:



#### Public Member Functions

- [YQContextMenu](#) ()  
*Constructor.*
- **YQContextMenu** (QWidget \*parent, const QPoint position)
- virtual [~YQContextMenu](#) ()  
*Destructor.*
- virtual void [rebuildMenuTree](#) ()  
*Change the label on the button.*
- virtual int [preferredWidth](#) ()  
*Preferred width of the widget.*
- virtual int [preferredHeight](#) ()  
*Preferred height of the widget.*
- virtual void [setSize](#) (int newWidth, int newHeight)  
*Set the new size of the widget.*

## Protected Slots

- void [menuEntryActivated](#) (QAction \*menuItem)  
*Triggered when any menu item is activated.*
- void [returnNow](#) ()  
*Triggered via [menuEntryActivated\(\)](#) by zero timer to get back in sync with the Qt event loop.*
- void [slotMenuHidden](#) ()  
*Triggered when the context menu is hidden.*
- void [slotReturnMenuHidden](#) ()  
*Triggered via [slotMenuHidden\(\)](#) by zero timer to get back in sync with the Qt event loop.*

## Protected Member Functions

- void [rebuildMenuTree](#) (QMenu \*menu, YItemIterator begin, YItemIterator end)  
*Recursively insert menu items into 'menu' from iterator 'begin' to iterator 'end'.*

## Protected Attributes

- YMenuItem \* **\_selectedItem**
- QPushButton \* **\_qt\_button**
- QHash< QAction \*, int > **\_serials**

### 3.27.1 Detailed Description

Definition at line 37 of file [YQContextMenu.h](#).

### 3.27.2 Member Function Documentation

#### 3.27.2.1 preferredHeight()

```
int YQContextMenu::preferredHeight ( ) [virtual]
```

Preferred height of the widget.

Reimplemented from YWidget.

Definition at line 198 of file [YQContextMenu.cc](#).

### 3.27.2.2 preferredWidth()

```
int YQContextMenu::preferredWidth ( ) [virtual]
```

Preferred width of the widget.

Reimplemented from YWidget.

Definition at line 192 of file [YQContextMenu.cc](#).

### 3.27.2.3 rebuildMenuTree()

```
void YQContextMenu::rebuildMenuTree ( ) [virtual]
```

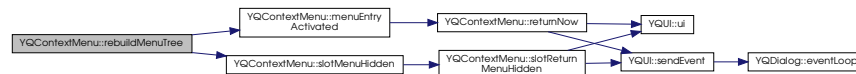
Change the label on the button.

Reimplemented from YContextMenu / YSelectionWidget. Rebuild the displayed menu tree from the internally stored YMenuItems.

Implemented from YContextMenu.

Definition at line 64 of file [YQContextMenu.cc](#).

Here is the call graph for this function:



### 3.27.2.4 setSize()

```
void YQContextMenu::setSize (
    int newWidth,
    int newHeight ) [virtual]
```

Set the new size of the widget.

Reimplemented from YWidget.

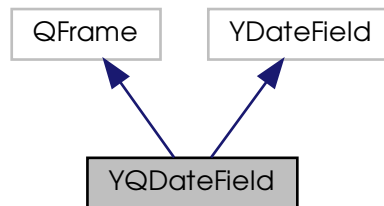
Definition at line 205 of file [YQContextMenu.cc](#).

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

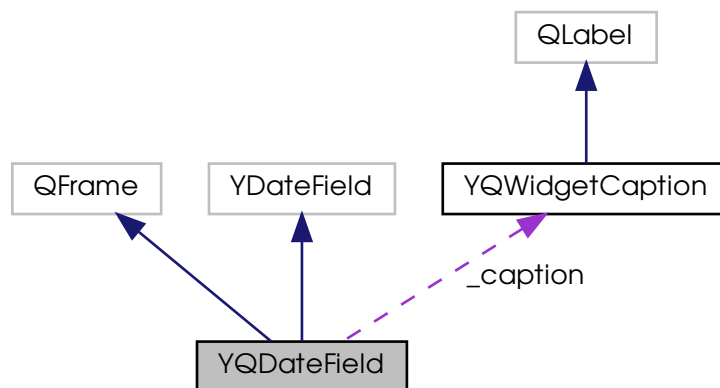
- `/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQContextMenu.h`
- `/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQContextMenu.cc`

## 3.28 YQDateField Class Reference

Inheritance diagram for YQDateField:



Collaboration diagram for YQDateField:



### Public Member Functions

- [YQDateField](#) (YWidget \*parent, const std::string &label)  
*Constructor.*
- virtual [~YQDateField](#) ()  
*Destructor.*
- virtual std::string [value](#) ()  
*Get the current value (the text entered by the user or set from the outside) of this input field.*

- virtual void [setValue](#) (const std::string &newValue)  
*Set the current value (the text entered by the user or set from the outside) of this input field.*
- virtual void [setLabel](#) (const std::string &label)  
*Set the label (the caption above the input field).*
- virtual void [setEnabled](#) (bool enabled)  
*Set enabled/disabled state.*
- virtual int [preferredWidth](#) ()  
*Preferred width of the widget.*
- virtual int [preferredHeight](#) ()  
*Preferred height of the widget.*
- virtual void [setSize](#) (int newWidth, int newHeight)  
*Set the new size of the widget.*
- virtual bool [setKeyboardFocus](#) ()  
*Accept the keyboard focus.*

### Protected Attributes

- [YQWidgetCaption](#) \* **\_caption**
- [QDateEdit](#) \* **\_qt\_dateEdit**

### 3.28.1 Detailed Description

Definition at line 37 of file [YQDateField.h](#).

### 3.28.2 Member Function Documentation

#### 3.28.2.1 preferredHeight()

```
int YQDateField::preferredHeight ( ) [virtual]
```

Preferred height of the widget.

Reimplemented from [YWidget](#).

Definition at line 108 of file [YQDateField.cc](#).

### 3.28.2.2 preferredWidth()

```
int YQDateField::preferredWidth ( ) [virtual]
```

Preferred width of the widget.

Reimplemented from YWidget.

Definition at line 102 of file [YQDateField.cc](#).

### 3.28.2.3 setEnabled()

```
void YQDateField::setEnabled (
    bool enabled ) [virtual]
```

Set enabled/disabled state.

Reimplemented from YWidget.

Definition at line 95 of file [YQDateField.cc](#).

### 3.28.2.4 setKeyboardFocus()

```
bool YQDateField::setKeyboardFocus ( ) [virtual]
```

Accept the keyboard focus.

Reimplemented from YWidget.

Definition at line 120 of file [YQDateField.cc](#).

### 3.28.2.5 setLabel()

```
void YQDateField::setLabel (
    const std::string & label ) [virtual]
```

Set the label (the caption above the input field).

Reimplemented from YSimpleInputField.

Definition at line 88 of file [YQDateField.cc](#).

Here is the call graph for this function:



### 3.28.2.6 setSize()

```
void YQDateField::setSize (
    int newWidth,
    int newHeight ) [virtual]
```

Set the new size of the widget.

Reimplemented from YWidget.

Definition at line 114 of file [YQDateField.cc](#).

### 3.28.2.7 setValue()

```
void YQDateField::setValue (
    const std::string & newValue ) [virtual]
```

Set the current value (the text entered by the user or set from the outside) of this input field.

Implemented from YSimpleInputField.

Definition at line 80 of file [YQDateField.cc](#).

### 3.28.2.8 value()

```
string YQDateField::value ( ) [virtual]
```

Get the current value (the text entered by the user or set from the outside) of this input field.

Implemented from YSimpleInputField.

Definition at line 74 of file [YQDateField.cc](#).

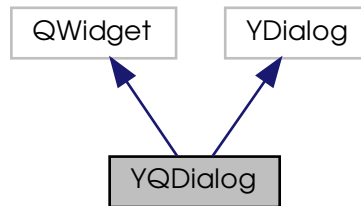
The documentation for this class was generated from the following files:

- [/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQDateField.h](#)
- [/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQDateField.cc](#)

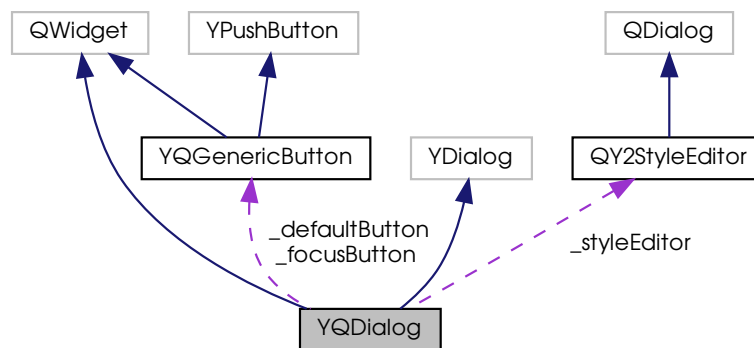


## 3.29 YQDialog Class Reference

Inheritance diagram for YQDialog:



Collaboration diagram for YQDialog:



### Public Member Functions

- [YQDialog](#) (YDialogType dialogType, YDialogColorMode colorMode=YDialogNormalColor)  
*Constructor.*
- void [closeEvent](#) (QCloseEvent \*ev)  
*Inherited from QDialog: The window was closed via the window manager close button.*
- virtual void [setEnabled](#) (bool enabled)  
*Set enabled/disabled state.*
- virtual int [preferredWidth](#) ()  
*Preferred width of the widget.*

- virtual int [preferredHeight](#) ()  
*Preferred height of the widget.*
- virtual void [setSize](#) (int newWidth, int newHeight)  
*Set the new size of the widget.*
- [YQGenericButton](#) \* [findDefaultButton](#) ()  
*Return this dialog's (first) default button or 0 if none.*
- bool [userResized](#) ()  
*Return 'true' if the user resized this dialog.*
- [YQGenericButton](#) \* [focusButton](#) () const  
*Returns the button that has the keyboard focus or 0 if no button has the keyboard focus.*
- [YQGenericButton](#) \* [defaultButton](#) () const  
*Returns the dialog's default button - the button that is activated with [Return] if no button has the keyboard focus.*
- void [losingFocus](#) ([YQGenericButton](#) \*button)  
*Notification that a button loses the keyboard focus.*
- void [gettingFocus](#) ([YQGenericButton](#) \*button)  
*Notification that a button gets the keyboard focus.*
- void [setDefaultButton](#) ([YQPushButton](#) \*newDefaultButton)  
*Set the dialog's default button - the button that is activated with [Return] if no other button has the keyboard focus.*
- void [ensureOnlyOneDefaultButton](#) ()  
*Ensure presence of no more than one single default button.*
- bool [activateDefaultButton](#) (bool warn=true)  
*Activate (i.e.*
- [YQWizard](#) \* [findWizard](#) () const  
*Find the first wizard in that dialog, if there is any.*
- [YQGenericButton](#) \* [wizardDefaultButton](#) ([YQWizard](#) \*wizard) const  
*Find a wizard button that would make sense as a default button.*
- virtual void [highlight](#) ([YWidget](#) \*child)  
*Highlight a child widget of this dialog.*
- [QEventLoop](#) \* [eventLoop](#) ()  
*Access to this dialog's event loop.*

## Static Public Member Functions

- static void [center](#) ([QWidget](#) \*dialog, [QWidget](#) \*parent=0)  
*Center a dialog relative to 'parent'.*

## Protected Slots

- void [waitForEventTimeout](#) ()  
*Timeout during waitForEvent()*

## Protected Member Functions

- virtual [~YQDialog](#) ()  
*Destructor.*
- [YQGenericButton](#) \* [findDefaultButton](#) (YWidgetListConstIterator begin, YWidgetListConstIterator end) const  
*Return the (first) default button between 'begin' and 'end' or 0 if there is none.*
- [YQWizard](#) \* [findWizard](#) (YWidgetListConstIterator begin, YWidgetListConstIterator end) const  
*Return the (first) wizard widget between 'begin' and 'end' or 0 if there is none.*
- [YQWizard](#) \* [ensureOnlyOneDefaultButton](#) (YWidgetListConstIterator begin, YWidgetListConstIterator end)  
*Helper function for [ensureOnlyOneDefaultButton\(\)](#): Recursively find all normal and wizard buttons between 'begin' and 'end' and make sure that no more than one button is marked as default.*
- virtual void [openInternal](#) ()  
*Internal open() method, called exactly once during the life time of the dialog in open().*
- virtual YEvent \* [waitForEventInternal](#) (int timeout\_millisec)  
*Wait for a user event.*
- virtual YEvent \* [pollEventInternal](#) ()  
*Check if a user event is pending.*
- virtual void [activate](#) ()  
*Activate this dialog: Make sure that it is shown as the topmost dialog of this application and that it can receive input.*
- virtual void [keyPressEvent](#) (QKeyEvent \*event)  
*Qt event handlers.*
- virtual void [focusInEvent](#) (QFocusEvent \*event)
- virtual void [resizeEvent](#) (QResizeEvent \*event)

## Static Protected Member Functions

- static QWidget \* [chooseParent](#) (YDialogType dialogType)  
*Choose a parent widget for a dialog of the specified type: Either the main window dock (if this is a YMainDialog and the dock currently accepts child dialogs) or 0.*

## Protected Attributes

- bool [\\_userResized](#)
- QSize [\\_userSize](#)
- [YQGenericButton](#) \* [\\_focusButton](#)
- [YQGenericButton](#) \* [\\_defaultButton](#)
- QTimer \* [\\_waitForEventTimer](#)
- QEventLoop \* [\\_eventLoop](#)
- YWidget \* [\\_highlightedChild](#)
- QPalette [\\_preHighlightPalette](#)
- bool [\\_preHighlightAutoFill](#)
- [QY2StyleEditor](#) \* [\\_styleEditor](#)

### 3.29.1 Detailed Description

Definition at line 42 of file [YQDialog.h](#).

### 3.29.2 Constructor & Destructor Documentation

#### 3.29.2.1 YQDialog()

```
YQDialog::YQDialog (
    YDialogType dialogType,
    YDialogColorMode colorMode = YDialogNormalColor )
```

Constructor.

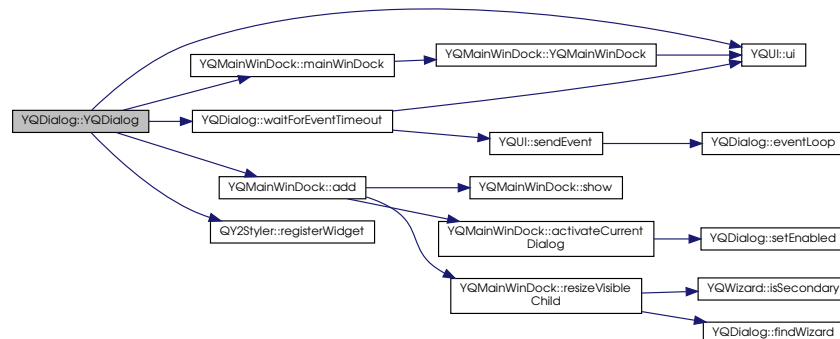
'dialogType' is one of YMainDialog, YWizardDialog, or YPopupDialog.

The Qt UI supports YWizardDialogs. They are handled very much like YMainDialogs, except for wizard dialogs that are opened over a wizard with a steps panel on the left side, in which case that new wizard dialog will be resized and moved so the steps panel from the wizard below will remain visible.

'colorMode' can be set to YDialogWarnColor to use very bright "warning" colors or YDialogInfoColor to use more prominent, yet not quite as bright as "warning" colors. Use both only very rarely.

Definition at line 60 of file [YQDialog.cc](#).

Here is the call graph for this function:



## 3.29.2.2 ~YQDialog()

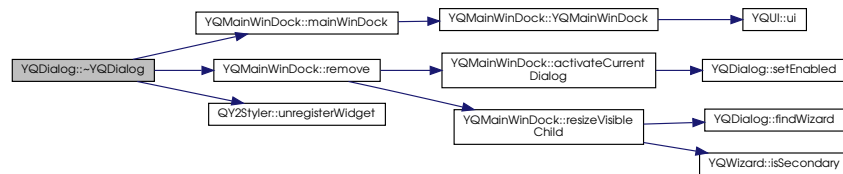
```
YQDialog::~YQDialog ( ) [protected], [virtual]
```

Destructor.

Don't delete a dialog directly, use YDialog::deleteTopmostDialog().

Definition at line 127 of file [YQDialog.cc](#).

Here is the call graph for this function:



## 3.29.3 Member Function Documentation

## 3.29.3.1 activate()

```
void YQDialog::activate ( ) [protected], [virtual]
```

Activate this dialog: Make sure that it is shown as the topmost dialog of this application and that it can receive input.

Implemented from YDialog.

Definition at line 185 of file [YQDialog.cc](#).

### 3.29.3.2 activateDefaultButton()

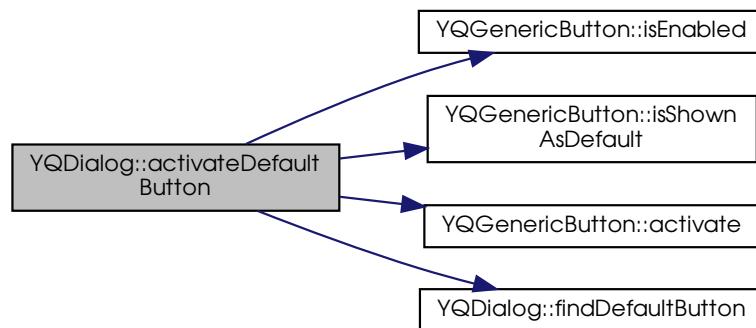
```
bool YQDialog::activateDefaultButton (
    bool warn = true )
```

Activate (i.e.

click) this dialog's default button, if there is any. Issue a warning to the log file if 'warn' is true.

Definition at line 536 of file [YQDialog.cc](#).

Here is the call graph for this function:



### 3.29.3.3 center()

```
void YQDialog::center (
    QWidget * dialog,
    QWidget * parent = 0 ) [static]
```

Center a dialog relative to 'parent'.

If 'parent' is 0, the dialog is centered relative to the application's main widget. If 'dialog' is the main widget and 'parent' is 0, the dialog is centered relative to the desktop.

Definition at line 845 of file [YQDialog.cc](#).

## 3.29.3.4 ensureOnlyOneDefaultButton()

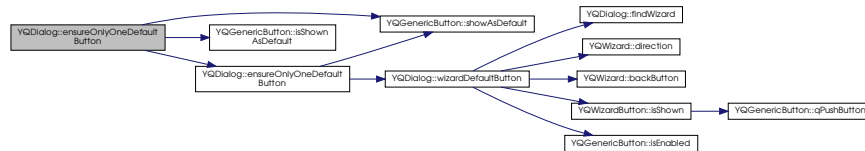
```
YQWizard * YQDialog::ensureOnlyOneDefaultButton (
    QWidgetListConstIterator begin,
    QWidgetListConstIterator end ) [protected]
```

Helper function for [ensureOnlyOneDefaultButton\(\)](#): Recursively find all normal and wizard buttons between 'begin' and 'end' and make sure that no more than one button is marked as default.

Return (the first) wizard widget found on the way.

Definition at line 351 of file [YQDialog.cc](#).

Here is the call graph for this function:



## 3.29.3.5 findWizard()

```
YQWizard * YQDialog::findWizard ( ) const
```

Find the first wizard in that dialog, if there is any.

Returns 0 if there is none.

Definition at line 427 of file [YQDialog.cc](#).

## 3.29.3.6 gettingFocus()

```
void YQDialog::gettingFocus (
    YQGenericButton * button )
```

Notification that a button gets the keyboard focus.

All pushbuttons are required to call this whenever they gain focus so the dialog can keep track of its focusButton.

Definition at line 591 of file [YQDialog.cc](#).

Here is the call graph for this function:







## 3.29.3.9 losingFocus()

```
void YQDialog::losingFocus (
    YQGenericButton * button )
```

Notification that a button loses the keyboard focus.

All pushbuttons are required to call this whenever they lose focus so the dialog can keep track of its focusButton.

Definition at line 575 of file [YQDialog.cc](#).

Here is the call graph for this function:



## 3.29.3.10 openInternal()

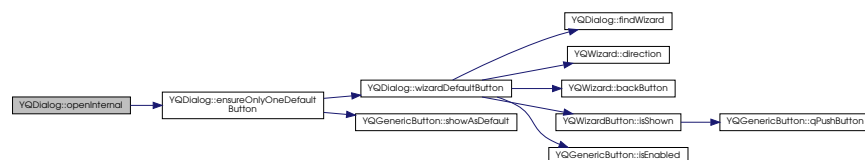
```
void YQDialog::openInternal ( ) [protected], [virtual]
```

Internal `open()` method, called exactly once during the life time of the dialog in `open()`.

Implemented from `YDialog`.

Definition at line 175 of file [YQDialog.cc](#).

Here is the call graph for this function:



### 3.29.3.11 pollEventInternal()

```
YEvent * YQDialog::pollEventInternal ( ) [protected], [virtual]
```

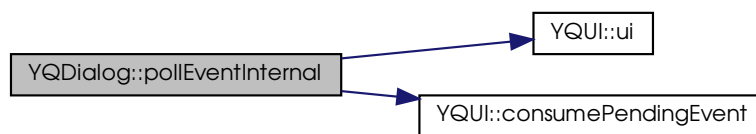
Check if a user event is pending.

If there is one, return it. If there is none, do not wait for one - return 0.

Implemented from YDialog.

Definition at line 811 of file [YQDialog.cc](#).

Here is the call graph for this function:



### 3.29.3.12 preferredHeight()

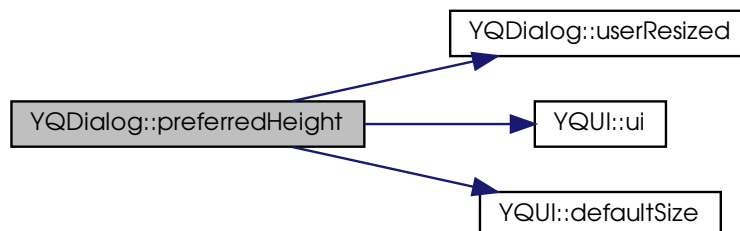
```
int YQDialog::preferredHeight ( ) [virtual]
```

Preferred height of the widget.

Reimplemented from YWidget.

Definition at line 224 of file [YQDialog.cc](#).

Here is the call graph for this function:



## 3.29.3.13 preferredWidth()

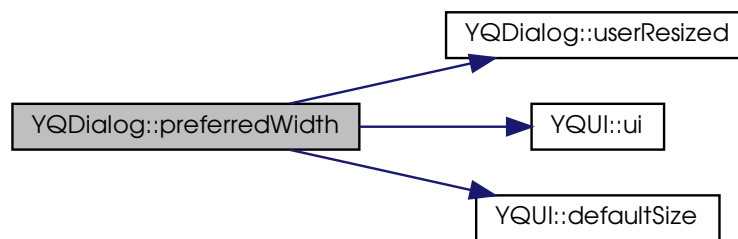
```
int YQDialog::preferredWidth ( ) [virtual]
```

Preferred width of the widget.

Reimplemented from YWidget.

Definition at line 193 of file [YQDialog.cc](#).

Here is the call graph for this function:



## 3.29.3.14 setDefaultButton()

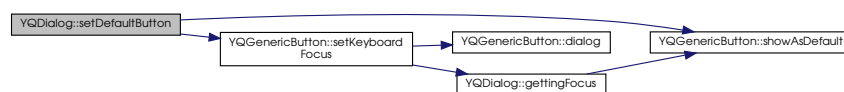
```
void YQDialog::setDefaultButton (
    YPushButton * newDefaultButton )
```

Set the dialog's default button - the button that is activated with [Return] if no other button has the keyboard focus.

'newDefaultButton' may be 0 if the former default button is destroyed.

Definition at line 496 of file [YQDialog.cc](#).

Here is the call graph for this function:



**3.29.3.15 setEnabled()**

```
void YQDialog::setEnabled (
    bool enabled ) [virtual]
```

Set enabled/disabled state.

Reimplemented from YWidget.

Definition at line 255 of file [YQDialog.cc](#).

**3.29.3.16 setSize()**

```
void YQDialog::setSize (
    int newWidth,
    int newHeight ) [virtual]
```

Set the new size of the widget.

Reimplemented from YWidget.

Definition at line 263 of file [YQDialog.cc](#).

**3.29.3.17 waitForEventInternal()**

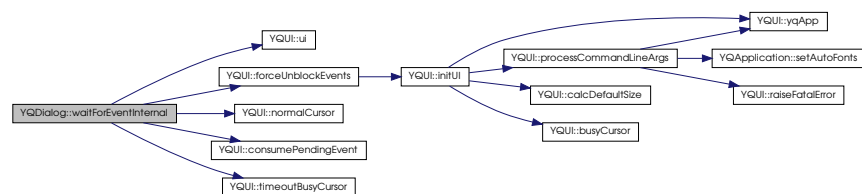
```
YEvent * YQDialog::waitForEventInternal (
    int timeout_millisec ) [protected], [virtual]
```

Wait for a user event.

Implemented from YDialog.

Definition at line 761 of file [YQDialog.cc](#).

Here is the call graph for this function:



## 3.29.3.18 wizardDefaultButton()

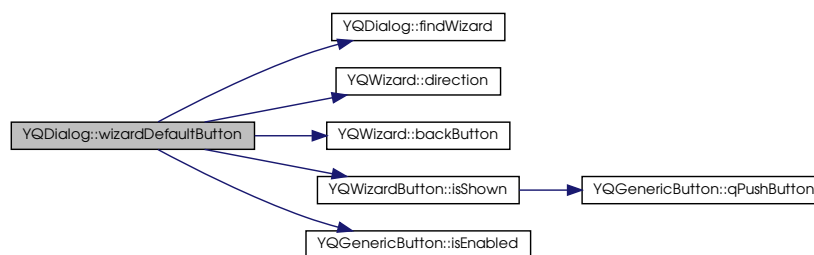
```
YQGenericButton * YQDialog::wizardDefaultButton (
    YQWizard * wizard ) const
```

Find a wizard button that would make sense as a default button.

Return 0 if none can be found.

Definition at line 459 of file [YQDialog.cc](#).

Here is the call graph for this function:

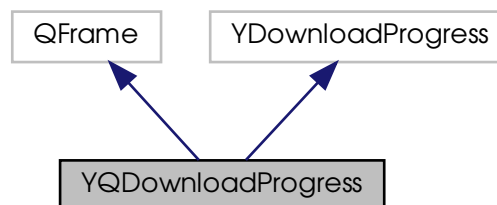


The documentation for this class was generated from the following files:

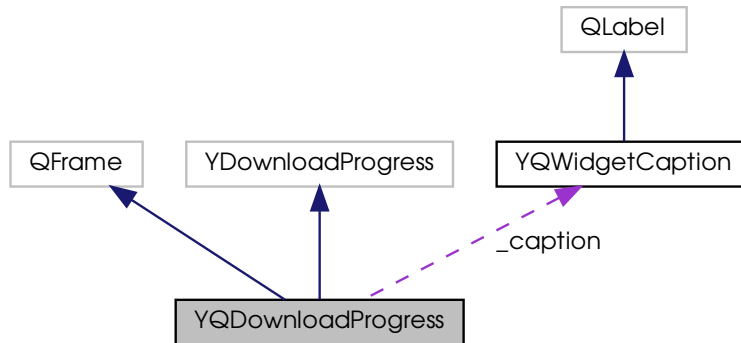
- `/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQDialog.h`
- `/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQDialog.cc`

## 3.30 YQDownloadProgress Class Reference

Inheritance diagram for YQDownloadProgress:



Collaboration diagram for YQDownloadProgress:



## Public Slots

- void [pollFileSize](#) (void)  
*Slot for polling and displaying the file size.*

## Public Member Functions

- [YQDownloadProgress](#) (YWidget \*parent, const std::string &label, const std::string &filename, YFileSize\_  
t expectedSize)  
*Constructor.*
- virtual [~YQDownloadProgress](#) ()  
*Destructor.*
- virtual void [setLabel](#) (const std::string &label)  
*Set the label (the text above the progress bar).*
- virtual void [setFilename](#) (const std::string &filename)  
*Set the name of a new file to monitor.*
- virtual void [setExpectedSize](#) (YFileSize\_t expectedSize)  
*Change the expected file size.*
- virtual void [setEnabled](#) (bool enabled)  
*Set enabled/disabled state.*
- virtual int [preferredWidth](#) ()  
*Preferred width of the widget.*
- virtual int [preferredHeight](#) ()  
*Preferred height of the widget.*
- virtual void [setSize](#) (int newWidth, int newHeight)  
*Set the new size of the widget.*

## Protected Attributes

- [YQWidgetCaption](#) \* `_caption`
- [QProgressBar](#) \* `_qt_progressBar`
- [QTimer](#) \* `_timer`

### 3.30.1 Detailed Description

Definition at line 36 of file [YQDownloadProgress.h](#).

### 3.30.2 Member Function Documentation

#### 3.30.2.1 preferredHeight()

```
int YQDownloadProgress::preferredHeight ( ) [virtual]
```

Preferred height of the widget.

Reimplemented from [YWidget](#).

Definition at line 127 of file [YQDownloadProgress.cc](#).

#### 3.30.2.2 preferredWidth()

```
int YQDownloadProgress::preferredWidth ( ) [virtual]
```

Preferred width of the widget.

Reimplemented from [YWidget](#).

Definition at line 120 of file [YQDownloadProgress.cc](#).

#### 3.30.2.3 setEnabled()

```
void YQDownloadProgress::setEnabled (
    bool enabled ) [virtual]
```

Set enabled/disabled state.

Reimplemented from [YWidget](#).

Definition at line 111 of file [YQDownloadProgress.cc](#).

### 3.30.2.4 `setExpectedSize()`

```
void YQDownloadProgress::setExpectedSize (
    YFileSize_t expectedSize ) [virtual]
```

Change the expected file size.

Reimplemented from `YDownloadProgress`.

Definition at line 96 of file [YQDownloadProgress.cc](#).

### 3.30.2.5 `setFilename()`

```
void YQDownloadProgress::setFilename (
    const std::string & filename ) [virtual]
```

Set the name of a new file to monitor.

Reimplemented from `YDownloadProgress`.

Definition at line 88 of file [YQDownloadProgress.cc](#).

### 3.30.2.6 `setLabel()`

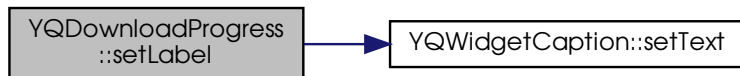
```
void YQDownloadProgress::setLabel (
    const std::string & label ) [virtual]
```

Set the label (the text above the progress bar).

Reimplemented from `YDownloadProgress`.

Definition at line 80 of file [YQDownloadProgress.cc](#).

Here is the call graph for this function:





### 3.30.2.7 setSize()

```
void YQDownloadProgress::setSize (
    int newWidth,
    int newHeight ) [virtual]
```

Set the new size of the widget.

Reimplemented from YWidget.

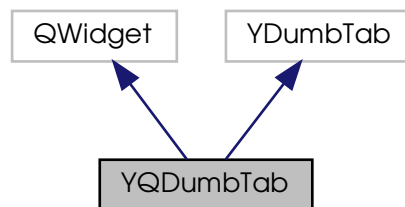
Definition at line 134 of file [YQDownloadProgress.cc](#).

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

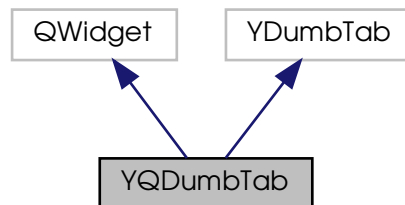
- /usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQDownloadProgress.h
- /usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQDownloadProgress.cc

## 3.31 YQDumbTab Class Reference

Inheritance diagram for YQDumbTab:



Collaboration diagram for YQDumbTab:



## Public Slots

- void [slotSelected](#) (int index)  
*Send an event that the tab with the specified index is selected.*

## Public Member Functions

- [YQDumbTab](#) (YWidget \*parent)  
*Constructor.*
- virtual [~YQDumbTab](#) ()  
*Destructor.*
- virtual void [addItem](#) (YItem \*item)  
*Add an item (a tab page).*
- virtual void [selectItem](#) (YItem \*item, bool selected=true)  
*Select or deselect an item.*
- virtual void [deselectAllItems](#) ()  
*Deselect all items.*
- virtual void [deleteAllItems](#) ()  
*Delete all items (all tab pages).*
- virtual void [shortcutChanged](#) ()  
*Notification that some shortcut was changed.*
- virtual void [setEnabled](#) (bool enabled)  
*Set enabled/disabled state.*
- virtual int [preferredWidth](#) ()  
*Preferred width of the widget.*
- virtual int [preferredHeight](#) ()  
*Preferred height of the widget.*
- virtual void [setSize](#) (int newWidth, int newHeight)  
*Set the new size of the widget.*

## Protected Attributes

- QTabBar \* [\\_tabBar](#)

### 3.31.1 Detailed Description

Definition at line 36 of file [YQDumbTab.h](#).

### 3.31.2 Member Function Documentation

#### 3.31.2.1 addItem()

```
void YQDumbTab::addItem (
    YItem * item ) [virtual]
```

Add an item (a tab page).

Reimplemented from YDumbTab.

Definition at line 74 of file [YQDumbTab.cc](#).

#### 3.31.2.2 deleteAllItems()

```
void YQDumbTab::deleteAllItems ( ) [virtual]
```

Delete all items (all tab pages).

Reimplemented from YSelectionWidget.

Definition at line 103 of file [YQDumbTab.cc](#).

#### 3.31.2.3 deselectAllItems()

```
void YQDumbTab::deselectAllItems ( ) [virtual]
```

Deselect all items.

Reimplemented from YSelectionWidget.

Definition at line 117 of file [YQDumbTab.cc](#).

#### 3.31.2.4 preferredHeight()

```
int YQDumbTab::preferredHeight ( ) [virtual]
```

Preferred height of the widget.

Reimplemented from YWidget.

Definition at line 170 of file [YQDumbTab.cc](#).

#### 3.31.2.5 preferredWidth()

```
int YQDumbTab::preferredWidth ( ) [virtual]
```

Preferred width of the widget.

Reimplemented from YWidget.

Definition at line 160 of file [YQDumbTab.cc](#).

#### 3.31.2.6 selectItem()

```
void YQDumbTab::selectItem (
    YItem * item,
    bool selected = true ) [virtual]
```

Select or deselect an item.

Reimplemented from YSelectionWidget.

Definition at line 88 of file [YQDumbTab.cc](#).

#### 3.31.2.7 setEnabled()

```
void YQDumbTab::setEnabled (
    bool enabled ) [virtual]
```

Set enabled/disabled state.

Reimplemented from YWidget.

Definition at line 152 of file [YQDumbTab.cc](#).

#### 3.31.2.8 setSize()

```
void YQDumbTab::setSize (
    int newWidth,
    int newHeight ) [virtual]
```

Set the new size of the widget.

Reimplemented from YWidget.

Definition at line 180 of file [YQDumbTab.cc](#).

### 3.31.2.9 shortcutChanged()

```
void YQDumbTab::shortcutChanged ( ) [virtual]
```

Notification that some shortcut was changed.

Reimplemented from YDumbTab.

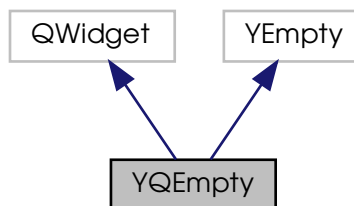
Definition at line 136 of file [YQDumbTab.cc](#).

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

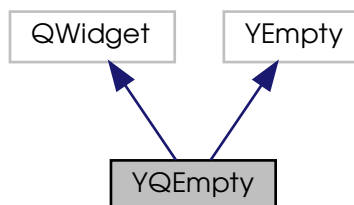
- /usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQDumbTab.h
- /usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQDumbTab.cc

## 3.32 YQEmpty Class Reference

Inheritance diagram for YQEmpty:



Collaboration diagram for YQEmpty:



## Public Member Functions

- [YQEmpty](#) (YWidget \*parent)  
*Constructor.*
- virtual [~YQEmpty](#) ()  
*Destructor.*
- virtual void [setSize](#) (int newWidth, int newHeight)  
*Set the new size of the widget.*

### 3.32.1 Detailed Description

Definition at line 32 of file [YQEmpty.h](#).

### 3.32.2 Member Function Documentation

#### 3.32.2.1 setSize()

```
void YQEmpty::setSize (  
    int newWidth,  
    int newHeight ) [virtual]
```

Set the new size of the widget.

Reimplemented from YWidget.

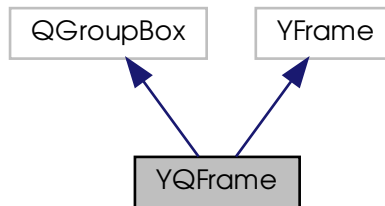
Definition at line 43 of file [YQEmpty.cc](#).

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

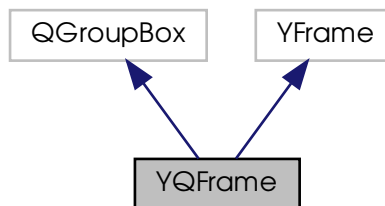
- /usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQEmpty.h
- /usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQEmpty.cc

### 3.33 YQFrame Class Reference

Inheritance diagram for YQFrame:



Collaboration diagram for YQFrame:



#### Public Member Functions

- [YQFrame](#) (YWidget \*parent, const std::string &label)  
*Constructor.*
- virtual [~YQFrame](#) ()  
*Destructor.*
- virtual void [setLabel](#) (const std::string &newLabel)  
*Change the frame label.*
- virtual void [setEnabled](#) (bool enabled)  
*Set enabled/disabled state.*
- virtual int [preferredWidth](#) ()  
*Preferred width of the widget.*
- virtual int [preferredHeight](#) ()  
*Preferred height of the widget.*
- virtual void [setSize](#) (int newWidth, int newHeight)  
*Set the new size of the widget.*

### 3.33.1 Detailed Description

Definition at line 35 of file [YQFrame.h](#).

### 3.33.2 Member Function Documentation

#### 3.33.2.1 preferredHeight()

```
int YQFrame::preferredHeight ( ) [virtual]
```

Preferred height of the widget.

Reimplemented from YWidget.

Definition at line 103 of file [YQFrame.cc](#).

#### 3.33.2.2 preferredWidth()

```
int YQFrame::preferredWidth ( ) [virtual]
```

Preferred width of the widget.

Reimplemented from YWidget.

Definition at line 88 of file [YQFrame.cc](#).

#### 3.33.2.3 setEnabled()

```
void YQFrame::setEnabled (
    bool enabled ) [virtual]
```

Set enabled/disabled state.

Reimplemented from YWidget.

Definition at line 53 of file [YQFrame.cc](#).



## 3.33.2.4 setSize()

```
void YQFrame::setSize (
    int newWidth,
    int newHeight ) [virtual]
```

Set the new size of the widget.

Reimplemented from YWidget.

Definition at line 61 of file [YQFrame.cc](#).

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

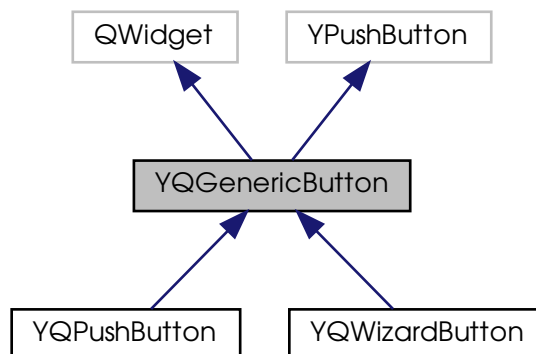
- [/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQFrame.h](#)
- [/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQFrame.cc](#)

## 3.34 YQGenericButton Class Reference

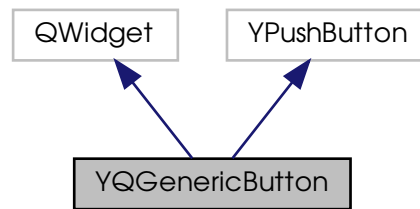
Abstract base class for push button and similar widgets - all that can become a [YQDialog](#)'s "default button".

```
#include <YQGenericButton.h>
```

Inheritance diagram for YQGenericButton:



Collaboration diagram for YQGenericButton:



## Public Slots

- void [activate](#) ()  
*Activate (animated) this button.*

## Public Member Functions

- virtual [~YQGenericButton](#) ()  
*Destructor.*
- virtual void [setEnabled](#) (bool enabled)  
*Set enabled/disabled state.*
- bool [isEnabled](#) () const  
*Returns 'true' if this button is enabled, 'false' otherwise.*
- void [setLabel](#) (const QString &label)  
*Changes the label (the text) of the button.*
- virtual void [setLabel](#) (const std::string &label)  
*Changes the label (the text) of the button.*
- void [showAsDefault](#) (bool show=true)  
*Show this button as the dialog's default button.*
- bool [isShownAsDefault](#) () const  
*Returns 'true' if this button is shown as a default button - which may mean that this really is the dialogs's default button or it is the dialog's focus button (a button that currently has the keyboard focus).*
- virtual bool [setKeyboardFocus](#) ()  
*Accept the keyboard focus.*
- virtual void [setIcon](#) (const std::string &iconName)  
*Set this button's icon.*
- QString [text](#) () const  
*Returns the button's text (label) - useful for log messages etc.*
- QPushButton \* [qPushButton](#) () const  
*Returns the internal Qt PushButton.*
- YQDialog \* [yQDialog](#) () const  
*Returns the internal parent dialog.*
- void [setShortcut](#) (const QKeySequence &key)  
*Set the keyboard shortcut (e.g.*

## Protected Member Functions

- [YQGenericButton](#) (YWidget \*parent, const std::string &label)  
*Constructor.*
- void [setQPushButton](#) (QPushButton \*pb)  
*Set the corresponding QPushButton.*
- bool [eventFilter](#) (QObject \*obj, QEvent \*event)  
*Redirect events from the \_qPushButton member to this object.*
- [YQDialog](#) \* [dialog](#) ()  
*Returns the corresponding YQDialog.*
- void **forgetDialog** ()

## Friends

- class **YQDialog**

### 3.34.1 Detailed Description

Abstract base class for push button and similar widgets - all that can become a [YQDialog](#)'s "default button".

Definition at line 44 of file [YQGenericButton.h](#).

### 3.34.2 Member Function Documentation

#### 3.34.2.1 [dialog\(\)](#)

```
YQDialog * YQGenericButton::dialog ( ) [protected]
```

Returns the corresponding [YQDialog](#).

Throws an exception if there is none.

Definition at line 82 of file [YQGenericButton.cc](#).

### 3.34.2.2 eventFilter()

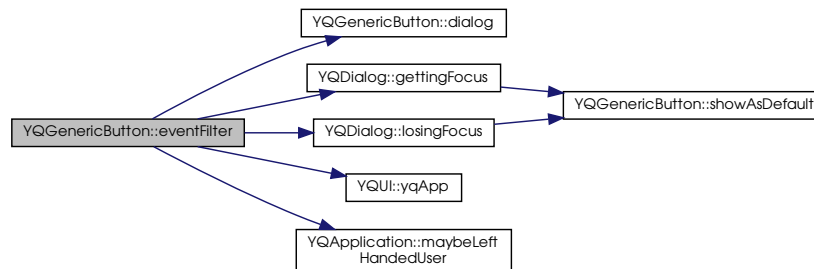
```
bool YQGenericButton::eventFilter (
    QObject * obj,
    QEvent * event ) [protected]
```

Redirect events from the `_qPushButton` member to this object.

Overwritten from `QObject`.

Definition at line 195 of file [YQGenericButton.cc](#).

Here is the call graph for this function:



### 3.34.2.3 isShownAsDefault()

```
bool YQGenericButton::isShownAsDefault ( ) const
```

Returns 'true' if this button is shown as a default button - which may mean that this really is the dialogs's default button or it is the dialog's focus button (a button that currently has the keyboard focus).

Don't confuse this with `YPushButton::isDefaultButton()`!

Definition at line 175 of file [YQGenericButton.cc](#).

### 3.34.2.4 setEnabled()

```
void YQGenericButton::setEnabled (
    bool enabled ) [virtual]
```

Set enabled/disabled state.

Reimplemented from `YWidget`.

Definition at line 98 of file [YQGenericButton.cc](#).

#### 3.34.2.5 setIcon()

```
void YQGenericButton::setIcon (
    const std::string & iconName ) [virtual]
```

Set this button's icon.

Reimplemented from YPushButton.

Definition at line 113 of file [YQGenericButton.cc](#).

#### 3.34.2.6 setLabel()

```
void YQGenericButton::setLabel (
    const std::string & label ) [virtual]
```

Changes the label (the text) of the button.

Reimplemented from YWidget.

Definition at line 153 of file [YQGenericButton.cc](#).

#### 3.34.2.7 setShortcut()

```
void YQGenericButton::setShortcut (
    const QKeySequence & key )
```

Set the keyboard shortcut (e.g.

F1 for help)

Definition at line 237 of file [YQGenericButton.cc](#).

#### 3.34.2.8 showAsDefault()

```
void YQGenericButton::showAsDefault (
    bool show = true )
```

Show this button as the dialog's default button.

The button never calls this by itself - the parent dialog is responsible for that.

Definition at line 164 of file [YQGenericButton.cc](#).

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

- [/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQGenericButton.h](#)
- [/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQGenericButton.cc](#)

### 3.35 YQGraphPluginIf Class Reference

#### Public Member Functions

- virtual YGraph \* **createGraph** (YWidget \*parent, const std::string &filename, const std::string &layout↵ Algorithm)=0
- virtual YGraph \* **createGraph** (YWidget \*parent, void \*graph)=0

#### 3.35.1 Detailed Description

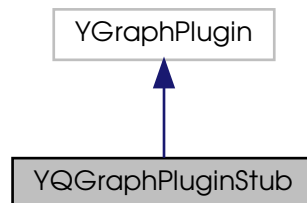
Definition at line 32 of file [YQGraphPluginIf.h](#).

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

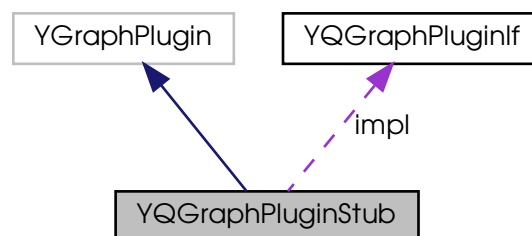
- /usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQGraphPluginIf.h

### 3.36 YQGraphPluginStub Class Reference

Inheritance diagram for YQGraphPluginStub:



Collaboration diagram for YQGraphPluginStub:



## Public Member Functions

- [YQGraphPluginStub](#) ()  
*Constructor: Load the plugin library for the package selector.*
- virtual [~YQGraphPluginStub](#) ()  
*Destructor.*
- virtual YGraph \* [createGraph](#) (YWidget \*parent, const std::string &filename, const std::string &layoutAlgorithm)  
*Create a package selector.*
- virtual YGraph \* **createGraph** (YWidget \*parent, void \*graph)

## Public Attributes

- [YQGraphPluginIf](#) \* **impl**

### 3.36.1 Detailed Description

Definition at line 41 of file [YQGraphPluginStub.h](#).

### 3.36.2 Member Function Documentation

#### 3.36.2.1 createGraph()

```
YGraph * YQGraphPluginStub::createGraph (  
    YWidget * parent,  
    const std::string & filename,  
    const std::string & layoutAlgorithm ) [virtual]
```

Create a package selector.

Implemented from YGraphPlugin.

This might return 0 if the plugin lib could not be loaded or if the appropriate symbol could not be located in the plugin lib.

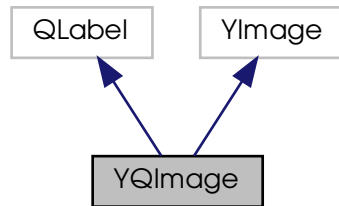
Definition at line 58 of file [YQGraphPluginStub.cc](#).

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

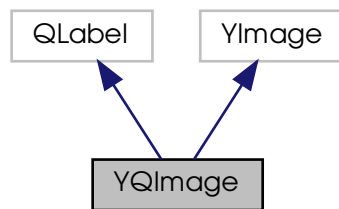
- /usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQGraphPluginStub.h
- /usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQGraphPluginStub.cc

### 3.37 YQImage Class Reference

Inheritance diagram for YQImage:



Collaboration diagram for YQImage:



#### Public Member Functions

- [YQImage](#) (YWidget \*parent, const std::string &imageFileName, bool animated=false)  
*Constructor.*
- virtual [~YQImage](#) ()  
*Destructor.*
- virtual void [setImage](#) (const std::string &imageFileName, bool animated=false)  
*Set and display a new image.*
- virtual void [setAutoScale](#) (bool autoScale=true)  
*Make the image fit into the available space.*
- virtual void [setEnabled](#) (bool enabled)  
*if false, the image will be displayed in gray*
- virtual int [preferredWidth](#) ()



*Preferred width of the widget.*

- virtual int [preferredHeight](#) ()

*Preferred height of the widget.*

- virtual void [setSize](#) (int newWidth, int newHeight)

*Set the new size of the widget.*

## Protected Attributes

- int [\\_pixmapWidth](#)
- int [\\_pixmapHeight](#)

### 3.37.1 Detailed Description

Definition at line 36 of file [YQImage.h](#).

### 3.37.2 Constructor & Destructor Documentation

#### 3.37.2.1 YQImage()

```
YQImage::YQImage (
    YWidget * parent,
    const std::string & imageFileName,
    bool animated = false )
```

Constructor.

'animated' indicates if 'imageFileName' is an animated image format (e.g., MNG).

Definition at line 39 of file [YQImage.cc](#).

Here is the call graph for this function:



### 3.37.3 Member Function Documentation

#### 3.37.3.1 preferredHeight()

```
int YQImage::preferredHeight ( ) [virtual]
```

Preferred height of the widget.

Reimplemented from YWidget.

Definition at line 154 of file [YQImage.cc](#).

#### 3.37.3.2 preferredWidth()

```
int YQImage::preferredWidth ( ) [virtual]
```

Preferred width of the widget.

Reimplemented from YWidget.

Definition at line 132 of file [YQImage.cc](#).

#### 3.37.3.3 setAutoScale()

```
void YQImage::setAutoScale (
    bool autoScale = true ) [virtual]
```

Make the image fit into the available space.

Reimplemented from YImage.

Definition at line 119 of file [YQImage.cc](#).

Here is the call graph for this function:



#### 3.37.3.4 setImage()

```
void YQImage::setImage (
    const std::string & imageFileName,
    bool animated = false ) [virtual]
```

Set and display a new image.

Reimplemented from YImage.

Definition at line 63 of file [YQImage.cc](#).

#### 3.37.3.5 setSize()

```
void YQImage::setSize (
    int newWidth,
    int newHeight ) [virtual]
```

Set the new size of the widget.

Reimplemented from YWidget.

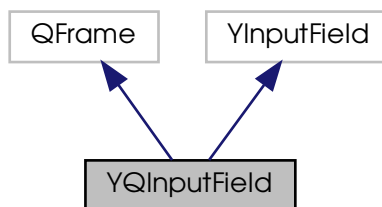
Definition at line 176 of file [YQImage.cc](#).

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

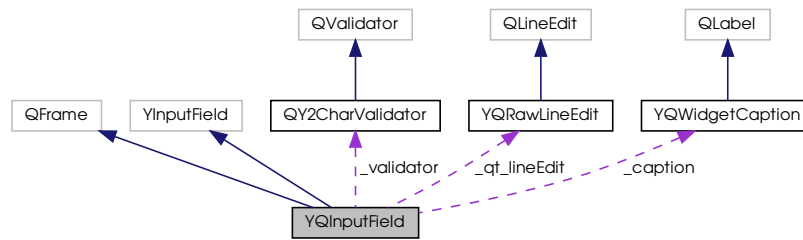
- [/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQImage.h](#)
- [/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQImage.cc](#)

## 3.38 YQInputField Class Reference

Inheritance diagram for YQInputField:



Collaboration diagram for YQInputField:



## Public Member Functions

- `YQInputField` (`YWidget *parent`, `const std::string &label`, `bool passwordMode=false`)  
*Constructor.*
- virtual `std::string value` ()  
*Get the current value (the text entered by the user or set from the outside) of this input field.*
- virtual void `setValue` (`const std::string &text`)  
*Set the current value (the text entered by the user or set from the outside) of this input field.*
- virtual void `setLabel` (`const std::string &label`)  
*Set the label (the caption above the input field).*
- virtual void `setValidChars` (`const std::string &validChars`)  
*Set the valid input characters.*
- virtual void `setInputMaxLength` (`int numberOfChars`)  
*Specify the amount of characters which can be inserted.*
- bool `isValidText` (`const QString &text`) const  
*Returns 'true' if a given text is valid according to ValidChars.*
- virtual void `setEnabled` (`bool enabled`)  
*Set enabled/disabled state.*
- virtual int `preferredWidth` ()  
*Preferred width of the widget.*
- virtual int `preferredHeight` ()  
*Preferred height of the widget.*
- virtual void `setSize` (`int newWidth`, `int newHeight`)  
*Set the new size of the widget.*
- virtual bool `setKeyboardFocus` ()  
*Accept the keyboard focus.*

## Protected Slots

- void `changed` (`const QString &`)  
*Triggered when the text in the InputField changes.*
- void `displayCapsLockWarning` ()  
*Display a warning that CapsLock is active: Replace the label with "CapsLock!".*
- void `clearCapsLockWarning` ()  
*Clear the CapsLock warning: Restore old label.*

## Protected Attributes

- [YQWidgetCaption](#) \* **\_caption**
- [YQRawLineEdit](#) \* **\_qt\_lineEdit**
- [QY2CharValidator](#) \* **\_validator**
- bool **\_shrinkable**
- bool **\_displayingCapsLockWarning**

### 3.38.1 Detailed Description

Definition at line 43 of file [YQInputField.h](#).

### 3.38.2 Member Function Documentation

#### 3.38.2.1 changed

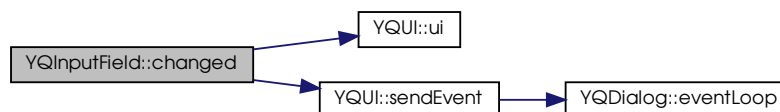
```
void YQInputField::changed (  
    const QString & ) [protected], [slot]
```

Triggered when the text in the InputField changes.

This *may* be of interest to the module.

Definition at line 212 of file [YQInputField.cc](#).

Here is the call graph for this function:



### 3.38.2.2 preferredHeight()

```
int YQInputField::preferredHeight ( ) [virtual]
```

Preferred height of the widget.

Reimplemented from YWidget.

Definition at line 138 of file [YQInputField.cc](#).

### 3.38.2.3 preferredWidth()

```
int YQInputField::preferredWidth ( ) [virtual]
```

Preferred width of the widget.

Reimplemented from YWidget.

Definition at line 127 of file [YQInputField.cc](#).

### 3.38.2.4 setEnabled()

```
void YQInputField::setEnabled (
    bool enabled ) [virtual]
```

Set enabled/disabled state.

Reimplemented from YWidget.

Definition at line 119 of file [YQInputField.cc](#).

### 3.38.2.5 setInputMaxLength()

```
void YQInputField::setInputMaxLength (
    int numberOfChars ) [virtual]
```

Specify the amount of characters which can be inserted.

Reimplemented from YInputField.

Definition at line 197 of file [YQInputField.cc](#).

### 3.38.2.6 setKeyboardFocus()

```
bool YQInputField::setKeyboardFocus ( ) [virtual]
```

Accept the keyboard focus.

Reimplemented from YWidget.

Definition at line 203 of file [YQInputField.cc](#).

### 3.38.2.7 setLabel()

```
void YQInputField::setLabel (
    const std::string & label ) [virtual]
```

Set the label (the caption above the input field).

Reimplemented from YInputField.

Definition at line 150 of file [YQInputField.cc](#).

Here is the call graph for this function:



### 3.38.2.8 setSize()

```
void YQInputField::setSize (
    int newWidth,
    int newHeight ) [virtual]
```

Set the new size of the widget.

Reimplemented from YWidget.

Definition at line 144 of file [YQInputField.cc](#).

### 3.38.2.9 setValidChars()

```
void YQInputField::setValidChars (
    const std::string & validChars ) [virtual]
```

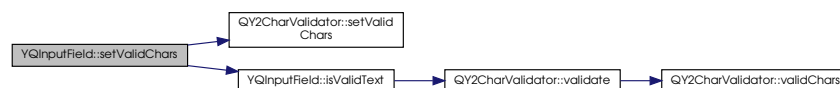
Set the valid input characters.

No input validation is performed (i.e., the user can enter anything) if this is empty.

Reimplemented from YInputField.

Definition at line 169 of file [YQInputField.cc](#).

Here is the call graph for this function:



### 3.38.2.10 setValue()

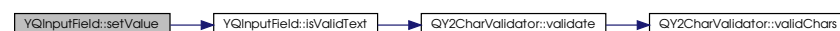
```
void YQInputField::setValue (
    const std::string & text ) [virtual]
```

Set the current value (the text entered by the user or set from the outside) of this input field.

Reimplemented from YInputField.

Definition at line 103 of file [YQInputField.cc](#).

Here is the call graph for this function:





#### 3.38.2.11 value()

```
string YQIntField::value ( ) [virtual]
```

Get the current value (the text entered by the user or set from the outside) of this input field.

Reimplemented from YIntField.

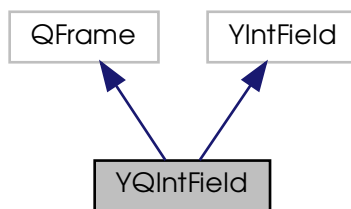
Definition at line 97 of file [YQIntField.cc](#).

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

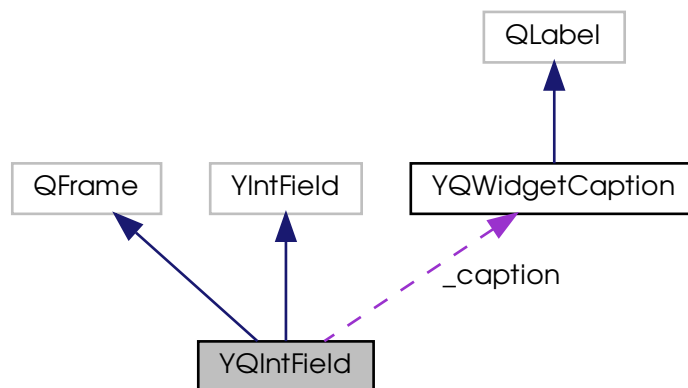
- /usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQIntField.h
- /usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQIntField.cc

## 3.39 YQIntField Class Reference

Inheritance diagram for YQIntField:



Collaboration diagram for YQIntField:



## Signals

- void [valueChanged](#) (int newValue)  
*Emitted when the value changes (regardless of the notify flag).*

## Public Member Functions

- [YQIntField](#) (YWidget \*parent, const std::string &label, int minValue, int maxValue, int initialValue)  
*Constructor.*
- virtual [~YQIntField](#) ()  
*Destructor.*
- virtual int [value](#) ()  
*Get the current value (the number entered by the user or set from the outside) of this IntField.*
- virtual void [setLabel](#) (const std::string &label)  
*Set the label (the caption above the input field).*
- virtual void [setEnabled](#) (bool enabled)  
*Sets the widget's enabled state.*
- virtual int [preferredWidth](#) ()  
*Preferred width of the widget.*
- virtual int [preferredHeight](#) ()  
*Preferred height of the widget.*
- virtual void [setSize](#) (int newWidth, int newHeight)  
*Set the new size of the widget.*
- virtual bool [setKeyboardFocus](#) ()  
*Accept the keyboard focus.*

## Protected Slots

- void [valueChangedSlot](#) (int newValue)  
*Slot for "value changed".*

## Protected Member Functions

- virtual void [setValueInternal](#) (int val)  
*Set the current value (the number entered by the user or set from the outside) of this IntField.*

## Protected Attributes

- [YQWidgetCaption](#) \* **\_caption**
- [QSpinBox](#) \* **\_qt\_spinBox**

### 3.39.1 Detailed Description

Definition at line 39 of file [YQIntField.h](#).

## 3.39.2 Member Function Documentation

### 3.39.2.1 preferredHeight()

```
int YQIntField::preferredHeight ( ) [virtual]
```

Preferred height of the widget.

Reimplemented from YWidget.

Definition at line 133 of file [YQIntField.cc](#).

### 3.39.2.2 preferredWidth()

```
int YQIntField::preferredWidth ( ) [virtual]
```

Preferred width of the widget.

Reimplemented from YWidget.

Definition at line 126 of file [YQIntField.cc](#).

### 3.39.2.3 setEnabled()

```
void YQIntField::setEnabled (
    bool enabled ) [virtual]
```

Sets the widget's enabled state.

Inherited from YWidget.

Definition at line 117 of file [YQIntField.cc](#).

### 3.39.2.4 setKeyboardFocus()

```
bool YQIntField::setKeyboardFocus ( ) [virtual]
```

Accept the keyboard focus.

Reimplemented from YWidget.

Definition at line 147 of file [YQIntField.cc](#).

### 3.39.2.5 setLabel()

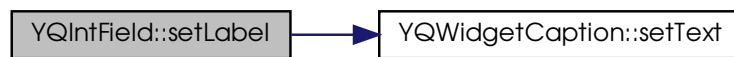
```
void YQIntField::setLabel (
    const std::string & label ) [virtual]
```

Set the label (the caption above the input field).

Reimplemented from YIntField.

Definition at line 109 of file [YQIntField.cc](#).

Here is the call graph for this function:



### 3.39.2.6 setSize()

```
void YQIntField::setSize (
    int newWidth,
    int newHeight ) [virtual]
```

Set the new size of the widget.

Reimplemented from YWidget.

Definition at line 140 of file [YQIntField.cc](#).

### 3.39.2.7 setValueInternal()

```
void YQIntField::setValueInternal (
    int val ) [protected], [virtual]
```

Set the current value (the number entered by the user or set from the outside) of this IntField.

'val' is guaranteed to be between minValue and maxValue; no further checks are required.

Implemented from YIntField.

Definition at line 92 of file [YQIntField.cc](#).

### 3.39.2.8 value()

```
int YQIntField::value ( ) [virtual]
```

Get the current value (the number entered by the user or set from the outside) of this IntField.

Implemented from YIntField.

Definition at line 85 of file [YQIntField.cc](#).

### 3.39.2.9 valueChangedSlot

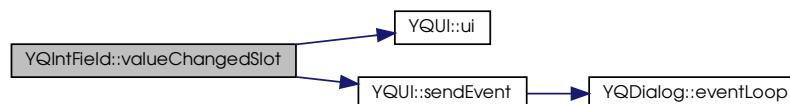
```
void YQIntField::valueChangedSlot (
    int newValue ) [protected], [slot]
```

Slot for "value changed".

This will send a ValueChanged event if 'notify' is set.

Definition at line 100 of file [YQIntField.cc](#).

Here is the call graph for this function:

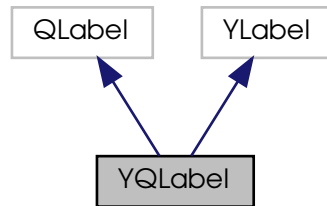


The documentation for this class was generated from the following files:

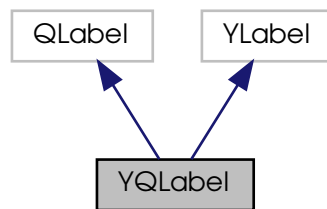
- `/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQIntField.h`
- `/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQIntField.cc`

### 3.40 YQLabel Class Reference

Inheritance diagram for YQLabel:



Collaboration diagram for YQLabel:



#### Public Member Functions

- [YQLabel](#) (YWidget \*parent, const std::string &text, bool isHeading=false, bool isOutputField=false)  
*Constructor.*
- virtual [~YQLabel](#) ()  
*Destructor.*
- virtual void [setText](#) (const std::string &newText)  
*Set the text the widget displays.*
- virtual void [setUseBoldFont](#) (bool bold)  
*Switch bold font on or off.*
- virtual void [setEnabled](#) (bool enabled)  
*Set enabled / disabled state.*
- virtual int [preferredWidth](#) ()

*Preferred width of the widget.*

- virtual int [preferredHeight](#) ()

*Preferred height of the widget.*

- virtual void [setSize](#) (int newWidth, int newHeight)

*Set the new size of the widget.*

### 3.40.1 Detailed Description

Definition at line 33 of file [YQLabel.h](#).

### 3.40.2 Member Function Documentation

#### 3.40.2.1 [preferredHeight\(\)](#)

```
int YQLabel::preferredHeight ( ) [virtual]
```

Preferred height of the widget.

Reimplemented from YWidget.

Definition at line 103 of file [YQLabel.cc](#).

#### 3.40.2.2 [preferredWidth\(\)](#)

```
int YQLabel::preferredWidth ( ) [virtual]
```

Preferred width of the widget.

Reimplemented from YWidget.

Definition at line 97 of file [YQLabel.cc](#).

#### 3.40.2.3 [setEnabled\(\)](#)

```
void YQLabel::setEnabled (
    bool enabled ) [virtual]
```

Set enabled / disabled state.

Reimplemented from YWidget.

Definition at line 90 of file [YQLabel.cc](#).

#### 3.40.2.4 setSize()

```
void YQLabel::setSize (
    int newWidth,
    int newHeight ) [virtual]
```

Set the new size of the widget.

Reimplemented from YWidget.

Definition at line 109 of file [YQLabel.cc](#).

#### 3.40.2.5 setText()

```
void YQLabel::setText (
    const std::string & newText ) [virtual]
```

Set the text the widget displays.

Reimplemented from YLabel.

Definition at line 73 of file [YQLabel.cc](#).

#### 3.40.2.6 setUseBoldFont()

```
void YQLabel::setUseBoldFont (
    bool bold ) [virtual]
```

Switch bold font on or off.

Reimplemented from YLabel.

Definition at line 80 of file [YQLabel.cc](#).

Here is the call graph for this function:



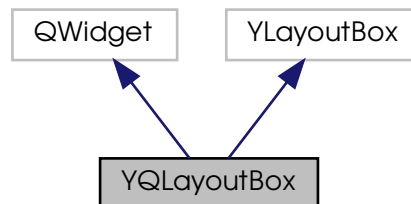
The documentation for this class was generated from the following files:

- `/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQLabel.h`
- `/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQLabel.cc`

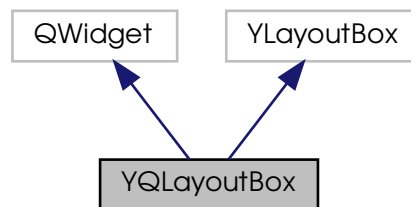


## 3.41 YQLayoutBox Class Reference

Inheritance diagram for YQLayoutBox:



Collaboration diagram for YQLayoutBox:



### Public Member Functions

- [YQLayoutBox](#) (YWidget \*parent, YUIDimension dim)  
*Constructor.*
- virtual [~YQLayoutBox](#) ()  
*Destructor.*
- virtual void [setEnabled](#) (bool enabled)  
*Set enabled/disabled state.*
- virtual void [setSize](#) (int newWidth, int newHeight)  
*Set the new size of the widget.*
- virtual void [moveChild](#) (YWidget \*child, int newX, int newY)  
*Move a child widget to a new position.*

## Protected Member Functions

- void [ensureChildrenVisible](#) ()  
*Go through the (direct) YWidget children list and make sure the corresponding QWidget is visible.*

## Protected Attributes

- bool [\\_needToEnsureChildrenVisible](#)

### 3.41.1 Detailed Description

Definition at line 34 of file [YQLayoutBox.h](#).

### 3.41.2 Constructor & Destructor Documentation

#### 3.41.2.1 YQLayoutBox()

```
YQLayoutBox::YQLayoutBox (
    YWidget * parent,
    YUIDimension dim )
```

Constructor.

Creates a VBox for dim == YD\_VERT or a HBox for YD\_HORIZ.

Definition at line 31 of file [YQLayoutBox.cc](#).

### 3.41.3 Member Function Documentation

#### 3.41.3.1 moveChild()

```
void YQLayoutBox::moveChild (
    YWidget * child,
    int newX,
    int newY ) [virtual]
```

Move a child widget to a new position.

Reimplemented from YLayoutBox.

Definition at line 77 of file [YQLayoutBox.cc](#).

### 3.41.3.2 `setEnabled()`

```
void YQLayoutBox::setEnabled (
    bool enabled ) [virtual]
```

Set enabled/disabled state.

Reimplemented from YWidget.

Definition at line 47 of file [YQLayoutBox.cc](#).

### 3.41.3.3 `setSize()`

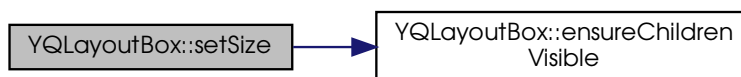
```
void YQLayoutBox::setSize (
    int newWidth,
    int newHeight ) [virtual]
```

Set the new size of the widget.

Reimplemented from YWidget.

Definition at line 54 of file [YQLayoutBox.cc](#).

Here is the call graph for this function:

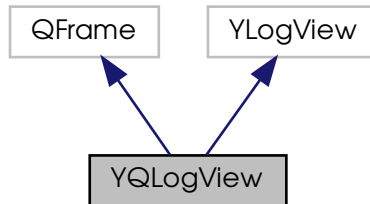


The documentation for this class was generated from the following files:

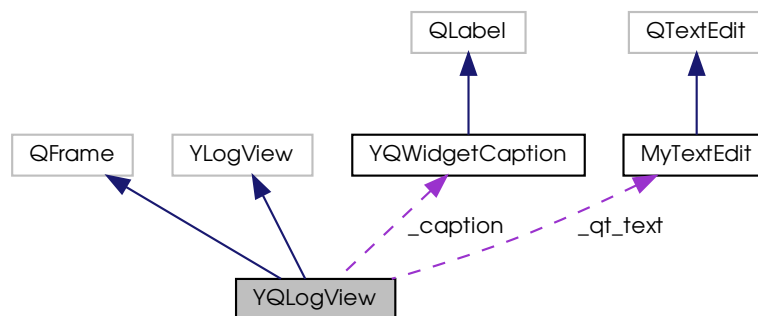
- `/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQLayoutBox.h`
- `/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQLayoutBox.cc`

### 3.42 YQLogView Class Reference

Inheritance diagram for YQLogView:



Collaboration diagram for YQLogView:



#### Public Member Functions

- [YQLogView](#) (YWidget \*parent, const std::string &label, int visibleLines, int maxLines)  
*Constructor.*
- virtual [~YQLogView](#) ()  
*Destructor.*
- virtual void [setLabel](#) (const std::string &label)  
*Set the label (the caption above the log text).*
- virtual void [setEnabled](#) (bool enabled)  
*Set enabled/disabled state.*
- virtual int [preferredWidth](#) ()

- Preferred width of the widget.*
- virtual int [preferredHeight](#) ()  
*Preferred height of the widget.*
- virtual void [setSize](#) (int newWidth, int newHeight)  
*Set the new size of the widget.*
- virtual bool [setKeyboardFocus](#) ()  
*Accept the keyboard focus.*

## Protected Member Functions

- virtual void [displayLogText](#) (const std::string &text)  
*Display the part of the log text that should be displayed.*

## Protected Attributes

- [YQWidgetCaption](#) \* [\\_caption](#)
- [MyTextEdit](#) \* [\\_qt\\_text](#)
- QString [\\_lastText](#)

### 3.42.1 Detailed Description

Definition at line 39 of file [YQLogView.h](#).

### 3.42.2 Member Function Documentation

#### 3.42.2.1 [displayLogText\(\)](#)

```
void YQLogView::displayLogText (  
    const std::string & text ) [protected], [virtual]
```

Display the part of the log text that should be displayed.

'text' contains the last 'visibleLines()' lines. This is called whenever the log text changes. Note that the text might also be empty, in which case the displayed log text should be cleared.

Implemented from YLogView.

Definition at line 81 of file [YQLogView.cc](#).

### 3.42.2.2 preferredHeight()

```
int YQLogView::preferredHeight ( ) [virtual]
```

Preferred height of the widget.

Reimplemented from YWidget.

Definition at line 144 of file [YQLogView.cc](#).

### 3.42.2.3 preferredWidth()

```
int YQLogView::preferredWidth ( ) [virtual]
```

Preferred width of the widget.

Reimplemented from YWidget.

Definition at line 137 of file [YQLogView.cc](#).

### 3.42.2.4 setEnabled()

```
void YQLogView::setEnabled (
    bool enabled ) [virtual]
```

Set enabled/disabled state.

Reimplemented from YWidget.

Definition at line 128 of file [YQLogView.cc](#).

### 3.42.2.5 setLabel()

```
void YQLogView::setLabel (
    const std::string & label ) [virtual]
```

Set the label (the caption above the log text).

Reimplemented from YLogView.

Definition at line 119 of file [YQLogView.cc](#).

Here is the call graph for this function:



#### 3.42.2.6 setSize()

```
void YQLogView::setSize (
    int newWidth,
    int newHeight ) [virtual]
```

Set the new size of the widget.

Reimplemented from YWidget.

Definition at line 172 of file [YQLogView.cc](#).

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

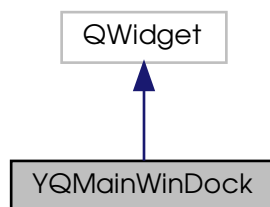
- [/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQLogView.h](#)
- [/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQLogView.cc](#)

### 3.43 YQMainWinDock Class Reference

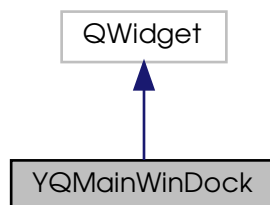
Container window for YQDialogs of type YMainWindowDialog:

```
#include <YQMainWinDock.h>
```

Inheritance diagram for YQMainWinDock:



Collaboration diagram for YQMainWinDock:



## Public Member Functions

- void [add](#) ([YQDialog](#) \*dialog)  
*Add a dialog (the widgetRep() of a [YQDialog](#)) to the MainWinDock (on top of its widget stack).*
- void [remove](#) ([YQDialog](#) \*dialog=0)  
*Remove a dialog from the MainWinDock (if it belongs to the MainWinDock).*
- [YQDialog](#) \* [topmostDialog](#) () const  
*Return the current topmost dialog (the widgetRep() of a [YQDialog](#)) or 0 if there is none.*
- bool [couldDock](#) ()  
*Return 'true' if the next main dialog could be docked, i.e., if there is either no open dialog at all or only main dialogs.*
- virtual void [show](#) ()  
*Show the widget (make it visible).*
- virtual void [closeEvent](#) (QCloseEvent \*event)  
*Window manager close event (Alt-F4): Send a YCancelEvent and let the application handle that event.*

## Static Public Member Functions

- static [YQMainWinDock](#) \* [mainWinDock](#) ()  
*Static method to access the singleton for this class.*

## Protected Slots

- void [showCurrentDialog](#) ()  
*Show the current dialog.*

## Protected Member Functions

- [YQMainWinDock](#) ()  
*Constructor.*
- virtual [~YQMainWinDock](#) ()  
*Destructor.*
- virtual void [paintEvent](#) (QPaintEvent \*event)  
*Paint event.*
- virtual void [resizeEvent](#) (QResizeEvent \*event)  
*Resize event.*
- void [resizeVisibleChild](#) ()  
*Resize the visible child to the current size of the dock.*
- void [setSideBarWidth](#) (int width)  
*For secondary wizards.*
- void [activateCurrentDialog](#) (bool active)  
*Activate or deactivate the next-lower dialog in the dock when a new dialog is opened or when a dialog is closed.*

## Friends

- class [YQWizard](#)



### 3.43.1 Detailed Description

Container window for YQDialogs of type YMainWindowDialog:

This widget "swallows" any main dialogs it gets so only the topmost of them is visible at any given time. It acts as a window stack for main dialogs, making the next lower dialog on the stack visible as when the (previously) topmost main dialog is closed.

This widget also handles its own visibility accordingly: It is visible if and only if it has a main dialog to display. It makes itself invisible when there is no more main dialog to display, and it makes itself visible again when a new main dialog appears.

This widget can swallow an arbitrary number of main dialogs as they are opened as long as there is no popup dialog in between.

Definition at line 51 of file [YQMainWinDock.h](#).

### 3.43.2 Constructor & Destructor Documentation

#### 3.43.2.1 YQMainWinDock()

```
YQMainWinDock::YQMainWinDock ( ) [protected]
```

Constructor.

Use the static [mainWinDock\(\)](#) method to access the singleton for this class.

Definition at line 50 of file [YQMainWinDock.cc](#).

Here is the call graph for this function:



### 3.43.3 Member Function Documentation

### 3.43.3.1 add()

```
void YQMainWinDock::add (
    YQDialog * dialog )
```

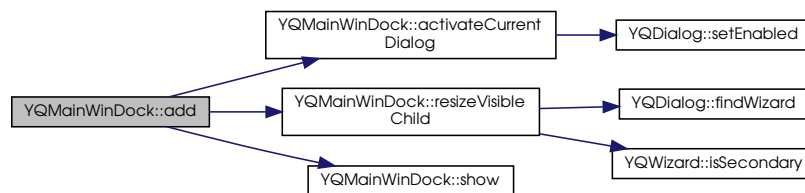
Add a dialog (the widgetRep() of a [YQDialog](#)) to the MainWinDock (on top of its widget stack).

The widget stack does not assume ownership of the widget.

If the MainWinDock is not visible yet, this operation makes it visible.

Definition at line 135 of file [YQMainWinDock.cc](#).

Here is the call graph for this function:



### 3.43.3.2 closeEvent()

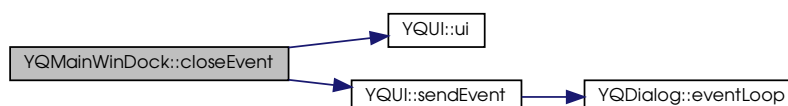
```
void YQMainWinDock::closeEvent (
    QCloseEvent * event ) [virtual]
```

Window manager close event (Alt-F4): Send a `YCancelEvent` and let the application handle that event.

Reimplemented from `QWidget`.

Definition at line 297 of file [YQMainWinDock.cc](#).

Here is the call graph for this function:



### 3.43.3.3 mainWinDock()

```
YQMainWinDock * YQMainWinDock::mainWinDock ( ) [static]
```

Static method to access the singleton for this class.

This creates the (one and only) instance of this class in the first call. Subsequent calls simply return this instance.

Definition at line 39 of file [YQMainWinDock.cc](#).

Here is the call graph for this function:



### 3.43.3.4 paintEvent()

```
void YQMainWinDock::paintEvent (
    QPaintEvent * event ) [protected], [virtual]
```

Paint event.

Reimplemented from QWidget.

Definition at line 310 of file [YQMainWinDock.cc](#).

### 3.43.3.5 remove()

```
void YQMainWinDock::remove (
    YQDialog * dialog = 0 )
```

Remove a dialog from the MainWinDock (if it belongs to the MainWinDock).

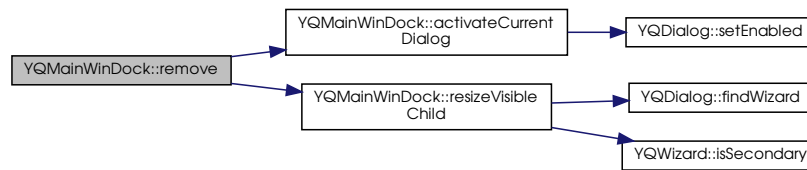
If dialog is 0, this removes the topmost dialog from the MainWinDock.

This can safely be called in the destructor of all dialogs, even those that were never added to the MainWinDock.

If that was the last main dialog in the MainWinDock, the MainWinDock will be hidden (until another main dialog is added).

Definition at line 208 of file [YQMainWinDock.cc](#).

Here is the call graph for this function:



### 3.43.3.6 `resizeEvent()`

```
void YQMainWinDock::resizeEvent (
    QResizeEvent * event ) [protected], [virtual]
```

Resize event.

Reimplemented from `QWidget`.

Definition at line 75 of file [YQMainWinDock.cc](#).

Here is the call graph for this function:



### 3.43.3.7 `show()`

```
void YQMainWinDock::show ( ) [virtual]
```

Show the widget (make it visible).

Reimplemented from `QWidget`.

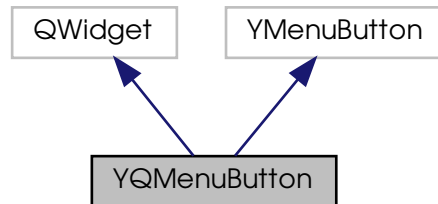
Definition at line 121 of file [YQMainWinDock.cc](#).

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

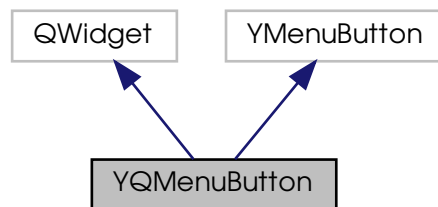
- `/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQMainWinDock.h`
- `/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQMainWinDock.cc`

## 3.44 YQMenuButton Class Reference

Inheritance diagram for YQMenuButton:



Collaboration diagram for YQMenuButton:



### Public Member Functions

- [YQMenuButton](#) (YWidget \*parent, const std::string &label)  
*Constructor.*
- virtual [~YQMenuButton](#) ()  
*Destructor.*
- virtual void [setLabel](#) (const std::string &label)  
*Change the label on the button.*
- virtual void [rebuildMenuTree](#) ()  
*Rebuild the displayed menu tree from the internally stored YMenuItems.*
- virtual void [setEnabled](#) (bool enabled)  
*Set enabled / disabled state.*
- virtual int [preferredWidth](#) ()

- Preferred width of the widget.
- virtual int [preferredHeight](#) ()
- Preferred height of the widget.
- virtual void [setSize](#) (int newWidth, int newHeight)
- Set the new size of the widget.
- virtual bool [setKeyboardFocus](#) ()
- Accept the keyboard focus.

## Protected Slots

- void [menuEntryActivated](#) (QAction \*menuItem)
- Triggered when any menu item is activated.
- void [returnNow](#) ()
- Triggered via [menuEntryActivated\(\)](#) by zero timer to get back in sync with the Qt event loop.

## Protected Member Functions

- void [rebuildMenuTree](#) (QMenu \*menu, YItemIterator begin, YItemIterator end)
- Recursively insert menu items into 'menu' from iterator 'begin' to iterator 'end'.

## Protected Attributes

- YMenuItem \* [\\_selectedItem](#)
- QPushButton \* [\\_qt\\_button](#)
- QHash< QAction \*, int > [\\_serials](#)

### 3.44.1 Detailed Description

Definition at line 37 of file [YQMenuButton.h](#).

### 3.44.2 Member Function Documentation

#### 3.44.2.1 preferredHeight()

```
int YQMenuButton::preferredHeight ( ) [virtual]
```

Preferred height of the widget.

Reimplemented from YWidget.

Definition at line 204 of file [YQMenuButton.cc](#).

### 3.44.2.2 preferredWidth()

```
int YQMenuButton::preferredWidth ( ) [virtual]
```

Preferred width of the widget.

Reimplemented from YWidget.

Definition at line 198 of file [YQMenuButton.cc](#).

### 3.44.2.3 rebuildMenuTree()

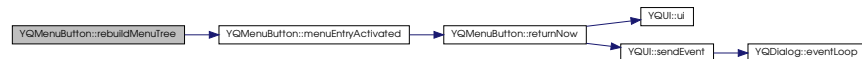
```
void YQMenuButton::rebuildMenuTree ( ) [virtual]
```

Rebuild the displayed menu tree from the internally stored YMenuItems.

Implemented from YMenuButton.

Definition at line 70 of file [YQMenuButton.cc](#).

Here is the call graph for this function:



### 3.44.2.4 setEnabled()

```
void YQMenuButton::setEnabled (
    bool enabled ) [virtual]
```

Set enabled / disabled state.

Reimplemented from YWidget.

Definition at line 191 of file [YQMenuButton.cc](#).

### 3.44.2.5 setLabel()

```
void YQMenuButton::setLabel (
    const std::string & label ) [virtual]
```

Change the label on the button.

Reimplemented from YMenuButton / YSelectionWidget.

Definition at line 62 of file [YQMenuButton.cc](#).

### 3.44.2.6 setSize()

```
void YQMenuButton::setSize (
    int newWidth,
    int newHeight ) [virtual]
```

Set the new size of the widget.

Reimplemented from YWidget.

Definition at line 211 of file [YQMenuButton.cc](#).

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

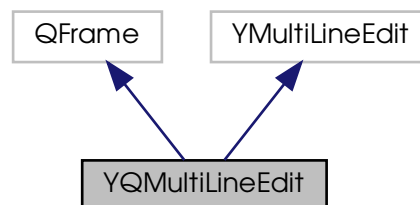
- /usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQMenuButton.h
- /usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQMenuButton.cc

## 3.45 YQMultiLineEdit Class Reference

MultiLineEdit - an input area for multi-line text.

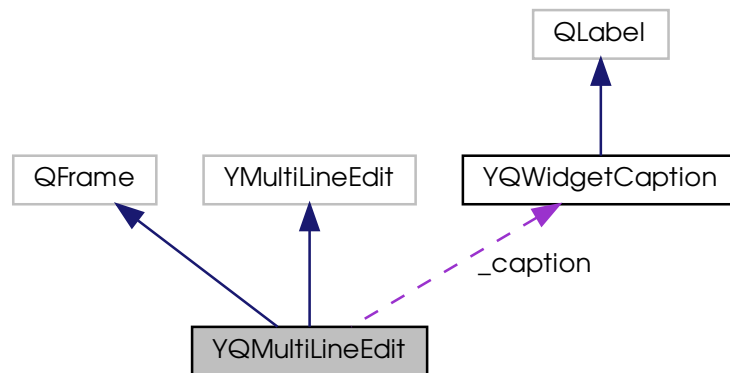
```
#include <YQMultiLineEdit.h>
```

Inheritance diagram for YQMultiLineEdit:





Collaboration diagram for YQMultiLineEdit:



## Public Member Functions

- **YQMultiLineEdit** (YWidget \*parent, const std::string &label)  
*Constructor.*
- virtual **~YQMultiLineEdit** ()  
*Destructor.*
- virtual std::string **value** ()  
*Get the current value (the text entered by the user or set from the outside) of this MultiLineEdit.*
- virtual void **setValue** (const std::string &text)  
*Set the current value (the text entered by the user or set from the outside) of this MultiLineEdit.*
- virtual void **setLabel** (const std::string &label)  
*Set the label (the caption above the MultiLineEdit).*
- virtual void **setInputMaxLength** (int numberOfChars)  
*Set the maximum input length, i.e., the maximum number of characters the user can enter.*
- virtual void **setEnabled** (bool enabled)  
*Set enabled/disabled state.*
- virtual int **preferredWidth** ()  
*Preferred width of the widget.*
- virtual int **preferredHeight** ()  
*Preferred height of the widget.*
- virtual void **setSize** (int newWidth, int newHeight)  
*Set the new size of the widget.*
- virtual bool **setKeyboardFocus** ()  
*Accept the keyboard focus.*

## Protected Slots

- void [changed](#) ()  
*Triggered when the text changes.*
- void [enforceMaxInputLength](#) ()  
*Enforce the maximum input length: If the text becomes too long, remove the just-entered character at the current cursor position.*

## Protected Attributes

- [YQWidgetCaption](#) \* `_caption`
- [QTextEdit](#) \* `_qt_textEdit`

### 3.45.1 Detailed Description

MultiLineEdit - an input area for multi-line text.

Definition at line 39 of file [YQMultiLineEdit.h](#).

### 3.45.2 Member Function Documentation

#### 3.45.2.1 [enforceMaxInputLength](#)

```
void YQMultiLineEdit::enforceMaxInputLength ( ) [protected], [slot]
```

Enforce the maximum input length: If the text becomes too long, remove the just-entered character at the current cursor position.

Note that this is a lot more user friendly than silently truncating at the end of the text: In the latter case, chances are that the user never gets to know that text was truncated. Removing the just-typed character OTOH is something he will notice very quickly.

Definition at line 112 of file [YQMultiLineEdit.cc](#).

#### 3.45.2.2 [preferredHeight\(\)](#)

```
int YQMultiLineEdit::preferredHeight ( ) [virtual]
```

Preferred height of the widget.

Reimplemented from [YWidget](#).

Definition at line 142 of file [YQMultiLineEdit.cc](#).

#### 3.45.2.3 preferredWidth()

```
int YQMultiLineEdit::preferredWidth ( ) [virtual]
```

Preferred width of the widget.

Reimplemented from YWidget.

Definition at line 136 of file [YQMultiLineEdit.cc](#).

#### 3.45.2.4 setEnabled()

```
void YQMultiLineEdit::setEnabled (
    bool enabled ) [virtual]
```

Set enabled/disabled state.

Reimplemented from YWidget.

Definition at line 128 of file [YQMultiLineEdit.cc](#).

#### 3.45.2.5 setInputMaxLength()

```
void YQMultiLineEdit::setInputMaxLength (
    int numberOfChars ) [virtual]
```

Set the maximum input length, i.e., the maximum number of characters the user can enter.

-1 means no limit.

Reimplemented from YMultiLineEdit.

Definition at line 98 of file [YQMultiLineEdit.cc](#).

#### 3.45.2.6 setLabel()

```
void YQMultiLineEdit::setLabel (
    const std::string & label ) [virtual]
```

Set the label (the caption above the MultiLineEdit).

Reimplemented from YMultiLineEdit.

Definition at line 91 of file [YQMultiLineEdit.cc](#).

Here is the call graph for this function:



#### 3.45.2.7 setSize()

```
void YQMultiLineEdit::setSize (
    int newWidth,
    int newHeight ) [virtual]
```

Set the new size of the widget.

Reimplemented from YWidget.

Definition at line 154 of file [YQMultiLineEdit.cc](#).

#### 3.45.2.8 setValue()

```
void YQMultiLineEdit::setValue (
    const std::string & text ) [virtual]
```

Set the current value (the text entered by the user or set from the outside) of this MultiLineEdit.

Implemented from YMultiLineEdit.

Definition at line 83 of file [YQMultiLineEdit.cc](#).

## 3.45.2.9 value()

```
string YQMultiLineEdit::value ( ) [virtual]
```

Get the current value (the text entered by the user or set from the outside) of this MultiLineEdit.

Implemented from YMultiLineEdit.

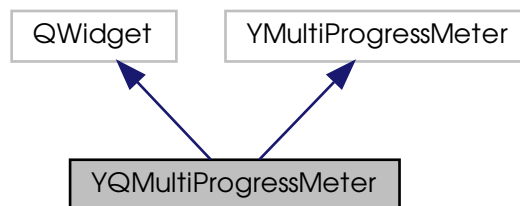
Definition at line 77 of file [YQMultiLineEdit.cc](#).

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

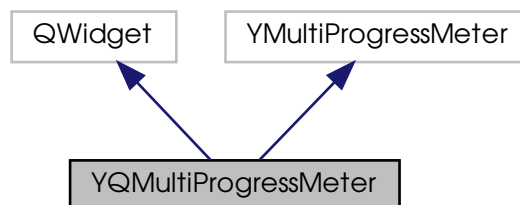
- /usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQMultiLineEdit.h
- /usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQMultiLineEdit.cc

## 3.46 YQMultiProgressMeter Class Reference

Inheritance diagram for YQMultiProgressMeter:



Collaboration diagram for YQMultiProgressMeter:



## Public Member Functions

- [YQMultiProgressMeter](#) (YWidget \*parent, YUIDimension dim, const vector< float > &maxValues)  
*Constructor.*
- virtual [~YQMultiProgressMeter](#) ()  
*Destructor.*
- int [thickness](#) ()  
*Overall thickness (in pixels) of the MultiProgressMeter.*
- int [length](#) ()  
*Overall length (in pixels) of the MultiProgressMeter.*
- int [margin](#) () const  
*Returns the margin around the widget contents.*
- void [setMargin](#) (int value)  
*Sets the margin around the widget contents.*
- int [spacing](#) () const  
*Returns the spacing between segments in pixels.*
- void [setSpacing](#) (int value)  
*Sets the spacing between segments in pixels.*
- int [segmentMinLength](#) () const  
*Returns the minimal length of a segment in pixels.*
- void [setSegmentMinLength](#) (int val)  
*Set the minimal length of a segment in pixels.*
- int [triThickness](#) () const  
*Returns the thickness (base to point) of the small triangles next to the spacing between individual segments.*
- void [setTriThickness](#) (int value)  
*Set the thickness (base to point) of the small triangles next to the spacing between individual segments.*
- int [triSpacing](#) () const  
*Returns the spacing between the segment indicators and the small triangles next to the spacing between segments.*
- void [setTriSpacing](#) (int value)  
*Sets the spacing between the segment indicators and the small triangles next to the spacing between segments.*
- virtual void [setEnabled](#) (bool enabled)  
*Set enabled/disabled state.*
- virtual int [preferredWidth](#) ()  
*Preferred width of the widget.*
- virtual int [preferredHeight](#) ()  
*Preferred height of the widget.*
- virtual void [setSize](#) (int newWidth, int newHeight)  
*Set the new size of the widget.*

## Protected Member Functions

- void [init](#) ()  
*Common initialization.*
- virtual void [doUpdate](#) ()  
*Perform a visual update on the screen.*
- virtual void [paintEvent](#) (QPaintEvent \*)  
*Paint the widget's contents.*

- void [drawSegment](#) (int segment, QPainter &painter, int offset, int [length](#), int [thickness](#))  
*Draw segment number 'segment' with pixel length 'length' from pixel coordinate 'offset' on and fill it according to that segment's current value.*
- void [drawMarkers](#) (QPainter &painter, int offset, int [thickness](#))  
*Draw markers between segments (or beside that spacing).*

### 3.46.1 Detailed Description

Definition at line 42 of file [YQMultiProgressMeter.h](#).

### 3.46.2 Member Function Documentation

#### 3.46.2.1 doUpdate()

```
void YQMultiProgressMeter::doUpdate ( ) [protected], [virtual]
```

Perform a visual update on the screen.

Reimplemented from YMultiProgressMeter.

Definition at line 66 of file [YQMultiProgressMeter.cc](#).

#### 3.46.2.2 drawSegment()

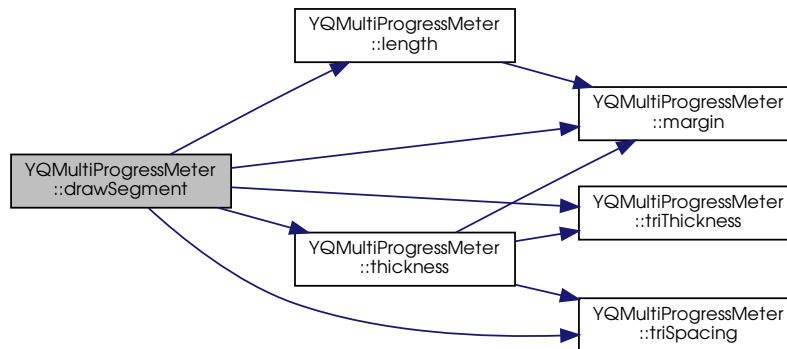
```
void YQMultiProgressMeter::drawSegment (
    int segment,
    QPainter & painter,
    int offset,
    int length,
    int thickness ) [protected]
```

Draw segment number 'segment' with pixel length 'length' from pixel coordinate 'offset' on and fill it according to that segment's current value.

'painter' is set up previously to take rotating into account (horizontal / vertical).

Definition at line 186 of file [YQMultiProgressMeter.cc](#).

Here is the call graph for this function:



### 3.46.2.3 `paintEvent()`

```
void YQMultiProgressMeter::paintEvent (
    QPaintEvent * event ) [protected], [virtual]
```

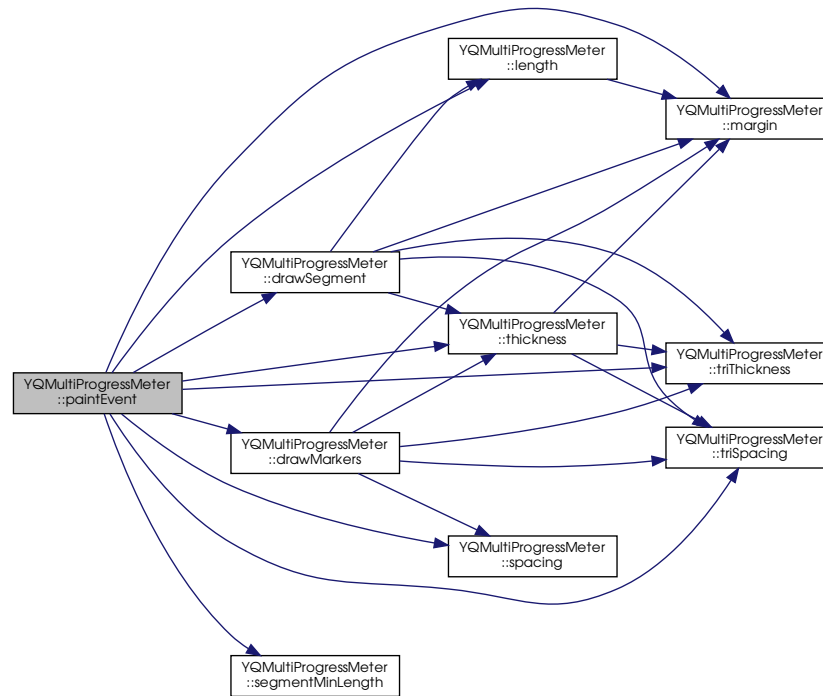
Paint the widget's contents.

Reimplemented from `QWidget`.

Definition at line 72 of file [YQMultiProgressMeter.cc](#).



Here is the call graph for this function:



#### 3.46.2.4 preferredHeight()

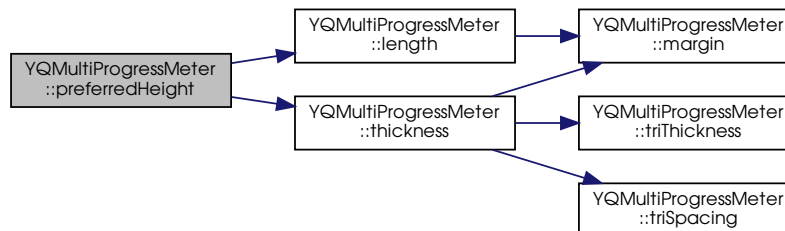
```
int YQMultiProgressMeter::preferredHeight ( ) [virtual]
```

Preferred height of the widget.

Reimplemented from `YWidget`.

Definition at line 329 of file [YQMultiProgressMeter.cc](#).

Here is the call graph for this function:



### 3.46.2.5 preferredWidth()

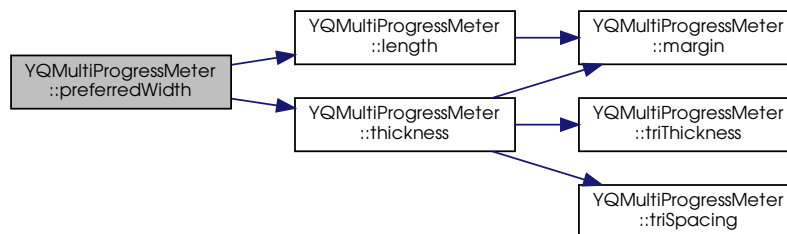
```
int YQMultiProgressMeter::preferredWidth ( ) [virtual]
```

Preferred width of the widget.

Reimplemented from YWidget.

Definition at line 323 of file [YQMultiProgressMeter.cc](#).

Here is the call graph for this function:



### 3.46.2.6 setEnabled()

```
void YQMultiProgressMeter::setEnabled (
    bool enabled ) [virtual]
```

Set enabled/disabled state.

Reimplemented from YWidget.

Definition at line 315 of file [YQMultiProgressMeter.cc](#).

### 3.46.2.7 setMargin()

```
void YQMultiProgressMeter::setMargin (
    int value ) [inline]
```

Sets the margin around the widget contents.

Does not trigger an update.

Definition at line 78 of file [YQMultiProgressMeter.h](#).

#### 3.46.2.8 setSize()

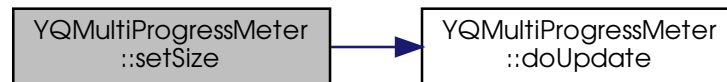
```
void YQMultiProgressMeter::setSize (
    int newWidth,
    int newHeight ) [virtual]
```

Set the new size of the widget.

Reimplemented from YWidget.

Definition at line [335](#) of file [YQMultiProgressMeter.cc](#).

Here is the call graph for this function:



#### 3.46.2.9 setSpacing()

```
void YQMultiProgressMeter::setSpacing (
    int value ) [inline]
```

Sets the spacing between segments in pixels.

Does not trigger an update.

Definition at line [89](#) of file [YQMultiProgressMeter.h](#).

#### 3.46.2.10 setTriThickness()

```
void YQMultiProgressMeter::setTriThickness (
    int value )
```

Set the thickness (base to point) of the small triangles next to the spacing between individual segments.

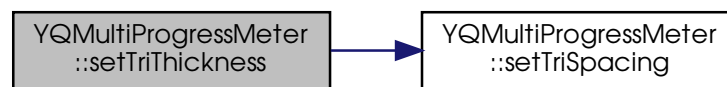
The base length of those small triangles is double this value minus one.

Use -1 to switch off those triangles - in which case triSpacing (see below) will be set to 0.

This call does not trigger a screen update.

Definition at line 306 of file [YQMultiProgressMeter.cc](#).

Here is the call graph for this function:



#### 3.46.2.11 triThickness()

```
int YQMultiProgressMeter::triThickness ( ) const [inline]
```

Returns the thickness (base to point) of the small triangles next to the spacing between individual segments.

The base length of those small triangles is double this value minus one. A value  $<1$  indicates no such triangles will be drawn.

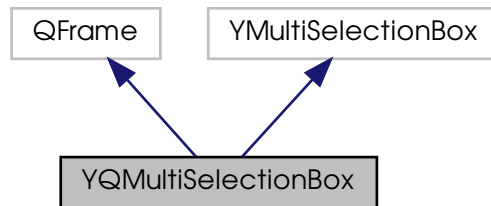
Definition at line 107 of file [YQMultiProgressMeter.h](#).

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

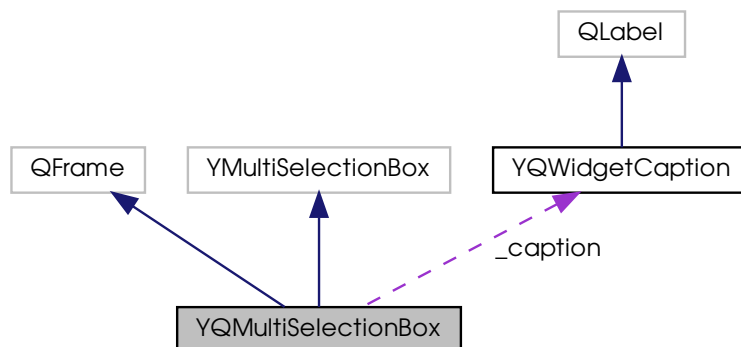
- `/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQMultiProgressMeter.h`
- `/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQMultiProgressMeter.cc`

## 3.47 YQMultiSelectionBox Class Reference

Inheritance diagram for YQMultiSelectionBox:



Collaboration diagram for YQMultiSelectionBox:



### Signals

- void [valueChanged](#) ()  
*Emitted upon when an item changes its state.*

### Public Member Functions

- [YQMultiSelectionBox](#) (YWidget \*parent, const std::string &label)  
*Constructor.*

- [~YQMultiSelectionBox](#) ()  
*Destructor.*
- virtual void [setLabel](#) (const std::string &label)  
*Change the label text.*
- virtual void [addItem](#) (YItem \*item)  
*Add an item.*
- virtual void [selectItem](#) (YItem \*item, bool selected=true)  
*Select or deselect an item.*
- virtual void [deselectAllItems](#) ()  
*Deselect all items.*
- virtual void [deleteAllItems](#) ()  
*Delete all items.*
- virtual YItem \* [currentItem](#) ()  
*Return the the item that currently has the keyboard focus or 0 if no item currently has the keyboard focus.*
- virtual void [setCurrentItem](#) (YItem \*item)  
*Set the keyboard focus to the specified item.*
- virtual void [setEnabled](#) (bool enabled)  
*Set enabled/disabled state.*
- virtual int [preferredWidth](#) ()  
*Preferred width of the widget.*
- virtual int [preferredHeight](#) ()  
*Preferred height of the widget.*
- virtual void [setSize](#) (int newWidth, int newHeight)  
*Set the new size of the widget.*
- virtual bool [setKeyboardFocus](#) ()  
*Accept the keyboard focus.*

## Protected Slots

- void [slotSelected](#) ()  
*Send SelectionChanged event if `opt` (notify) is set.*
- void [slotValueChanged](#) ()  
*Send ValueChanged event if `opt` (notify) is set.*
- void [slotItemChanged](#) (QTreeWidgetItem \*, int)

## Protected Member Functions

- [YQMultiSelectionBoxItem](#) \* [findItem](#) (YItem \*item)  
*Find the counterpart of 'item'.*

## Protected Attributes

- [YQWidgetCaption](#) \* [\\_caption](#)
- QTreeWidgetItem \* [\\_qt\\_listView](#)

### 3.47.1 Detailed Description

Definition at line 38 of file [YQMultiSelectionBox.h](#).

### 3.47.2 Member Function Documentation

#### 3.47.2.1 addItem()

```
void YQMultiSelectionBox::addItem (
    YItem * item ) [virtual]
```

Add an item.

Reimplemented from YSelectionWidget.

Definition at line 104 of file [YQMultiSelectionBox.cc](#).

#### 3.47.2.2 currentItem()

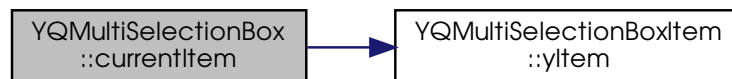
```
YItem * YQMultiSelectionBox::currentItem ( ) [virtual]
```

Return the the item that currently has the keyboard focus or 0 if no item currently has the keyboard focus.

Reimplemented from YMultiSelectionBox.

Definition at line 165 of file [YQMultiSelectionBox.cc](#).

Here is the call graph for this function:



### 3.47.2.3 deleteAllItems()

```
void YQMultiSelectionBox::deleteAllItems ( ) [virtual]
```

Delete all items.

Reimplemented from YSelectionWidget.

Definition at line 155 of file [YQMultiSelectionBox.cc](#).

### 3.47.2.4 deselectAllItems()

```
void YQMultiSelectionBox::deselectAllItems ( ) [virtual]
```

Deselect all items.

Reimplemented from YSelectionWidget.

Definition at line 135 of file [YQMultiSelectionBox.cc](#).

### 3.47.2.5 findItem()

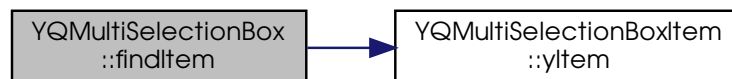
```
YQMultiSelectionBoxItem * YQMultiSelectionBox::findItem (
    YItem * item ) [protected]
```

Find the counterpart of 'item'.

Return 0 if there is none.

Definition at line 310 of file [YQMultiSelectionBox.cc](#).

Here is the call graph for this function:





### 3.47.2.6 preferredHeight()

```
int YQMultiSelectionBox::preferredHeight ( ) [virtual]
```

Preferred height of the widget.

Reimplemented from YWidget.

Definition at line 250 of file [YQMultiSelectionBox.cc](#).

### 3.47.2.7 preferredWidth()

```
int YQMultiSelectionBox::preferredWidth ( ) [virtual]
```

Preferred width of the widget.

Reimplemented from YWidget.

Definition at line 241 of file [YQMultiSelectionBox.cc](#).

### 3.47.2.8 selectItem()

```
void YQMultiSelectionBox::selectItem (
    YItem * item,
    bool selected = true ) [virtual]
```

Select or deselect an item.

Reimplemented from YSelectionWidget.

Definition at line 124 of file [YQMultiSelectionBox.cc](#).

Here is the call graph for this function:



### 3.47.2.9 setCurrentItem()

```
void YQMultiSelectionBox::setCurrentItem (
    YItem * item ) [virtual]
```

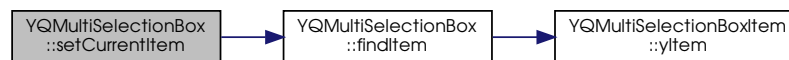
Set the keyboard focus to the specified item.

0 means clear the keyboard focus.

Reimplemented from YMultiSelectionBox.

Definition at line [204](#) of file [YQMultiSelectionBox.cc](#).

Here is the call graph for this function:



### 3.47.2.10 setEnabled()

```
void YQMultiSelectionBox::setEnabled (
    bool enabled ) [virtual]
```

Set enabled/disabled state.

Reimplemented from YWidget.

Definition at line [232](#) of file [YQMultiSelectionBox.cc](#).

#### 3.47.2.11 setLabel()

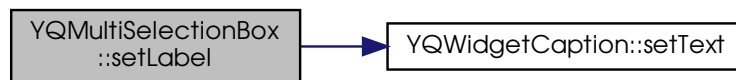
```
void YQMultiSelectionBox::setLabel (
    const std::string & label ) [virtual]
```

Change the label text.

Reimplemented from YSelectionWidget.

Definition at line 96 of file [YQMultiSelectionBox.cc](#).

Here is the call graph for this function:



#### 3.47.2.12 setSize()

```
void YQMultiSelectionBox::setSize (
    int newWidth,
    int newHeight ) [virtual]
```

Set the new size of the widget.

Reimplemented from YWidget.

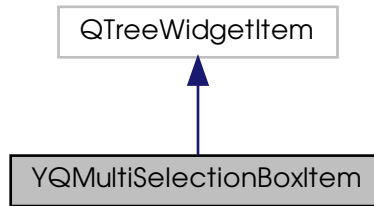
Definition at line 262 of file [YQMultiSelectionBox.cc](#).

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

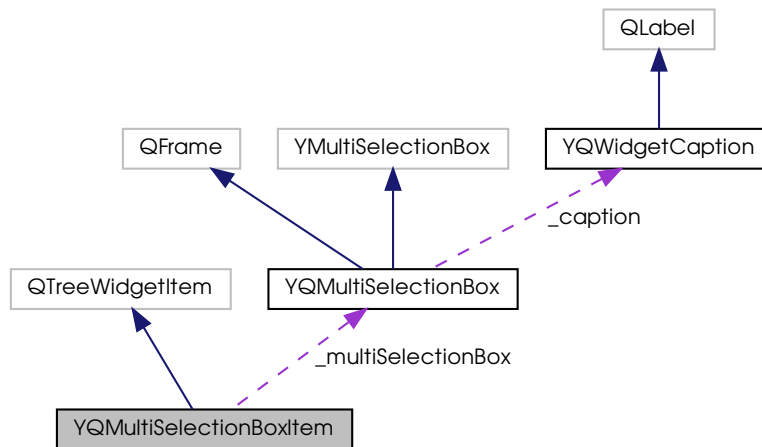
- `/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQMultiSelectionBox.h`
- `/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQMultiSelectionBox.cc`

### 3.48 YQMultiSelectionBoxItem Class Reference

Inheritance diagram for YQMultiSelectionBoxItem:



Collaboration diagram for YQMultiSelectionBoxItem:



#### Public Member Functions

- [YQMultiSelectionBoxItem](#) ([YQMultiSelectionBox](#) \*parent, `QTreeWidgetItem` \*listView, `YItem` \*yItem)  
*Constructor.*
- virtual [~YQMultiSelectionBoxItem](#) ()  
*Destructor.*
- `YItem` \* [yItem](#) () const  
*Return the corresponding YItem.*

### Protected Attributes

- YItem \* **\_yItem**
- int **\_serial**
- [YQMultiSelectionBox](#) \* **\_multiSelectionBox**

### Static Protected Attributes

- static int **\_item\_count** = 0  
*The next serial number to use.*

#### 3.48.1 Detailed Description

Definition at line 172 of file [YQMultiSelectionBox.h](#).

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

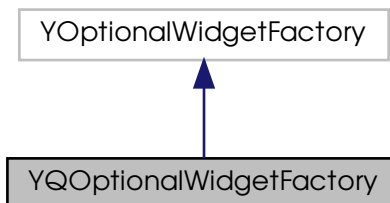
- /usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQMultiSelectionBox.h
- /usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQMultiSelectionBox.cc

## 3.49 YQOptionalWidgetFactory Class Reference

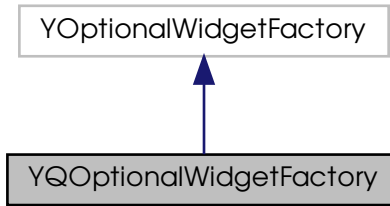
Widget factory for optional ("special") widgets.

```
#include <YQOptionalWidgetFactory.h>
```

Inheritance diagram for YQOptionalWidgetFactory:



Collaboration diagram for YQOptionalWidgetFactory:



### Public Member Functions

- virtual bool **hasWizard** ()
- virtual YWizard \* **createWizard** (YWidget \*parent, const std::string &backButtonLabel, const std::string &abortButtonLabel, const std::string &nextButtonLabel, YWizardMode wizardMode=YWizardMode\_Standard)
- virtual bool **hasDumbTab** ()
- virtual YQDumbTab \* **createDumbTab** (YWidget \*parent)
- virtual bool **hasSlider** ()
- virtual YQSlider \* **createSlider** (YWidget \*parent, const std::string &label, int minVal, int maxVal, int initialVal)
- virtual bool **hasDateField** ()
- virtual YQDateField \* **createDateField** (YWidget \*parent, const std::string &label)
- virtual bool **hasTimeField** ()
- virtual YQTimeField \* **createTimeField** (YWidget \*parent, const std::string &label)
- virtual bool **hasBarGraph** ()
- virtual YQBarGraph \* **createBarGraph** (YWidget \*parent)
- virtual bool **hasPatternSelector** ()
- virtual YWidget \* **createPatternSelector** (YWidget \*parent, long modeFlags=0)
- virtual bool **hasSimplePatchSelector** ()
- virtual YWidget \* **createSimplePatchSelector** (YWidget \*parent, long modeFlags=0)
- virtual bool **hasMultiProgressMeter** ()
- virtual YQMultiProgressMeter \* **createMultiProgressMeter** (YWidget \*parent, YUIDimension dim, const vector< float > &maxValues)
- virtual bool **hasPartitionSplitter** ()
- virtual YQPartitionSplitter \* **createPartitionSplitter** (YWidget \*parent, int usedSize, int totalFreeSize, int newPartSize, int minNewPartSize, int minFreeSize, const std::string &usedLabel, const std::string &freeLabel, const std::string &newPartLabel, const std::string &freeFieldLabel, const std::string &newPartFieldLabel)
- virtual bool **hasDownloadProgress** ()
- virtual YQDownloadProgress \* **createDownloadProgress** (YWidget \*parent, const std::string &label, const std::string &filename, YFileSize\_t expectedFileSize)
- virtual bool **hasTimezoneSelector** ()
- virtual YTimezoneSelector \* **createTimezoneSelector** (YWidget \*parent, const std::string &pixmap, const map< string, string > &timezones)
- virtual bool **hasGraph** ()
- virtual YGraph \* **createGraph** (YWidget \*parent, const std::string &filename, const std::string &layoutAlgorithm)
- virtual YGraph \* **createGraph** (YWidget \*parent, void \*graph)
- virtual bool **hasContextMenu** ()

## Protected Member Functions

- [YQOptionalWidgetFactory](#) ()  
*Constructor.*
- virtual [~YQOptionalWidgetFactory](#) ()  
*Destructor.*

## Friends

- class **YQUI**

### 3.49.1 Detailed Description

Widget factory for optional ("special") widgets.

Remember to always check with the corresponding "has..()" method if the current UI actually provides the requested widget. Otherwise the "create...()" method will throw an exception.

Definition at line 54 of file [YQOptionalWidgetFactory.h](#).

### 3.49.2 Constructor & Destructor Documentation

#### 3.49.2.1 YQOptionalWidgetFactory()

```
YQOptionalWidgetFactory::YQOptionalWidgetFactory ( ) [protected]
```

Constructor.

Use `YUI::optionalWidgetFactory()` to get the singleton for this class.

Definition at line 39 of file [YQOptionalWidgetFactory.cc](#).

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

- `/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQOptionalWidgetFactory.h`
- `/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQOptionalWidgetFactory.cc`

## 3.50 YQPackageSelectorPluginIf Class Reference

### Public Member Functions

- virtual `YPackageSelector * createPackageSelector (YWidget *parent, long modeFlags)=0`
- virtual `YWidget * createPatternSelector (YWidget *parent, long modeFlags)=0`
- virtual `YWidget * createSimplePatchSelector (YWidget *parent, long modeFlags)=0`

### 3.50.1 Detailed Description

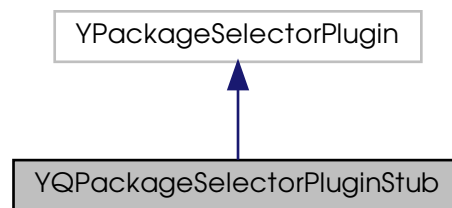
Definition at line 33 of file [YQPackageSelectorPluginIf.h](#).

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

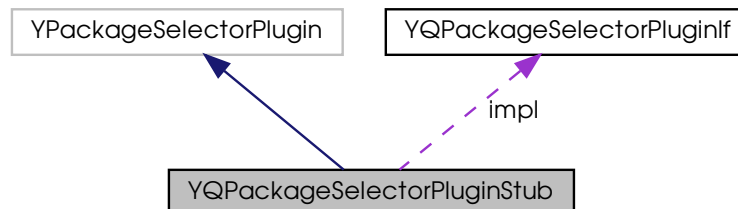
- /usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQPackageSelectorPluginIf.h

## 3.51 YQPackageSelectorPluginStub Class Reference

Inheritance diagram for YQPackageSelectorPluginStub:



Collaboration diagram for YQPackageSelectorPluginStub:





## Public Member Functions

- [YQPackageSelectorPluginStub](#) ()  
*Constructor: Load the plugin library for the package selector.*
- virtual [~YQPackageSelectorPluginStub](#) ()  
*Destructor.*
- virtual YPackageSelector \* [createPackageSelector](#) (YWidget \*parent, long modeFlags)  
*Create a package selector.*
- virtual YWidget \* [createPatternSelector](#) (YWidget \*parent, long modeFlags)  
*Create a pattern selector (optional widget).*
- virtual YWidget \* [createSimplePatchSelector](#) (YWidget \*parent, long modeFlags)  
*Create a simple patch selector (optional widget).*

## Public Attributes

- [YQPackageSelectorPluginIf](#) \* **impl**

### 3.51.1 Detailed Description

Definition at line 43 of file [YQPackageSelectorPluginStub.h](#).

### 3.51.2 Member Function Documentation

#### 3.51.2.1 createPackageSelector()

```
YPackageSelector * YQPackageSelectorPluginStub::createPackageSelector (
    YWidget * parent,
    long modeFlags ) [virtual]
```

Create a package selector.

Implemented from YPackageSelectorPlugin.

This might return 0 if the plugin lib could not be loaded or if the appropriate symbol could not be located in the plugin lib.

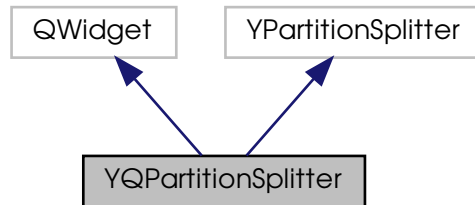
Definition at line 61 of file [YQPackageSelectorPluginStub.cc](#).

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

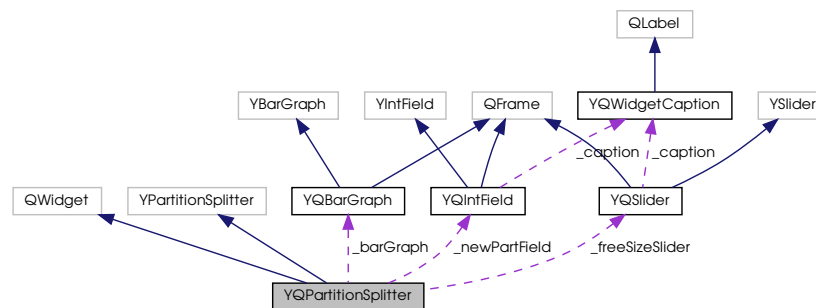
- /usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQPackageSelectorPluginStub.h
- /usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQPackageSelectorPluginStub.cc

### 3.52 YQPartitionSplitter Class Reference

Inheritance diagram for YQPartitionSplitter:



Collaboration diagram for YQPartitionSplitter:



#### Public Slots

- void [setFreeSizeSlot](#) (int newFreeSize)  
*Slot for setting the free size.*
- void [setNewPartSizeSlot](#) (int newNewSize)  
*Slot for setting the new size.*

#### Public Member Functions

- [YQPartitionSplitter](#) (YWidget \*parent, int usedSize, int freeSize, int newPartSize, int minNewPartSize, int minFreeSize, const std::string &usedLabel, const std::string &freeLabel, const std::string &newPartLabel, const std::string &freeFieldLabel, const std::string &newPartFieldLabel)  
*Constructor.*

- [~YQPartitionSplitter](#) ()  
*Destructor.*
- virtual int [value](#) ()  
*Return the value (the size of the new partition).*
- virtual void [setValue](#) (int newValue)  
*Set the value (the size of the new partition).*
- virtual void [setEnabled](#) (bool enabled)  
*Set enabled/disabled state.*
- virtual int [preferredWidth](#) ()  
*Preferred width of the widget.*
- virtual int [preferredHeight](#) ()  
*Preferred height of the widget.*
- virtual void [setSize](#) (int newWidth, int newHeight)  
*Set the new size of the widget.*
- virtual bool [setKeyboardFocus](#) ()  
*Accept the keyboard focus.*

### Protected Types

- enum { **usedSegment** = 0, **freeSegment** = 1, **newPartSegment** = 2 }

### Protected Attributes

- YLayoutBox \* **\_vbox**
- [YQBarGraph](#) \* **\_barGraph**
- YLayoutBox \* **\_hbox**
- [YQSlider](#) \* **\_freeSizeSlider**
- [YQIntField](#) \* **\_newPartField**

### 3.52.1 Detailed Description

Definition at line 38 of file [YQPartitionSplitter.h](#).

### 3.52.2 Member Function Documentation

#### 3.52.2.1 preferredHeight()

```
int YQPartitionSplitter::preferredHeight ( ) [virtual]
```

Preferred height of the widget.

Reimplemented from YWidget.

Definition at line 138 of file [YQPartitionSplitter.cc](#).

### 3.52.2.2 preferredWidth()

```
int YQPartitionSplitter::preferredWidth ( ) [virtual]
```

Preferred width of the widget.

Reimplemented from YWidget.

Definition at line 132 of file [YQPartitionSplitter.cc](#).

### 3.52.2.3 setEnabled()

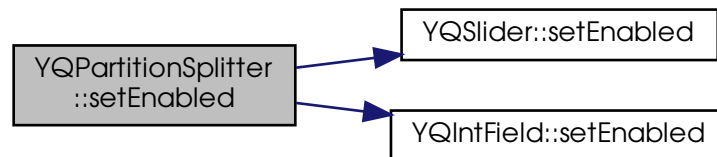
```
void YQPartitionSplitter::setEnabled (
    bool enabled ) [virtual]
```

Set enabled/disabled state.

Reimplemented from YWidget.

Definition at line 123 of file [YQPartitionSplitter.cc](#).

Here is the call graph for this function:



### 3.52.2.4 setSize()

```
void YQPartitionSplitter::setSize (
    int newWidth,
    int newHeight ) [virtual]
```

Set the new size of the widget.

Reimplemented from YWidget.

Definition at line 144 of file [YQPartitionSplitter.cc](#).

### 3.52.2.5 setValue()

```
void YQPartitionSplitter::setValue (
    int newValue ) [virtual]
```

Set the value (the size of the new partition).

Implemented from YPartitionSplitter.

Definition at line 159 of file [YQPartitionSplitter.cc](#).

### 3.52.2.6 value()

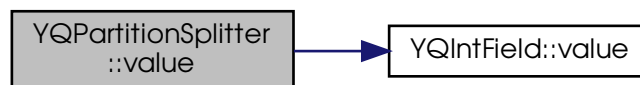
```
int YQPartitionSplitter::value ( ) [virtual]
```

Return the value (the size of the new partition).

Implemented from YPartitionSplitter.

Definition at line 151 of file [YQPartitionSplitter.cc](#).

Here is the call graph for this function:

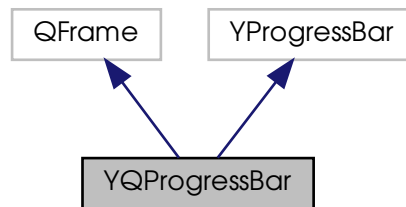


The documentation for this class was generated from the following files:

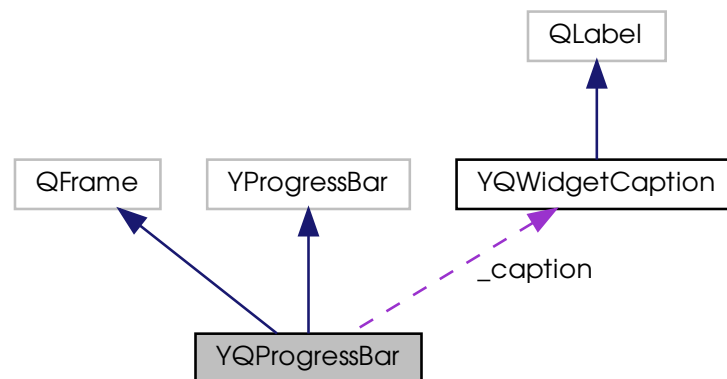
- `/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQPartitionSplitter.h`
- `/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQPartitionSplitter.cc`

### 3.53 YQProgressBar Class Reference

Inheritance diagram for YQProgressBar:



Collaboration diagram for YQProgressBar:



#### Public Member Functions

- [YQProgressBar](#) (YWidget \*parent, const std::string &label, int maxValue=100)  
*Constructor.*
- virtual [~YQProgressBar](#) ()  
*Destructor.*
- virtual void [setLabel](#) (const std::string &label)  
*Set the label (the caption above the progress bar).*
- virtual void [setValue](#) (int newValue)

- Set the current progress value (  $\leq \text{maxValue}()$  ).*
  - virtual void [setEnabled](#) (bool enabled)  
*Set enabled/disabled state.*
  - virtual int [preferredWidth](#) ()  
*Preferred width of the widget.*
  - virtual int [preferredHeight](#) ()  
*Preferred height of the widget.*
  - virtual void [setSize](#) (int newWidth, int newHeight)  
*Set the new size of the widget.*
  - virtual bool [setKeyboardFocus](#) ()  
*Accept the keyboard focus.*

### Protected Attributes

- [YQWidgetCaption](#) \* **\_caption**
- [QProgressBar](#) \* **\_qt\_progressbar**

### 3.53.1 Detailed Description

Definition at line 37 of file [YQProgressBar.h](#).

### 3.53.2 Member Function Documentation

#### 3.53.2.1 preferredHeight()

```
int YQProgressBar::preferredHeight ( ) [virtual]
```

Preferred height of the widget.

Reimplemented from [YWidget](#).

Definition at line 106 of file [YQProgressBar.cc](#).

#### 3.53.2.2 preferredWidth()

```
int YQProgressBar::preferredWidth ( ) [virtual]
```

Preferred width of the widget.

Reimplemented from [YWidget](#).

Definition at line 97 of file [YQProgressBar.cc](#).

### 3.53.2.3 `setEnabled()`

```
void YQProgressBar::setEnabled (
    bool enabled ) [virtual]
```

Set enabled/disabled state.

Reimplemented from `YWidget`.

Definition at line 89 of file [YQProgressBar.cc](#).

### 3.53.2.4 `setLabel()`

```
void YQProgressBar::setLabel (
    const std::string & label ) [virtual]
```

Set the label (the caption above the progress bar).

Reimplemented from `YProgressBar`.

Definition at line 74 of file [YQProgressBar.cc](#).

Here is the call graph for this function:



### 3.53.2.5 `setSize()`

```
void YQProgressBar::setSize (
    int newWidth,
    int newHeight ) [virtual]
```

Set the new size of the widget.

Reimplemented from `YWidget`.

Definition at line 112 of file [YQProgressBar.cc](#).



## 3.53.2.6 setValue()

```
void YQProgressBar::setValue (
    int newValue ) [virtual]
```

Set the current progress value (  $\leq$  `maxValue()` ).

Reimplemented from `YProgressBar`.

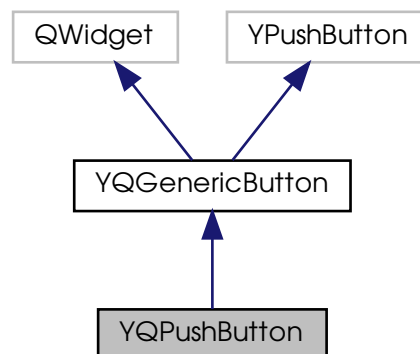
Definition at line 81 of file [YQProgressBar.cc](#).

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

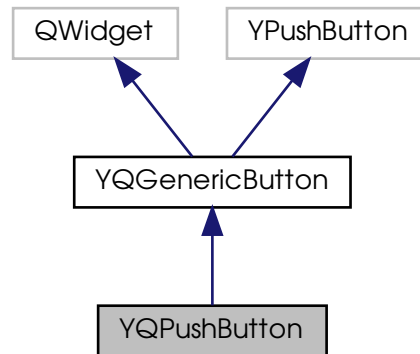
- `/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQProgressBar.h`
- `/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQProgressBar.cc`

## 3.54 YQPushButton Class Reference

Inheritance diagram for `YQPushButton`:



Collaboration diagram for YQPushButton:



## Public Slots

- void [hit](#) ()  
*Triggered when the button is activated (clicked or via keyboard).*

## Public Member Functions

- [YQPushButton](#) (QWidget \*parent, const std::string &label)  
*Constructor.*
- virtual [~YQPushButton](#) ()  
*Destructor.*
- virtual int [preferredWidth](#) ()  
*Preferred width of the widget.*
- virtual int [preferredHeight](#) ()  
*Preferred height of the widget.*
- virtual void [setSize](#) (int newWidth, int newHeight)  
*Set the new size of the widget.*

## Additional Inherited Members

### 3.54.1 Detailed Description

Definition at line 34 of file [YQPushButton.h](#).

### 3.54.2 Member Function Documentation

#### 3.54.2.1 hit

```
void YQPushButton::hit ( ) [slot]
```

Triggered when the button is activated (clicked or via keyboard).

Notifies the the ui about this.

Definition at line 83 of file [YQPushButton.cc](#).

Here is the call graph for this function:



#### 3.54.2.2 preferredHeight()

```
int YQPushButton::preferredHeight ( ) [virtual]
```

Preferred height of the widget.

Reimplemented from `YWidget`.

Definition at line 69 of file [YQPushButton.cc](#).

Here is the call graph for this function:



### 3.54.2.3 preferredWidth()

```
int YQPushButton::preferredWidth ( ) [virtual]
```

Preferred width of the widget.

Reimplemented from YWidget.

Definition at line 63 of file [YQPushButton.cc](#).

Here is the call graph for this function:



### 3.54.2.4 setSize()

```
void YQPushButton::setSize (
    int newWidth,
    int newHeight ) [virtual]
```

Set the new size of the widget.

Reimplemented from YWidget.

Definition at line 75 of file [YQPushButton.cc](#).

Here is the call graph for this function:

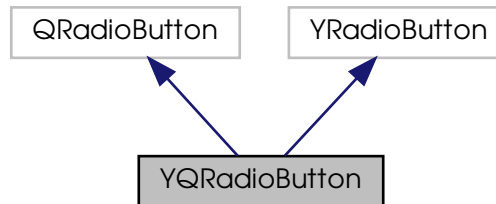


The documentation for this class was generated from the following files:

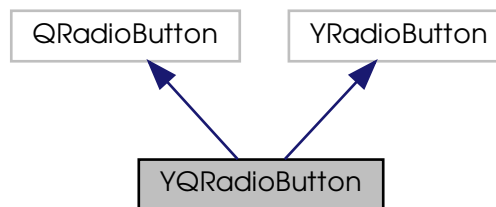
- `/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQPushButton.h`
- `/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQPushButton.cc`

## 3.55 YQRadioButton Class Reference

Inheritance diagram for YQRadioButton:



Collaboration diagram for YQRadioButton:



### Public Member Functions

- **YQRadioButton** (YWidget \*parent, const std::string &label, bool checked)  
*Constructor.*
- virtual bool **value** ()  
*Return the "checked" state of the RadioButton.*
- virtual void **setValue** (bool checked)  
*Set the "checked" state of the RadioButton.*
- virtual void **setLabel** (const std::string &label)  
*Change the label (the text) on the RadioButton.*
- virtual void **setUseBoldFont** (bool bold=true)  
*Use a bold font.*
- virtual void **setEnabled** (bool enabled)

- Set enabled / disabled state.*
- virtual int [preferredWidth](#) ()  
*Preferred width of the widget.*
- virtual int [preferredHeight](#) ()  
*Preferred height of the widget.*
- virtual void [setSize](#) (int newWidth, int newHeight)  
*Set the new size of the widget.*
- virtual bool [setKeyboardFocus](#) ()  
*Accept the keyboard focus.*

### Protected Slots

- void [changed](#) (bool newState)  
*Triggered when the RadioButton is toggled.*
- bool [eventFilter](#) (QObject \*obj, QEvent \*event)  
*Redirect events to this object.*

### 3.55.1 Detailed Description

Definition at line [35](#) of file [YQRadioButton.h](#).

### 3.55.2 Member Function Documentation

#### 3.55.2.1 [preferredHeight\(\)](#)

```
int YQRadioButton::preferredHeight ( ) [virtual]
```

Preferred height of the widget.

Reimplemented from YWidget.

Definition at line [91](#) of file [YQRadioButton.cc](#).

#### 3.55.2.2 [preferredWidth\(\)](#)

```
int YQRadioButton::preferredWidth ( ) [virtual]
```

Preferred width of the widget.

Reimplemented from YWidget.

Definition at line [85](#) of file [YQRadioButton.cc](#).

### 3.55.2.3 `setEnabled()`

```
void YQRadioButton::setEnabled (
    bool enabled ) [virtual]
```

Set enabled / disabled state.

Reimplemented from YWidget.

Definition at line 133 of file [YQRadioButton.cc](#).

### 3.55.2.4 `setKeyboardFocus()`

```
bool YQRadioButton::setKeyboardFocus ( ) [virtual]
```

Accept the keyboard focus.

Reimplemented from YWidget.

Definition at line 140 of file [YQRadioButton.cc](#).

### 3.55.2.5 `setLabel()`

```
void YQRadioButton::setLabel (
    const std::string & label ) [virtual]
```

Change the label (the text) on the RadioButton.

Reimplemented from YRadioButton.

Definition at line 126 of file [YQRadioButton.cc](#).

### 3.55.2.6 `setSize()`

```
void YQRadioButton::setSize (
    int newWidth,
    int newHeight ) [virtual]
```

Set the new size of the widget.

Reimplemented from YWidget.

Definition at line 97 of file [YQRadioButton.cc](#).

### 3.55.2.7 `setUseBoldFont()`

```
void YQRadioButton::setUseBoldFont (
    bool bold = true ) [virtual]
```

Use a bold font.

Reimplemented from YRadioButton.

Definition at line 75 of file [YQRadioButton.cc](#).

Here is the call graph for this function:



### 3.55.2.8 `setValue()`

```
void YQRadioButton::setValue (
    bool checked ) [virtual]
```

Set the "checked" state of the RadioButton.

Reimplemented from YRadioButton.

Definition at line 109 of file [YQRadioButton.cc](#).

### 3.55.2.9 `value()`

```
bool YQRadioButton::value ( ) [virtual]
```

Return the "checked" state of the RadioButton.

Reimplemented from YRadioButton.

Definition at line 103 of file [YQRadioButton.cc](#).

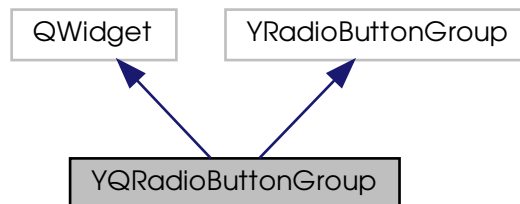
The documentation for this class was generated from the following files:

- `/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQRadioButton.h`
- `/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQRadioButton.cc`

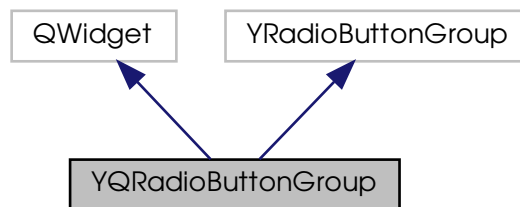


## 3.56 YQRadioButtonGroup Class Reference

Inheritance diagram for YQRadioButtonGroup:



Collaboration diagram for YQRadioButtonGroup:



### Public Member Functions

- [YQRadioButtonGroup](#) (YWidget \*parent)  
*Constructor.*
- virtual [~YQRadioButtonGroup](#) ()  
*Destructor.*
- virtual void [addRadioButton](#) (YRadioButton \*button)  
*Add a RadioButton.*
- virtual void [setEnabled](#) (bool enabled)  
*Set enabled/disabled state.*
- virtual void [setSize](#) (int newWidth, int newHeight)  
*Set the new size of the widget.*

### 3.56.1 Detailed Description

Definition at line 32 of file [YQRadioButtonGroup.h](#).

### 3.56.2 Member Function Documentation

#### 3.56.2.1 `addRadioButton()`

```
void YQRadioButtonGroup::addRadioButton (
    YRadioButton * button ) [virtual]
```

Add a RadioButton.

Reimplemented from YRadioButtonGroup.

Definition at line 50 of file [YQRadioButtonGroup.cc](#).

#### 3.56.2.2 `setEnabled()`

```
void YQRadioButtonGroup::setEnabled (
    bool enabled ) [virtual]
```

Set enabled/disabled state.

Reimplemented from YWidget.

Definition at line 62 of file [YQRadioButtonGroup.cc](#).

#### 3.56.2.3 `setSize()`

```
void YQRadioButtonGroup::setSize (
    int newWidth,
    int newHeight ) [virtual]
```

Set the new size of the widget.

Reimplemented from YWidget.

Definition at line 70 of file [YQRadioButtonGroup.cc](#).

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

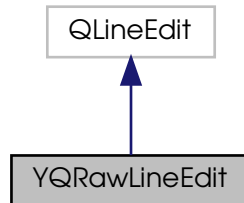
- `/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQRadioButtonGroup.h`
- `/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQRadioButtonGroup.cc`

## 3.57 YQRawLineEdit Class Reference

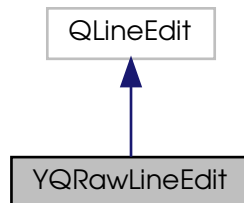
Helper class that can obtain the CapsLock status, too.

```
#include <YQInputField.h>
```

Inheritance diagram for YQRawLineEdit:



Collaboration diagram for YQRawLineEdit:



### Signals

- void **capsLockActivated** ()
- void **capsLockDeactivated** ()

### Public Member Functions

- [YQRawLineEdit](#) (QWidget \*parent)  
*Constructor.*
- virtual [~YQRawLineEdit](#) ()  
*Destructor.*
- bool [isCapsLockActive](#) () const  
*Check if CapsLock is active (rather: was active at the time of the last key or focus event)*

## Protected Member Functions

- `bool x11Event (XEvent *event)`

*X11 raw event handler.*

### 3.57.1 Detailed Description

Helper class that can obtain the CapsLock status, too.

For some reason, Qt does not propagate that information from X11.

Definition at line 167 of file [YQInputField.h](#).

### 3.57.2 Member Function Documentation

#### 3.57.2.1 x11Event()

```
bool YQRawLineEdit::x11Event (
    XEvent * event ) [protected]
```

X11 raw event handler.

Propagates all events to the Qt event handlers, but updates `_capsLockActive` for key events.

Reimplemented from `QWidget`.

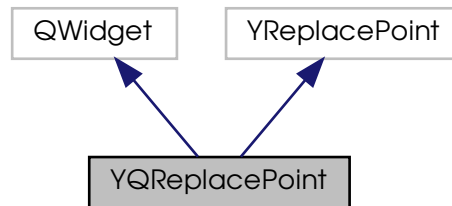
Definition at line 257 of file [YQInputField.cc](#).

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

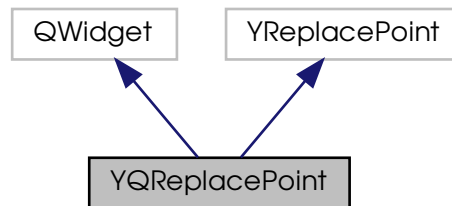
- `/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQInputField.h`
- `/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQInputField.cc`

## 3.58 YQReplacePoint Class Reference

Inheritance diagram for YQReplacePoint:



Collaboration diagram for YQReplacePoint:



### Public Member Functions

- [YQReplacePoint](#) (YWidget \*parent)  
*Constructor.*
- virtual void [showChild](#) ()  
*Show a newly added child.*
- virtual void [setEnabled](#) (bool enabled)  
*Set enabled/disabled state.*
- virtual void [setSize](#) (int newWidth, int newHeight)  
*Set the new size of the widget.*

#### 3.58.1 Detailed Description

Definition at line 35 of file [YQReplacePoint.h](#).

### 3.58.2 Member Function Documentation

#### 3.58.2.1 `setEnabled()`

```
void YQReplacePoint::setEnabled (
    bool enabled ) [virtual]
```

Set enabled/disabled state.

Reimplemented from `YWidget`.

Definition at line 64 of file [YQReplacePoint.cc](#).

#### 3.58.2.2 `setSize()`

```
void YQReplacePoint::setSize (
    int newWidth,
    int newHeight ) [virtual]
```

Set the new size of the widget.

Reimplemented from `YWidget`.

Definition at line 71 of file [YQReplacePoint.cc](#).

#### 3.58.2.3 `showChild()`

```
void YQReplacePoint::showChild ( ) [virtual]
```

Show a newly added child.

Reimplemented from `YReplacePoint`.

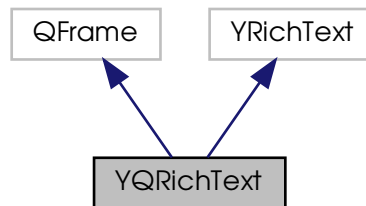
Definition at line 41 of file [YQReplacePoint.cc](#).

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

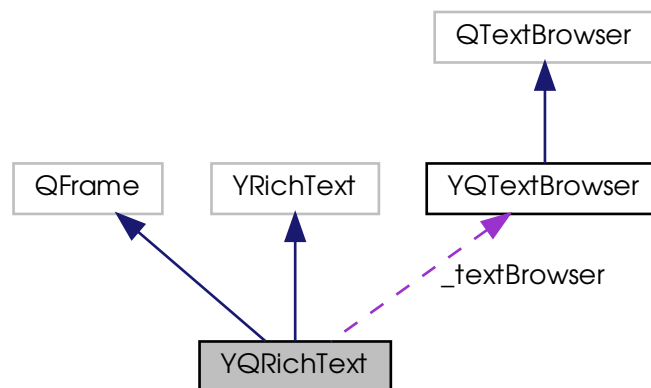
- `/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQReplacePoint.h`
- `/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQReplacePoint.cc`

## 3.59 YQRichText Class Reference

Inheritance diagram for YQRichText:



Collaboration diagram for YQRichText:



### Public Member Functions

- [YQRichText](#) (YWidget \*parent, const std::string &text, bool plainTextMode=false)  
*Constructor.*
- virtual [~YQRichText](#) ()  
*Destructor.*
- virtual void [setValue](#) (const std::string &newValue)  
*Change the text content of the RichText widget.*

- virtual void [setPlainTextMode](#) (bool on=true)  
*Set this RichText widget's "plain text" mode on or off.*
- virtual void [setAutoScrollDown](#) (bool on=true)  
*Set this RichText widget's "auto scroll down" mode on or off.*
- virtual int [preferredWidth](#) ()  
*Preferred width of the widget.*
- virtual int [preferredHeight](#) ()  
*Preferred height of the widget.*
- virtual void [setSize](#) (int newWidth, int newHeight)  
*Set the new size of the widget.*
- virtual void [setEnabled](#) (bool enabled)  
*Set enabled/disabled state.*
- virtual bool [setKeyboardFocus](#) ()  
*Accept the keyboard focus.*
- virtual bool [eventFilter](#) (QObject \*obj, QEvent \*ev)  
*Event filter.*
- bool [haveHyperLinks](#) ()  
*Returns 'true' if the current text of this RichText widget contains hyperlinks.*

## Protected Slots

- void [linkClicked](#) (const QUrl &url)  
*Notification that a hyperlink is clicked.*

## Protected Attributes

- [YQTextBrowser](#) \* [\\_textBrowser](#)

### 3.59.1 Detailed Description

Definition at line 38 of file [YQRichText.h](#).

### 3.59.2 Member Function Documentation



## 3.59.2.1 eventFilter()

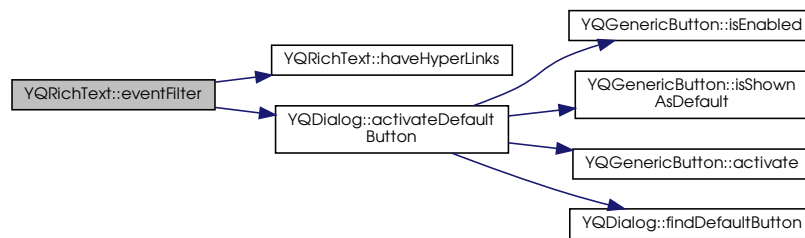
```
bool YQRichText::eventFilter (
    QObject * obj,
    QEvent * ev ) [virtual]
```

Event filter.

Reimplemented from QWidget.

Definition at line 157 of file [YQRichText.cc](#).

Here is the call graph for this function:



## 3.59.2.2 preferredHeight()

```
int YQRichText::preferredHeight ( ) [virtual]
```

Preferred height of the widget.

Reimplemented from YWidget.

Definition at line 196 of file [YQRichText.cc](#).

## 3.59.2.3 preferredWidth()

```
int YQRichText::preferredWidth ( ) [virtual]
```

Preferred width of the widget.

Reimplemented from YWidget.

Definition at line 190 of file [YQRichText.cc](#).

#### 3.59.2.4 setAutoScrollDown()

```
void YQRichText::setAutoScrollDown (
    bool on = true ) [virtual]
```

Set this RichText widget's "auto scroll down" mode on or off.

Reimplemented from YRichText.

Definition at line 141 of file [YQRichText.cc](#).

#### 3.59.2.5 setEnabled()

```
void YQRichText::setEnabled (
    bool enabled ) [virtual]
```

Set enabled/disabled state.

Reimplemented from YWidget.

Definition at line 208 of file [YQRichText.cc](#).

#### 3.59.2.6 setKeyboardFocus()

```
bool YQRichText::setKeyboardFocus ( ) [virtual]
```

Accept the keyboard focus.

Reimplemented from YWidget.

Definition at line 215 of file [YQRichText.cc](#).

#### 3.59.2.7 setPlainTextMode()

```
void YQRichText::setPlainTextMode (
    bool on = true ) [virtual]
```

Set this RichText widget's "plain text" mode on or off.

Reimplemented from YRichText.

Definition at line 130 of file [YQRichText.cc](#).

### 3.59.2.8 setSize()

```
void YQRichText::setSize (
    int newWidth,
    int newHeight ) [virtual]
```

Set the new size of the widget.

Reimplemented from YWidget.

Definition at line 202 of file [YQRichText.cc](#).

### 3.59.2.9 setValue()

```
void YQRichText::setValue (
    const std::string & newValue ) [virtual]
```

Change the text content of the RichText widget.

Reimplemented from YRichText.

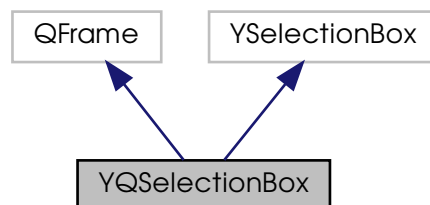
Definition at line 98 of file [YQRichText.cc](#).

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

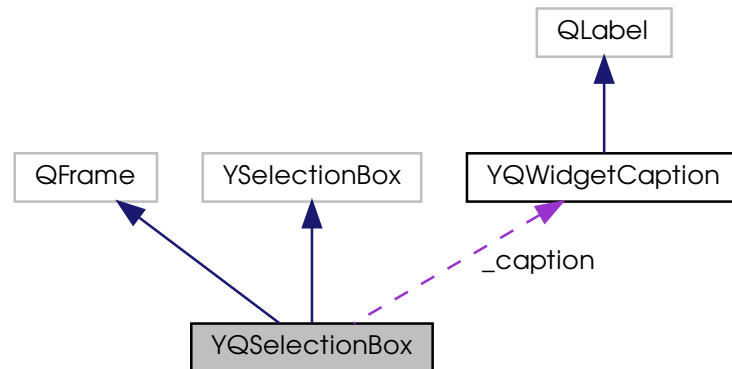
- [/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQRichText.h](#)
- [/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQRichText.cc](#)

## 3.60 YQSelectionBox Class Reference

Inheritance diagram for YQSelectionBox:



Collaboration diagram for YQSelectionBox:



## Public Member Functions

- [YQSelectionBox](#) (YWidget \*parent, const std::string &label)  
*Constructor.*
- virtual [~YQSelectionBox](#) ()  
*Destructor.*
- virtual void [setLabel](#) (const std::string &label)  
*Change the label text.*
- virtual void [addItem](#) (YItem \*item)  
*Add an item.*
- virtual void [addItem](#)s (const YItemCollection &itemCollection)  
*Add multiple items.*
- virtual void [selectItem](#) (YItem \*item, bool selected=true)  
*Select or deselect an item.*
- virtual void [deselectAllItems](#) ()  
*Deselect all items.*
- virtual void [deleteAllItems](#) ()  
*Delete all items.*
- virtual void [setEnabled](#) (bool enabled)  
*Set enabled/disabled state.*
- virtual int [preferredWidth](#) ()  
*Preferred width of the widget.*
- virtual int [preferredHeight](#) ()  
*Preferred height of the widget.*
- virtual void [setSize](#) (int newWidth, int newHeight)  
*Set the new size of the widget.*
- virtual bool [setKeyboardFocus](#) ()  
*Accept the keyboard focus.*
- virtual bool [eventFilter](#) (QObject \*obj, QEvent \*ev)  
*Event filter.*

## Protected Slots

- void [slotSelectionChanged](#) ()  
*Notification that an item has been selected.*
- void [slotActivated](#) (QListWidgetItem \*item)  
*Notification that an item has been activated (double clicked).*
- void [returnDelayed](#) ()  
*Return after some milliseconds delay - collect multiple events.*
- void [returnImmediately](#) ()  
*Return immediately.*

## Protected Member Functions

- void [selectItem](#) (int index)  
*Select an item by index.*
- void [addItem](#) (YItem \*item, bool batchMode)  
*Internal [addItem\(\)](#) method that will not do expensive operations in batch mode.*

## Protected Attributes

- [YQWidgetCaption](#) \* **\_caption**
- QListWidget \* **\_qt\_listWidget**
- QTimer **\_timer**

### 3.60.1 Detailed Description

Definition at line 39 of file [YQSelectionBox.h](#).

### 3.60.2 Member Function Documentation

#### 3.60.2.1 addItem()

```
void YQSelectionBox::addItem (  
    YItem * item ) [virtual]
```

Add an item.

Reimplemented from YSelectionWidget.

Definition at line 118 of file [YQSelectionBox.cc](#).

### 3.60.2.2 addItem()

```
void YQSelectionBox::addItem (
    const YItemCollection & itemCollection ) [virtual]
```

Add multiple items.

Reimplemented for efficiency from YSelectionWidget.

Definition at line 103 of file [YQSelectionBox.cc](#).

Here is the call graph for this function:



### 3.60.2.3 deleteAllItems()

```
void YQSelectionBox::deleteAllItems ( ) [virtual]
```

Delete all items.

Reimplemented from YSelectionWidget.

Definition at line 212 of file [YQSelectionBox.cc](#).

### 3.60.2.4 deselectAllItems()

```
void YQSelectionBox::deselectAllItems ( ) [virtual]
```

Deselect all items.

Reimplemented from YSelectionWidget.

Definition at line 188 of file [YQSelectionBox.cc](#).

Here is the call graph for this function:



## 3.60.2.5 eventFilter()

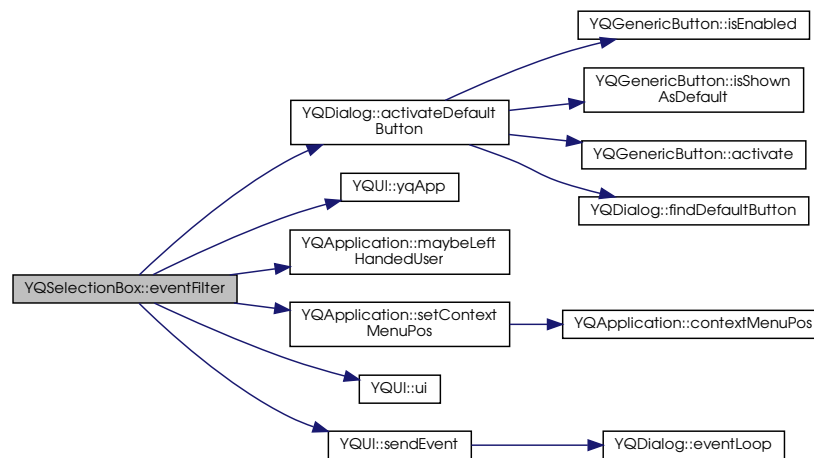
```
bool YQSelectionBox::eventFilter (
    QObject * obj,
    QEvent * ev ) [virtual]
```

Event filter.

Reimplemented from QWidget.

Definition at line 265 of file [YQSelectionBox.cc](#).

Here is the call graph for this function:



## 3.60.2.6 preferredHeight()

```
int YQSelectionBox::preferredHeight ( ) [virtual]
```

Preferred height of the widget.

Reimplemented from QWidget.

Definition at line 231 of file [YQSelectionBox.cc](#).

### 3.60.2.7 preferredWidth()

```
int YQSelectionBox::preferredWidth ( ) [virtual]
```

Preferred width of the widget.

Reimplemented from YWidget.

Definition at line 222 of file [YQSelectionBox.cc](#).

### 3.60.2.8 returnDelayed

```
void YQSelectionBox::returnDelayed ( ) [protected], [slot]
```

Return after some milliseconds delay - collect multiple events.

This is only relevant if `opt (notify )` is set.

Definition at line 381 of file [YQSelectionBox.cc](#).

### 3.60.2.9 returnImmediately

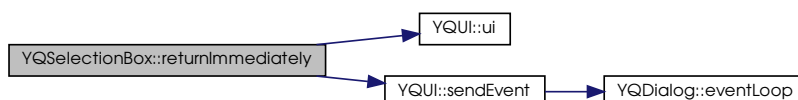
```
void YQSelectionBox::returnImmediately ( ) [protected], [slot]
```

Return immediately.

This is only relevant if `opt (notify )` is set.

Definition at line 358 of file [YQSelectionBox.cc](#).

Here is the call graph for this function:





#### 3.60.2.10 selectItem()

```
void YQSelectionBox::selectItem (
    YItem * item,
    bool selected = true ) [virtual]
```

Select or deselect an item.

Reimplemented from YSelectionWidget.

Definition at line 161 of file [YQSelectionBox.cc](#).

#### 3.60.2.11 setEnabled()

```
void YQSelectionBox::setEnabled (
    bool enabled ) [virtual]
```

Set enabled/disabled state.

Reimplemented from YWidget.

Definition at line 248 of file [YQSelectionBox.cc](#).

#### 3.60.2.12 setKeyboardFocus()

```
bool YQSelectionBox::setKeyboardFocus ( ) [virtual]
```

Accept the keyboard focus.

Reimplemented from YWidget.

Definition at line 257 of file [YQSelectionBox.cc](#).

### 3.60.2.13 `setLabel()`

```
void YQSelectionBox::setLabel (
    const std::string & label ) [virtual]
```

Change the label text.

Reimplemented from `YSelectionWidget`.

Definition at line 96 of file [YQSelectionBox.cc](#).

Here is the call graph for this function:



### 3.60.2.14 `setSize()`

```
void YQSelectionBox::setSize (
    int newWidth,
    int newHeight ) [virtual]
```

Set the new size of the widget.

Reimplemented from `YWidget`.

Definition at line 242 of file [YQSelectionBox.cc](#).

## 3.60.2.15 slotSelectionChanged

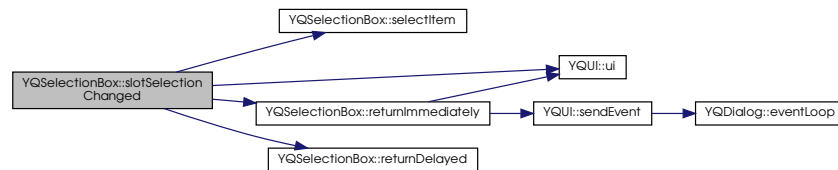
```
void YQSelectionBox::slotSelectionChanged ( ) [protected], [slot]
```

Notification that an item has been selected.

This is only relevant if `opt (notify )` is set.

Definition at line 306 of file [YQSelectionBox.cc](#).

Here is the call graph for this function:



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

- `/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQSelectionBox.h`
- `/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQSelectionBox.cc`

## 3.61 YQSignalBlocker Class Reference

Helper class to block Qt signals for QWidgets or QObjects as long as this object exists.

```
#include <YQSignalBlocker.h>
```

### Public Member Functions

- [YQSignalBlocker](#) (QObject \*qobject)  
*Constructor.*
- [~YQSignalBlocker](#) ()  
*Destructor.*

#### 3.61.1 Detailed Description

Helper class to block Qt signals for QWidgets or QObjects as long as this object exists.

This object will restore the old blocked state when it goes out of scope.

Definition at line 36 of file [YQSignalBlocker.h](#).

### 3.61.2 Constructor & Destructor Documentation

#### 3.61.2.1 YQSignalBlocker()

```
YQSignalBlocker::YQSignalBlocker (
    QObject * qobject )
```

Constructor.

Qt signals for 'qobject' will be blocked as long as this object exists. Remember that a QWidget is also a QObject.

Definition at line 29 of file [YQSignalBlocker.cc](#).

#### 3.61.2.2 ~YQSignalBlocker()

```
YQSignalBlocker::~~YQSignalBlocker ( )
```

Destructor.

This will restore the old signal state.

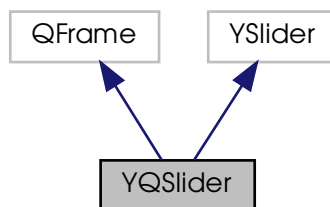
Definition at line 40 of file [YQSignalBlocker.cc](#).

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

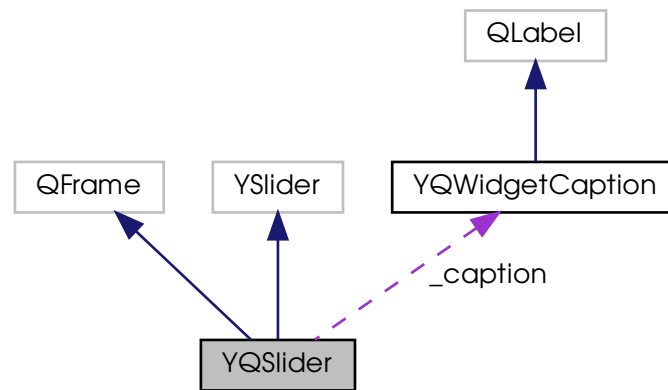
- /usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQSignalBlocker.h
- /usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQSignalBlocker.cc

## 3.62 YQSlider Class Reference

Inheritance diagram for YQSlider:



Collaboration diagram for YQSlider:



### Public Slots

- void [valueChangedSlot](#) (int newValue)  
*Slot for "value changed".*

### Signals

- void [valueChanged](#) (int newValue)  
*Emitted when the value changes (regardless of the notify flag).*

### Public Member Functions

- [YQSlider](#) (YWidget \*parent, const std::string &label, int minValue, int maxValue, int initialValue, bool reverse←Layout=false)  
*Constructor.*
- virtual [~YQSlider](#) ()  
*Destructor.*
- virtual int [value](#) ()  
*Get the current value (the number entered by the user or set from the outside) of this slider.*
- virtual void [setLabel](#) (const std::string &label)  
*Set the label (the caption above the input field).*
- virtual void [setEnabled](#) (bool enabled)  
*Set enabled/disabled state.*
- virtual int [preferredWidth](#) ()  
*Preferred width of the widget.*

- virtual int [preferredHeight](#) ()  
*Preferred height of the widget.*
- virtual void [setSize](#) (int newWidth, int newHeight)  
*Set the new size of the widget.*
- virtual bool [setKeyboardFocus](#) ()  
*Accept the keyboard focus.*

### Protected Member Functions

- virtual void [setValueInternal](#) (int val)  
*Set the current value (the number entered by the user or set from the outside) of this slider.*

### Protected Attributes

- [YQWidgetCaption](#) \* **\_caption**
- [QSlider](#) \* **\_qt\_slider**
- [QSpinBox](#) \* **\_qt\_spinBox**
- [QFrame](#) \* **\_hbox**

### 3.62.1 Detailed Description

Definition at line 36 of file [YQSlider.h](#).

### 3.62.2 Constructor & Destructor Documentation

#### 3.62.2.1 YQSlider()

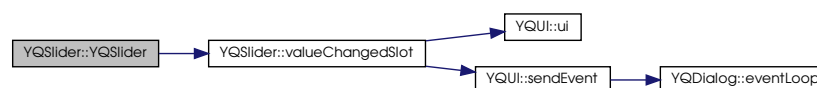
```
YQSlider::YQSlider (
    QWidget * parent,
    const std::string & label,
    int minValue,
    int maxValue,
    int initialValue,
    bool reverseLayout = false )
```

Constructor.

'reverseLayout' creates the QSpinBox to the left and the QSlider to the right.

Definition at line 41 of file [YQSlider.cc](#).

Here is the call graph for this function:



### 3.62.3 Member Function Documentation

#### 3.62.3.1 preferredHeight()

```
int YQSlider::preferredHeight ( ) [virtual]
```

Preferred height of the widget.

Reimplemented from YWidget.

Definition at line 174 of file [YQSlider.cc](#).

#### 3.62.3.2 preferredWidth()

```
int YQSlider::preferredWidth ( ) [virtual]
```

Preferred width of the widget.

Reimplemented from YWidget.

Definition at line 164 of file [YQSlider.cc](#).

#### 3.62.3.3 setEnabled()

```
void YQSlider::setEnabled (
    bool enabled ) [virtual]
```

Set enabled/disabled state.

Reimplemented from YWidget.

Definition at line 154 of file [YQSlider.cc](#).

#### 3.62.3.4 setLabel()

```
void YQSlider::setLabel (
    const std::string & label ) [virtual]
```

Set the label (the caption above the input field).

Reimplemented from YSlider / YIntField.

Definition at line 188 of file [YQSlider.cc](#).

Here is the call graph for this function:



#### 3.62.3.5 setSize()

```
void YQSlider::setSize (
    int newWidth,
    int newHeight ) [virtual]
```

Set the new size of the widget.

Reimplemented from YWidget.

Definition at line 181 of file [YQSlider.cc](#).

#### 3.62.3.6 setValueInternal()

```
void YQSlider::setValueInternal (
    int val ) [protected], [virtual]
```

Set the current value (the number entered by the user or set from the outside) of this slider.

'val' is guaranteed to be between minValue and maxValue; no further checks are required.

Implemented from YSlider/ YIntField.

Definition at line 134 of file [YQSlider.cc](#).



### 3.62.3.7 value()

```
int YQSlider::value ( ) [virtual]
```

Get the current value (the number entered by the user or set from the outside) of this slider.

Implemented from YSlider / YIntField.

Definition at line 127 of file [YQSlider.cc](#).

### 3.62.3.8 valueChangedSlot

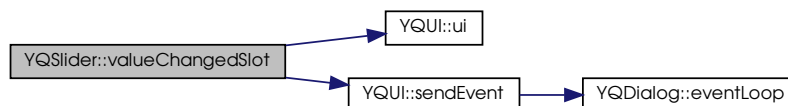
```
void YQSlider::valueChangedSlot (
    int newValue ) [slot]
```

Slot for "value changed".

This will send a ValueChanged event if 'notify' is set.

Definition at line 144 of file [YQSlider.cc](#).

Here is the call graph for this function:

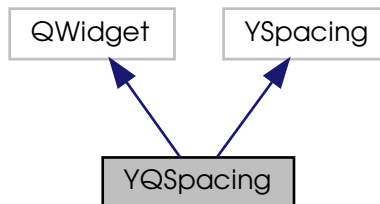


The documentation for this class was generated from the following files:

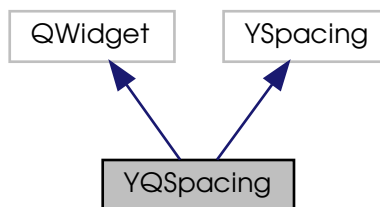
- `/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQSlider.h`
- `/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQSlider.cc`

### 3.63 YQSpacing Class Reference

Inheritance diagram for YQSpacing:



Collaboration diagram for YQSpacing:



#### Public Member Functions

- [YQSpacing](#) (YWidget \*parent, YUIDimension dim, bool stretchable=false, YLayoutSize\_t layoutUnits=0.0)  
*Constructor.*
- virtual void [setSize](#) (int newWidth, int newHeight)  
*Set the new size of the widget.*

#### 3.63.1 Detailed Description

Definition at line 32 of file [YQSpacing.h](#).

### 3.63.2 Member Function Documentation

#### 3.63.2.1 setSize()

```
void YQSpacing::setSize (
    int newWidth,
    int newHeight ) [virtual]
```

Set the new size of the widget.

Reimplemented from YWidget.

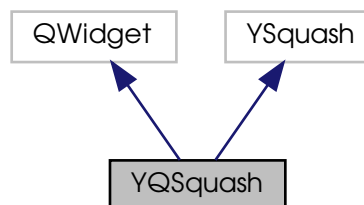
Definition at line 46 of file [YQSpacing.cc](#).

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

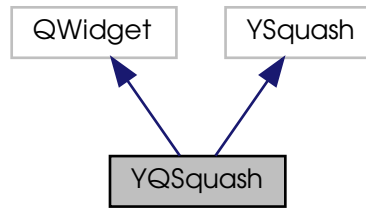
- [/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQSpacing.h](#)
- [/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQSpacing.cc](#)

## 3.64 YQSquash Class Reference

Inheritance diagram for YQSquash:



Collaboration diagram for YQSquash:



### Public Member Functions

- [YQSquash](#) (YWidget \*parent, bool horSquash, bool vertSquash)  
*Constructor.*
- virtual [~YQSquash](#) ()  
*Destructor.*
- virtual void [setEnabled](#) (bool enabled)  
*Set enabled/disabled state.*
- virtual void [setSize](#) (int newWidth, int newHeight)  
*Set the new size of the widget.*

### 3.64.1 Detailed Description

Definition at line 35 of file [YQSquash.h](#).

### 3.64.2 Member Function Documentation

#### 3.64.2.1 setEnabled()

```
void YQSquash::setEnabled (  
    bool enabled ) [virtual]
```

Set enabled/disabled state.

Reimplemented from YWidget.

Definition at line 45 of file [YQSquash.cc](#).

## 3.64.2.2 setSize()

```
void YQSquash::setSize (
    int newWidth,
    int newHeight ) [virtual]
```

Set the new size of the widget.

Reimplemented from YWidget.

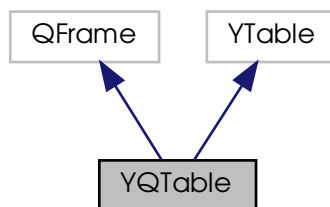
Definition at line 52 of file [YQSquash.cc](#).

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

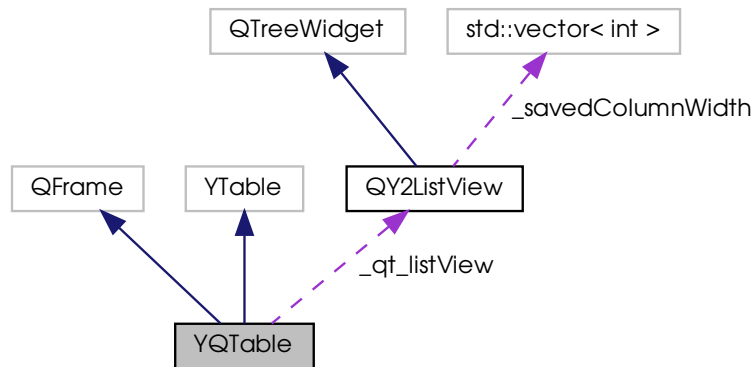
- [/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQSquash.h](#)
- [/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQSquash.cc](#)

## 3.65 YQTable Class Reference

Inheritance diagram for YQTable:



Collaboration diagram for YQTable:



## Public Member Functions

- **YQTable** (YWidget \*parent, YTableHeader \*header, bool multiSelection)  
*Constructor.*
- virtual **~YQTable** ()  
*Destructor.*
- virtual void **setKeepSorting** (bool keepSorting)  
*Switch between sorting by item insertion order (keepSorting: true) or allowing the user to sort by an arbitrary column (by clicking on the column header).*
- virtual void **addItem** (YItem \*item)  
*Add an item.*
- virtual void **addItems** (const YItemCollection &itemCollection)  
*Add multiple items.*
- virtual void **selectItem** (YItem \*item, bool selected=true)  
*Select or deselect an item.*
- virtual void **deselectAllItems** ()  
*Deselect all items.*
- virtual void **deleteAllItems** ()  
*Delete all items.*
- virtual void **cellChanged** (const YTableCell \*cell)  
*Notification that a cell (its text and/or its icon) was changed from the outside.*
- virtual void **setEnabled** (bool enabled)  
*Set enabled/disabled state.*
- virtual int **preferredWidth** ()  
*Preferred width of the widget.*
- virtual int **preferredHeight** ()  
*Preferred height of the widget.*
- virtual void **setSize** (int newWidth, int newHeight)

*Set the new size of the widget.*

- virtual bool [setKeyboardFocus](#) ()

*Accept the keyboard focus.*

## Protected Slots

- void [slotSelected](#) (QTreeWidgetItem \*)  
*Notification that an item is selected (single click or keyboard).*
- void [slotSelectionChanged](#) ()  
*Notification that the item selection changed (relevant for multiSelection mode).*
- void [slotActivated](#) (QTreeWidgetItem \*)  
*Notification that an item is activated (double click or keyboard).*
- void [slotContextMenu](#) (const QPoint &pos)  
*Propagate a context menu selection.*

## Protected Member Functions

- void [selectOrigItem](#) (QTreeWidgetItem \*listViewItem)  
*Select the original item (the YTableWidgetItem) that corresponds to the specified listViewItem.*
- void [addItem](#) (YItem \*item, bool batchMode, bool resizeColumnsToContent)  
*Internal [addItem\(\)](#) method that will not do expensive operations in batch mode.*

## Protected Attributes

- [QY2ListView](#) \* [\\_qt\\_listView](#)

### 3.65.1 Detailed Description

Definition at line 37 of file [YQTable.h](#).

### 3.65.2 Member Function Documentation

#### 3.65.2.1 addItem()

```
void YQTable::addItem (  
    YItem * item ) [virtual]
```

Add an item.

Reimplemented from YSelectionWidget.

Definition at line 124 of file [YQTable.cc](#).

### 3.65.2.2 addItem()

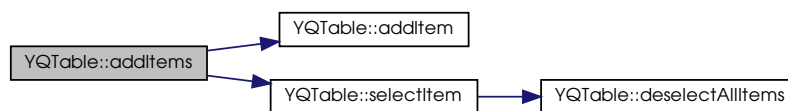
```
void YQTable::addItem (
    const YItemCollection & itemCollection ) [virtual]
```

Add multiple items.

Reimplemented for efficiency from YSelectionWidget.

Definition at line 181 of file [YQTable.cc](#).

Here is the call graph for this function:



### 3.65.2.3 cellChanged()

```
void YQTable::cellChanged (
    const YTableCell * cell ) [virtual]
```

Notification that a cell (its text and/or its icon) was changed from the outside.

Applications are required to call this whenever a table cell is changed after adding the corresponding table item (the row) to the table widget.

Reimplemented from YTable.

Definition at line 253 of file [YQTable.cc](#).

Here is the call graph for this function:





#### 3.65.2.4 deleteAllItems()

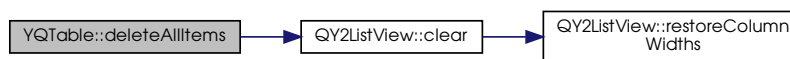
```
void YQTable::deleteAllItems ( ) [virtual]
```

Delete all items.

Reimplemented from YSelectionWidget.

Definition at line 245 of file [YQTable.cc](#).

Here is the call graph for this function:



#### 3.65.2.5 deselectAllItems()

```
void YQTable::deselectAllItems ( ) [virtual]
```

Deselect all items.

Reimplemented from YSelectionWidget.

Definition at line 235 of file [YQTable.cc](#).

#### 3.65.2.6 preferredHeight()

```
int YQTable::preferredHeight ( ) [virtual]
```

Preferred height of the widget.

Reimplemented from YWidget.

Definition at line 373 of file [YQTable.cc](#).

### 3.65.2.7 preferredWidth()

```
int YQTable::preferredWidth ( ) [virtual]
```

Preferred width of the widget.

Reimplemented from YWidget.

Definition at line 363 of file [YQTable.cc](#).

### 3.65.2.8 selectItem()

```
void YQTable::selectItem (
    YItem * item,
    bool selected = true ) [virtual]
```

Select or deselect an item.

Reimplemented from YSelectionWidget.

Definition at line 208 of file [YQTable.cc](#).

Here is the call graph for this function:



### 3.65.2.9 setEnabled()

```
void YQTable::setEnabled (
    bool enabled ) [virtual]
```

Set enabled/disabled state.

Reimplemented from YWidget.

Definition at line 353 of file [YQTable.cc](#).

#### 3.65.2.10 setKeepSorting()

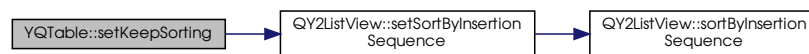
```
void YQTable::setKeepSorting (
    bool keepSorting ) [virtual]
```

Switch between sorting by item insertion order (keepSorting: true) or allowing the user to sort by an arbitrary column (by clicking on the column header).

Reimplemented from YTable.

Definition at line 115 of file [YQTable.cc](#).

Here is the call graph for this function:



#### 3.65.2.11 setKeyboardFocus()

```
bool YQTable::setKeyboardFocus ( ) [virtual]
```

Accept the keyboard focus.

Reimplemented from YWidget.

Definition at line 390 of file [YQTable.cc](#).

#### 3.65.2.12 setSize()

```
void YQTable::setSize (
    int newWidth,
    int newHeight ) [virtual]
```

Set the new size of the widget.

Reimplemented from YWidget.

Definition at line 383 of file [YQTable.cc](#).

### 3.65.2.13 slotContextMenu

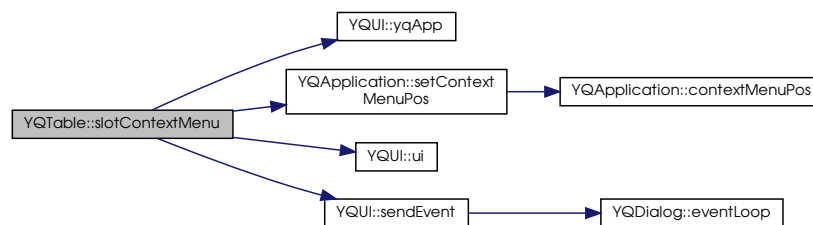
```
void YQTable::slotContextMenu (
    const QPoint & pos ) [protected], [slot]
```

Propagate a context menu selection.

This will trigger an 'ContextMenuActivated' event if 'notifyContextMenu' is set.

Definition at line 399 of file [YQTable.cc](#).

Here is the call graph for this function:



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

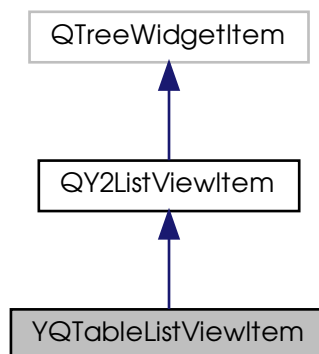
- `/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQTable.h`
- `/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQTable.cc`

## 3.66 YQTableListViewItem Class Reference

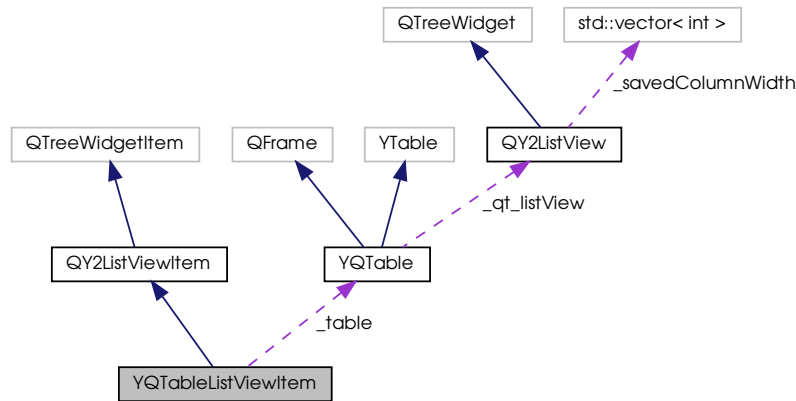
Visual representation of a YTableWidgetItem.

```
#include <YQTable.h>
```

Inheritance diagram for YQTableListViewItem:



Collaboration diagram for YQTableListViewItem:



## Public Member Functions

- `YQTableListViewItem` (`YQTable` \*`table`, `QY2ListView` \*`parent`, `YTableItem` \*`origItem`)  
*Constructor.*
- `YQTable` \* `table` () const  
*Return the parent table widget.*
- `YTableItem` \* `origItem` () const  
*Return the corresponding YTableItem.*
- void `updateCell` (const `YTableCell` \*`cell`)  
*Update this item's display with the content of 'cell'.*

## Protected Attributes

- `YQTable` \* `_table`
- `YTableItem` \* `_origItem`

### 3.66.1 Detailed Description

Visual representation of a `YTableItem`.

Definition at line 196 of file `YQTable.h`.

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

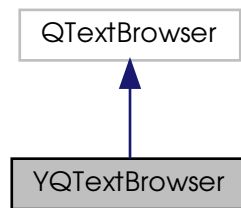
- `/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQTable.h`
- `/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQTable.cc`

### 3.67 YQTextBrowser Class Reference

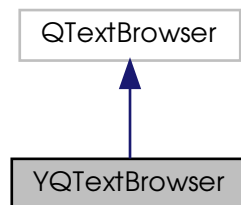
Helper class - needed to have the benefits of both QVBox as the base class for [YQRichText](#) so uniform margins can be implemented (outside, not inside the scroll window as QTextBrowser normally does it) and QTextBrowser's features (enable hyperlinks) without removing the text each time the user clicks.

```
#include <YQRichText.h>
```

Inheritance diagram for YQTextBrowser:



Collaboration diagram for YQTextBrowser:



#### Public Slots

- virtual void [setSource](#) (const QUrl &name)  
*Get the document pointed to by a hyperlink.*

#### Public Member Functions

- [YQTextBrowser](#) (QWidget \*parent=0)  
*Constructor.*

### 3.67.1 Detailed Description

Helper class - needed to have the benefits of both QVBox as the base class for [YQRichText](#) so uniform margins can be implemented (outside, not inside the scroll window as QTextBrowser normally does it) and QTextBrowser's features (enable hyperlinks) without removing the text each time the user clicks.

This class is required only to overwrite [setSource\(\)](#).

Definition at line 156 of file [YQRichText.h](#).

### 3.67.2 Member Function Documentation

#### 3.67.2.1 setSource

```
void YQTextBrowser::setSource (
    const QUrl & name ) [virtual], [slot]
```

Get the document pointed to by a hyperlink.

Reimplemented from QTextBrowser to avoid having an empty text each time the user clicks on a hyperlink.

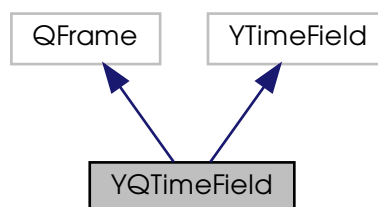
Definition at line 222 of file [YQRichText.cc](#).

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

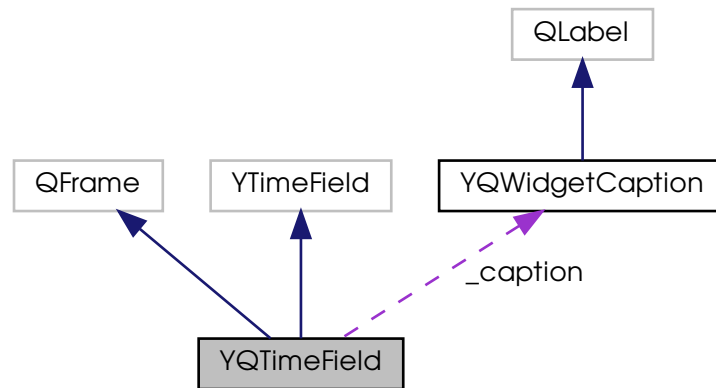
- [/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQRichText.h](#)
- [/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQRichText.cc](#)

## 3.68 YQTimeField Class Reference

Inheritance diagram for YQTimeField:



Collaboration diagram for YQTimeField:



## Public Member Functions

- [YQTimeField](#) (YWidget \*parent, const std::string &label)  
*Constructor.*
- virtual [~YQTimeField](#) ()  
*Destructor.*
- virtual std::string [value](#) ()  
*Get the current value (the text entered by the user or set from the outside) of this input field.*
- virtual void [setValue](#) (const std::string &newValue)  
*Set the current value (the text entered by the user or set from the outside) of this input field.*
- virtual void [setLabel](#) (const std::string &label)  
*Set the label (the caption above the input field).*
- virtual void [setEnabled](#) (bool enabled)  
*Set enabled/disabled state.*
- virtual int [preferredWidth](#) ()  
*Preferred width of the widget.*
- virtual int [preferredHeight](#) ()  
*Preferred height of the widget.*
- virtual void [setSize](#) (int newWidth, int newHeight)  
*Set the new size of the widget.*
- virtual bool [setKeyboardFocus](#) ()  
*Accept the keyboard focus.*

## Protected Attributes

- [YQWidgetCaption](#) \* `_caption`
- QTimeEdit \* `_qt_timeEdit`



### 3.68.1 Detailed Description

Definition at line 36 of file [YQTimeField.h](#).

### 3.68.2 Member Function Documentation

#### 3.68.2.1 preferredHeight()

```
int YQTimeField::preferredHeight ( ) [virtual]
```

Preferred height of the widget.

Reimplemented from YWidget.

Definition at line 106 of file [YQTimeField.cc](#).

#### 3.68.2.2 preferredWidth()

```
int YQTimeField::preferredWidth ( ) [virtual]
```

Preferred width of the widget.

Reimplemented from YWidget.

Definition at line 100 of file [YQTimeField.cc](#).

#### 3.68.2.3 setEnabled()

```
void YQTimeField::setEnabled (
    bool enabled ) [virtual]
```

Set enabled/disabled state.

Reimplemented from YWidget.

Definition at line 93 of file [YQTimeField.cc](#).

#### 3.68.2.4 setKeyboardFocus()

```
bool YQTimeField::setKeyboardFocus ( ) [virtual]
```

Accept the keyboard focus.

Reimplemented from YWidget.

Definition at line 118 of file [YQTimeField.cc](#).

#### 3.68.2.5 setLabel()

```
void YQTimeField::setLabel (
    const std::string & label ) [virtual]
```

Set the label (the caption above the input field).

Reimplemented from YSimpleInputField.

Definition at line 86 of file [YQTimeField.cc](#).

Here is the call graph for this function:



#### 3.68.2.6 setSize()

```
void YQTimeField::setSize (
    int newWidth,
    int newHeight ) [virtual]
```

Set the new size of the widget.

Reimplemented from YWidget.

Definition at line 112 of file [YQTimeField.cc](#).

### 3.68.2.7 setValue()

```
void YQTimeField::setValue (
    const std::string & newValue ) [virtual]
```

Set the current value (the text entered by the user or set from the outside) of this input field.

Implemented from YSimpleInputField.

Definition at line 78 of file [YQTimeField.cc](#).

### 3.68.2.8 value()

```
string YQTimeField::value ( ) [virtual]
```

Get the current value (the text entered by the user or set from the outside) of this input field.

Implemented from YSimpleInputField.

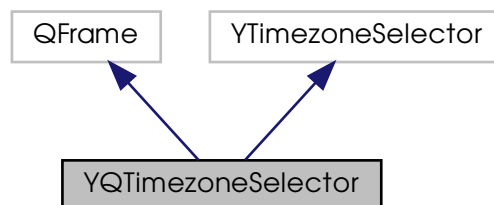
Definition at line 72 of file [YQTimeField.cc](#).

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

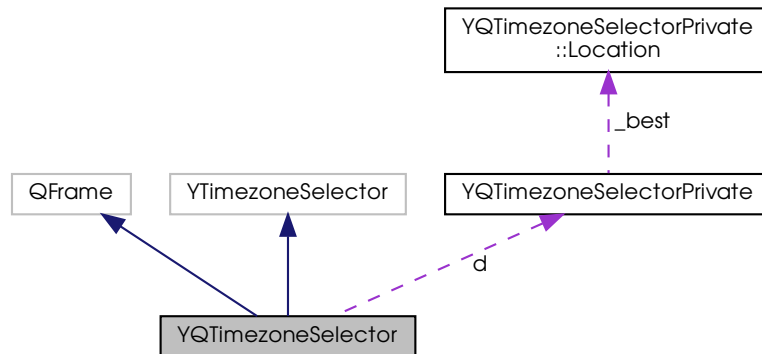
- [/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQTimeField.h](#)
- [/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQTimeField.cc](#)

## 3.69 YQTimezoneSelector Class Reference

Inheritance diagram for YQTimezoneSelector:



Collaboration diagram for YQTimezoneSelector:



## Public Member Functions

- [YQTimezoneSelector](#) (YWidget \*parent, const std::string &ixmap, const std::map< std::string, std::string > &timezones)  
*Constructor.*
- virtual [~YQTimezoneSelector](#) ()  
*Destructor.*
- virtual int [preferredWidth](#) ()  
*Preferred width of the widget.*
- virtual int [preferredHeight](#) ()  
*Preferred height of the widget.*
- virtual void [setSize](#) (int newWidth, int newHeight)  
*Set the new size of the widget.*
- virtual void **paintEvent** (QPaintEvent \*event)
- virtual void **mousePressEvent** (QMouseEvent \*event)
- virtual bool **event** (QEvent \*event)
- virtual std::string [currentZone](#) () const  
*subclasses have to implement this to return value*
- virtual void [setCurrentZone](#) (const std::string &zone, bool zoom)  
*subclasses have to implement this to set value*

## Protected Slots

- void **slotBlink** ()

## Protected Attributes

- [YQTimezoneSelectorPrivate](#) \* **d**

### 3.69.1 Detailed Description

Definition at line 34 of file [YQTimezoneSelector.h](#).

### 3.69.2 Member Function Documentation

#### 3.69.2.1 preferredHeight()

```
int YQTimezoneSelector::preferredHeight ( ) [virtual]
```

Preferred height of the widget.

Reimplemented from YWidget.

Definition at line 180 of file [YQTimezoneSelector.cc](#).

#### 3.69.2.2 preferredWidth()

```
int YQTimezoneSelector::preferredWidth ( ) [virtual]
```

Preferred width of the widget.

Reimplemented from YWidget.

Definition at line 174 of file [YQTimezoneSelector.cc](#).

#### 3.69.2.3 setSize()

```
void YQTimezoneSelector::setSize (
    int newWidth,
    int newHeight ) [virtual]
```

Set the new size of the widget.

Reimplemented from YWidget.

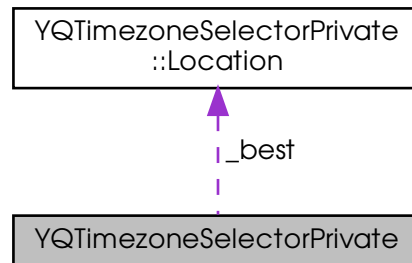
Definition at line 186 of file [YQTimezoneSelector.cc](#).

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

- [/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQTimezoneSelector.h](#)
- [/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQTimezoneSelector.cc](#)

### 3.70 YQTimezoneSelectorPrivate Class Reference

Collaboration diagram for YQTimezoneSelectorPrivate:



#### Classes

- struct [Location](#)

#### Public Member Functions

- **YQTimezoneSelectorPrivate** (QWidget \*p)
- [Location](#) **findBest** (const QPoint &pos) const
- QPoint **pixPosition** (const [Location](#) &pos) const
- QPoint **pixToWindow** (const QPoint &pos) const

#### Public Attributes

- QImage **\_pix**
- QPoint **\_zoom**
- [Location](#) **\_best**
- QList< [Location](#) > **locations**
- QTimer \* **blink**
- int **highlight**
- QPixmap **cachePix**

#### 3.70.1 Detailed Description

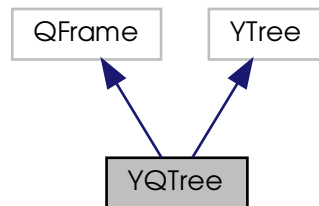
Definition at line 44 of file [YQTimezoneSelector.cc](#).

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

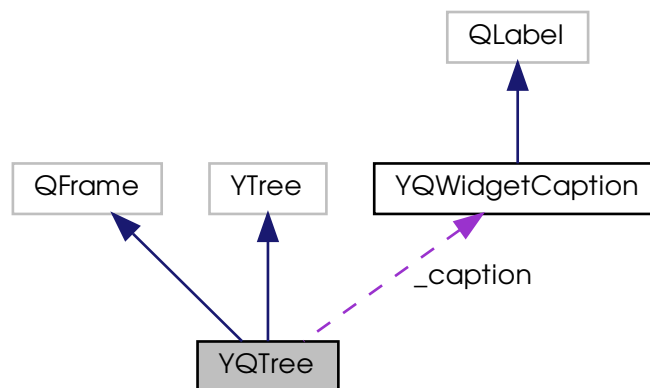
- /usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQTimezoneSelector.cc

## 3.71 YQTree Class Reference

Inheritance diagram for YQTree:



Collaboration diagram for YQTree:



### Public Member Functions

- [YQTree](#) (YWidget \*parent, const std::string &label, bool multiSelection, bool recursiveSelection)  
*Constructor.*
- virtual [~YQTree](#) ()  
*Destructor.*
- virtual void [setLabel](#) (const std::string &label)  
*Change the label text.*

- virtual void **rebuildTree** ()  
*Rebuild the displayed tree from the internally stored YTreeItems.*
- virtual void **selectItem** (YItem \*item, bool selected=true)  
*Select or deselect an item.*
- virtual void **deselectAllItems** ()  
*Deselect all items.*
- virtual void **deleteAllItems** ()  
*Delete all items.*
- virtual void **setEnabled** (bool enabled)  
*Set enabled/disabled state.*
- virtual int **preferredWidth** ()  
*Preferred width of the widget.*
- virtual int **preferredHeight** ()  
*Preferred height of the widget.*
- virtual void **setSize** (int newWidth, int newHeight)  
*Set the new size of the widget.*
- virtual bool **setKeyboardFocus** ()  
*Accept the keyboard focus.*
- virtual YTreeItem \* **currentItem** ()  
*Return the the item that currently has the keyboard focus or 0 if no item currently has the keyboard focus.*

## Protected Slots

- void **slotSelectionChanged** ()  
*Propagate a tree item selection.*
- void **slotItemChanged** (QTreeWidgetItem \*item)  
*Propagate a tree item change.*
- void **slotItemClicked** (QTreeWidgetItem \*item, int column)
- void **slotActivated** (QTreeWidgetItem \*item)  
*Propagate a double click or pressing the space key on a tree item.*
- void **slotItemExpanded** (QTreeWidgetItem \*item)  
*Propagate an "item expanded" event to the underlying YTreeItem.*
- void **slotItemCollapsed** (QTreeWidgetItem \*item)  
*Propagate an "item collapsed" event to the underlying YTreeItem.*
- void **slotContextMenu** (const QPoint &pos)  
*Propagate a context menu selection.*

## Protected Member Functions

- void **selectItem** (YQTreeItem \*item)  
*Select an item via the corresponding YQTreeItem.*
- void **selectItem** (QTreeWidgetItem \*item, bool selected, bool recursive)  
*Select or deselect an item, recursively.*
- void **openBranch** (YQTreeItem \*item)  
*Open the branch of 'item' recursively to its toplevel item.*
- void **buildDisplayTree** (YQTreeItem \*parentItem, YItemIterator begin, YItemIterator end)  
*Build a tree of items that will be displayed (YQTreeItems) from the original items between iterators 'begin' and 'end' as child items of 'parentItem' (or as toplevel items if 'parentItem' is 0).*



## Protected Attributes

- [YQWidgetCaption](#) \* `_caption`
- [QTreeWidget](#) \* `_qt_treeWidget`
- `int _nextSerialNo`

### 3.71.1 Detailed Description

Definition at line 38 of file [YQTree.h](#).

### 3.71.2 Member Function Documentation

#### 3.71.2.1 `currentItem()`

```
YTreeWidgetItem * YQTree::currentItem ( ) [virtual]
```

Return the the item that currently has the keyboard focus or 0 if no item currently has the keyboard focus.

Reimplemented from [YMultiSelectionBox](#).

Definition at line 422 of file [YQTree.cc](#).

Here is the call graph for this function:



#### 3.71.2.2 `deleteAllItems()`

```
void YQTree::deleteAllItems ( ) [virtual]
```

Delete all items.

Reimplemented from [YSelectionWidget](#).

Definition at line 262 of file [YQTree.cc](#).

### 3.71.2.3 `deselectAllItems()`

```
void YQTree::deselectAllItems ( ) [virtual]
```

Deselect all items.

Reimplemented from `YSelectionWidget`.

Definition at line 236 of file [YQTree.cc](#).

Here is the call graph for this function:



### 3.71.2.4 `preferredHeight()`

```
int YQTree::preferredHeight ( ) [virtual]
```

Preferred height of the widget.

Reimplemented from `YWidget`.

Definition at line 377 of file [YQTree.cc](#).

### 3.71.2.5 `preferredWidth()`

```
int YQTree::preferredWidth ( ) [virtual]
```

Preferred width of the widget.

Reimplemented from `YWidget`.

Definition at line 370 of file [YQTree.cc](#).

### 3.71.2.6 rebuildTree()

```
void YQTree::rebuildTree ( ) [virtual]
```

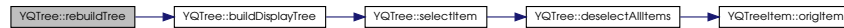
Rebuild the displayed tree from the internally stored YTreeItems.

The application should call this (once) after all items have been added with addItem(). YTree::addItem() calls this automatically.

Implemented from YTree.

Definition at line 122 of file [YQTree.cc](#).

Here is the call graph for this function:



### 3.71.2.7 selectItem()

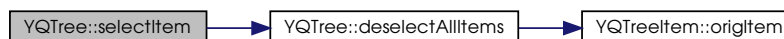
```
void YQTree::selectItem (
    YItem * item,
    bool selected = true ) [virtual]
```

Select or deselect an item.

Reimplemented from YSelectionWidget.

Definition at line 159 of file [YQTree.cc](#).

Here is the call graph for this function:



### 3.71.2.8 `setEnabled()`

```
void YQTree::setEnabled (
    bool enabled ) [virtual]
```

Set enabled/disbled state.

Reimplemented from YWidget.

Definition at line [394](#) of file [YQTree.cc](#).

### 3.71.2.9 `setLabel()`

```
void YQTree::setLabel (
    const std::string & label ) [virtual]
```

Change the label text.

Reimplemented from YSelectionWidget.

Definition at line [115](#) of file [YQTree.cc](#).

Here is the call graph for this function:



### 3.71.2.10 `setSize()`

```
void YQTree::setSize (
    int newWidth,
    int newHeight ) [virtual]
```

Set the new size of the widget.

Reimplemented from YWidget.

Definition at line [388](#) of file [YQTree.cc](#).

## 3.71.2.11 slotActivated

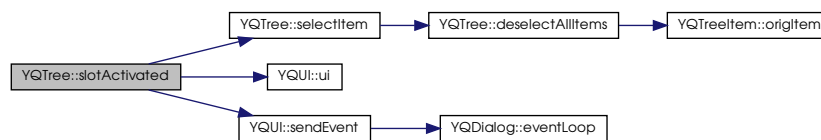
```
void YQTree::slotActivated (
    QTreeWidgetItem * item ) [protected], [slot]
```

Propagate a double click or pressing the space key on a tree item.

This will trigger an 'Activated' event if 'notify' is set.

Definition at line 361 of file [YQTree.cc](#).

Here is the call graph for this function:



## 3.71.2.12 slotContextMenu

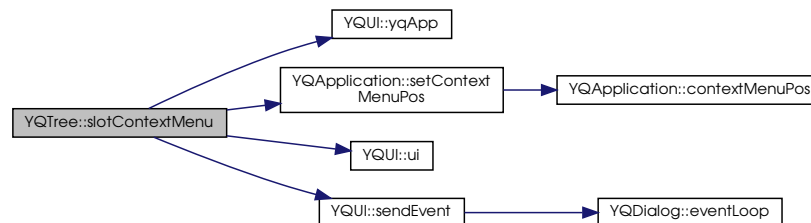
```
void YQTree::slotContextMenu (
    const QPoint & pos ) [protected], [slot]
```

Propagate a context menu selection.

This will trigger an 'ContextMenuActivated' event if 'notifyContextMenu' is set.

Definition at line 410 of file [YQTree.cc](#).

Here is the call graph for this function:



### 3.71.2.13 slotItemChanged

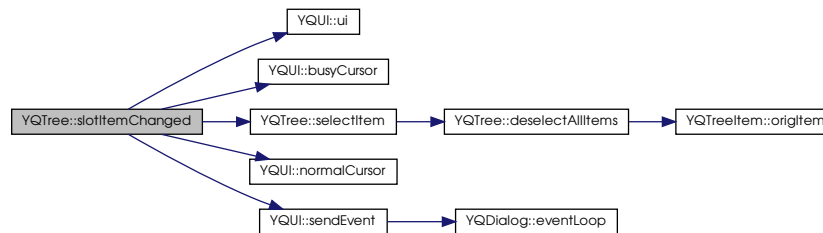
```
void YQTree::slotItemChanged (
    QTreeWidgetItem * item ) [protected], [slot]
```

Propagate a tree item change.

This will trigger a 'ValueChanged' event if 'notify' is set.

Definition at line 294 of file [YQTree.cc](#).

Here is the call graph for this function:



### 3.71.2.14 slotSelectionChanged

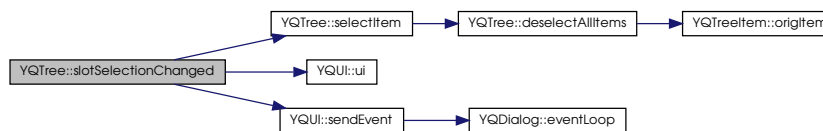
```
void YQTree::slotSelectionChanged ( ) [protected], [slot]
```

Propagate a tree item selection.

This will trigger a 'SelectionChanged' event if 'notify' is set.

Definition at line 345 of file [YQTree.cc](#).

Here is the call graph for this function:

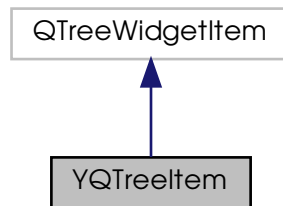


The documentation for this class was generated from the following files:

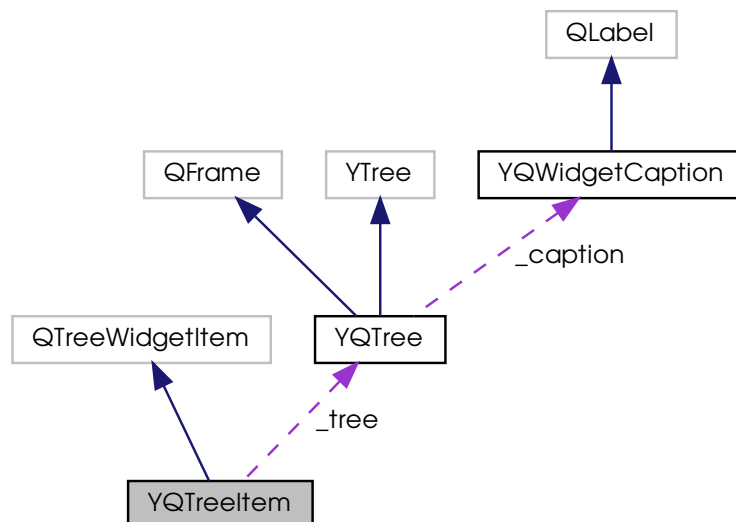
- `/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQTree.h`
- `/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQTree.cc`

## 3.72 YQTreeWidgetItem Class Reference

Inheritance diagram for YQTreeWidgetItem:



Collaboration diagram for YQTreeWidgetItem:



### Public Member Functions

- [YQTreeWidgetItem](#) ([YQTree](#) \*tree, [QTreeWidgetItem](#) \*parent, [YTreeWidgetItem](#) \*origItem, int serial)  
*Constructor for a top level item.*
- [YQTreeWidgetItem](#) ([YQTree](#) \*tree, [YQTreeWidgetItem](#) \*parent, [YTreeWidgetItem](#) \*origItem, int serial)

*Constructor for a non-top level item.*

- `YTreeItem * origItem ()`

*Returns the original YTreeItem of which this item is a clone.*

- `virtual void setOpen (bool open)`

*Open this item.*

## Protected Member Functions

- `QString key (int column, bool ascending) const`

*Sort key of this item.*

## Protected Attributes

- `YQTree * _tree`
- `YTreeItem * _origItem`
- `int _serialNo`

### 3.72.1 Detailed Description

Definition at line 220 of file [YQTree.h](#).

### 3.72.2 Member Function Documentation

#### 3.72.2.1 `key()`

```
QString YQTreeItem::key (  
    int column,  
    bool ascending ) const [protected]
```

Sort key of this item.

Reimplemented from `QTreeWidgetItem`.

Definition at line 520 of file [YQTree.cc](#).



## 3.72.2.2 setOpen()

```
void YQTreeItem::setOpen (
    bool open ) [virtual]
```

Open this item.

Reimplemented from QTreeWidgetItem.

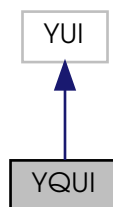
Definition at line 511 of file [YQTree.cc](#).

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

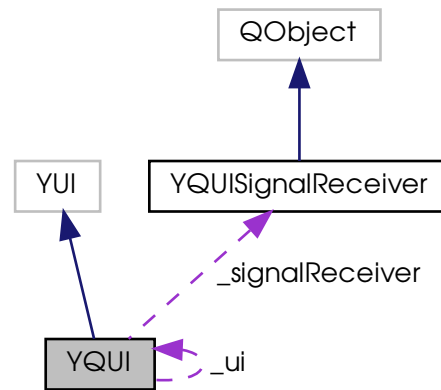
- [/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQTree.h](#)
- [/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQTree.cc](#)

## 3.73 YQUI Class Reference

Inheritance diagram for YQUI:



Collaboration diagram for YQUI:



## Public Member Functions

- [YQUI](#) (bool withThreads)  
*Constructor.*
- virtual [~YQUI](#) ()  
*Destructor.*
- void [initUI](#) ()  
*Post-constructor initialization.*
- void [sendEvent](#) (YEvent \*event)  
*Widget event handlers (slots) call this when an event occurred that should be the answer to a UserInput() / PollInput() (etc.) call.*
- bool [eventPendingFor](#) (YWidget \*widget) const  
*Returns 'true' if there is any event pending for the specified widget.*
- YEvent \* [pendingEvent](#) () const  
*Returns the last event that isn't processed yet or 0 if there is none.*
- YEvent \* [consumePendingEvent](#) ()  
*Return the pending event, if there is one, and mark it as "consumed".*
- virtual void [deleteNotify](#) (YWidget \*widget)  
*Notification that a widget is being deleted.*
- bool [fullscreen](#) () const  
*Return 'true' if defaultsize windows should use the full screen.*
- bool [noBorder](#) () const  
*Return 'true' if defaultsize windows should not get window manager borders / frames.*
- bool [fatalError](#) () const  
*Returns 'true' if the UI had a fatal error that requires the application to abort.*
- void [raiseFatalError](#) ()

- Raise a fatal UI error.*

  - int [defaultSize](#) (YUIDimension dim) const

*Returns size for `opt` (defaultsize) dialogs (in one dimension).*
- void [makeScreenShot](#) (std::string filename)

*Make a screen shot in .png format and save it to 'filename'.*
- virtual YEvent \* [runPkgSelection](#) (YWidget \*packageSelector)

*UI-specific runPkgSeleciton method: Start the package selection.*
- void [toggleRecordMacro](#) ()

*Toggle macro recording (activated by Ctrl-Shift-Alt-M): Stop macro recording if it is in progress, open a file selection box and ask for a macro file name to save to and start recording if no recording has been in progress.*
- void [askPlayMacro](#) ()

*Open file selection box and ask for a macro file to play (activated by Ctrl-Shift-Alt-P)*
- void [askSendWidgetID](#) ()

*Open a pop-up dialog to ask the user for a widget ID and then send it with `sendWidgetID()`.*
- virtual void [blockEvents](#) (bool block=true)

*Block (or unblock) events.*
- virtual bool [eventsBlocked](#) () const

*Returns 'true' if events are currently blocked.*
- void [forceUnblockEvents](#) ()

*Force unblocking all events, no matter how many times `blockEvents()` has This returns 0 if there is no pending eventbeen called before.*
- void [busyCursor](#) ()

*Show mouse cursor indicating busy state.*
- void [normalCursor](#) ()

*Show normal mouse cursor not indicating busy status.*
- void [timeoutBusyCursor](#) ()

*Show mouse cursor indicating busy state if the UI is unable to respond to user input for more than a predefined timeout (200 millisec).*
- void [askSaveLogs](#) ()

*Open file selection box and let the user save y2logs to that location.*
- void [askConfigureLogging](#) ()

*Open dialog to configure logging.*
- QString [applicationTitle](#) ()

*Returns the application name for the window title (e.g.*
- void [setApplicationTitle](#) (const QString &title)

*Sets the application name for the window title.*
- QIcon [loadIcon](#) (const string &iconName) const

## Static Public Member Functions

- static YQUI \* [ui](#) ()

*Access the global Qt-UI.*
- static YQApplication \* [yqApp](#) ()

*Return the global YApplication object as [YQApplication](#).*
- static void [setTextdomain](#) (const char \*domain)

*Initialize and set a textdomain for gettext()*

## Protected Member Functions

- virtual YWidgetFactory \* [createWidgetFactory](#) ()  
*Create the widget factory that provides all the createXY() methods for standard (mandatory, i.e.*
- virtual YOptionalWidgetFactory \* [createOptionalWidgetFactory](#) ()  
*Create the widget factory that provides all the createXY() methods for optional ("special") widgets and the corresponding hasXYWidget() methods.*
- virtual YApplication \* **createApplication** ()
- void [processCommandLineArgs](#) (int argc, char \*\*argv)  
*Handle command line args.*
- void [probeX11Display](#) (const YCommandLine &cmdLine)  
*Probe the X11 display.*
- void [calcDefaultSize](#) ()  
*Calculate size of opt (defaultsize) dialogs.*
- virtual void [idleLoop](#) (int fd\_ycp)  
*Idle around until fd\_ycp is readable and handle repaints.*
- virtual void [uiThreadDestructor](#) ()  
*Destroy whatever needs to be destroyed within the UI thread.*
- void [receivedYCPCommand](#) ()  
*Notification that a YCP command has been received on fd\_ycp to leave [idleLoop\(\)](#)*
- bool [close](#) ()  
*Application shutdown.*

## Protected Attributes

- QMap< QString, int > **screenShotNo**
- QString **screenShotNameTemplate**
- bool **\_fullscreen**
- bool **\_noborder**
- QSize **\_defaultSize**
- bool **\_do\_exit\_loop**
- bool **\_received\_ycp\_command**
- bool **\_fatalError**
- QTimer \* **\_busyCursorTimer**
- YSimpleEventHandler **\_eventHandler**
- int **\_blockedLevel**
- bool **\_leftHandedMouse**
- bool **\_askedForLeftHandedMouse**
- bool **\_uiInitialized**
- [YUISignalReceiver](#) \* **\_signalReceiver**
- QString **\_applicationTitle**
- int **\_ui\_argc**

## Static Protected Attributes

- static [YUI](#) \* **\_ui** = 0

## Friends

- class **YQUISignalReceiver**

### 3.73.1 Detailed Description

Definition at line 62 of file [YQUI.h](#).

### 3.73.2 Member Function Documentation

#### 3.73.2.1 applicationTitle()

```
QString YQUI::applicationTitle ( ) [inline]
```

Returns the application name for the window title (e.g.

"YaST2@hostname")

Definition at line 283 of file [YQUI.h](#).

#### 3.73.2.2 askConfigureLogging()

```
void YQUI::askConfigureLogging ( )
```

Open dialog to configure logging.

(Shift-F7)

Definition at line 265 of file [YQUI\\_builtins.cc](#).

### 3.73.2.3 askSaveLogs()

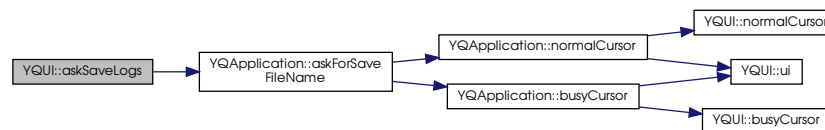
```
void YQUI::askSaveLogs ( )
```

Open file selection box and let the user save y2logs to that location.

(Shift-F8)

Definition at line 206 of file [YQUI\\_builtins.cc](#).

Here is the call graph for this function:



### 3.73.2.4 blockEvents()

```
void YQUI::blockEvents (
    bool block = true ) [virtual]
```

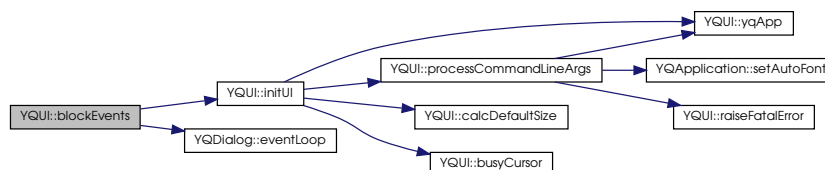
Block (or unblock) events.

If events are blocked, any event sent should be ignored until events are unblocked again.

Reimplemented from YUI.

Definition at line 503 of file [YQUI.cc](#).

Here is the call graph for this function:



#### 3.73.2.5 consumePendingEvent()

```
YEvent* YQUI::consumePendingEvent ( ) [inline]
```

Return the pending event, if there is one, and mark it as "consumed".

This returns 0 if there is no pending event.

Definition at line 155 of file [YQUI.h](#).

#### 3.73.2.6 createOptionalWidgetFactory()

```
YOptionalWidgetFactory * YQUI::createOptionalWidgetFactory ( ) [protected], [virtual]
```

Create the widget factory that provides all the createXY() methods for optional ("special") widgets and the corresponding hasXYWidget() methods.

Reimplemented from YUI.

Definition at line 362 of file [YQUI.cc](#).

#### 3.73.2.7 createWidgetFactory()

```
YWidgetFactory * YQUI::createWidgetFactory ( ) [protected], [virtual]
```

Create the widget factory that provides all the createXY() methods for standard (mandatory, i.e. non-optional) widgets.

Reimplemented from YUI.

Definition at line 351 of file [YQUI.cc](#).

#### 3.73.2.8 deleteNotify()

```
void YQUI::deleteNotify (
    YWidget * widget ) [virtual]
```

Notification that a widget is being deleted.

Reimplemented from YUI.

Definition at line 588 of file [YQUI.cc](#).

### 3.73.2.9 eventsBlocked()

```
bool YQUI::eventsBlocked ( ) const [virtual]
```

Returns 'true' if events are currently blocked.

Reimplemented from YUI.

Definition at line 545 of file [YQUI.cc](#).

### 3.73.2.10 idleLoop()

```
void YQUI::idleLoop (
    int fd_ycp ) [protected], [virtual]
```

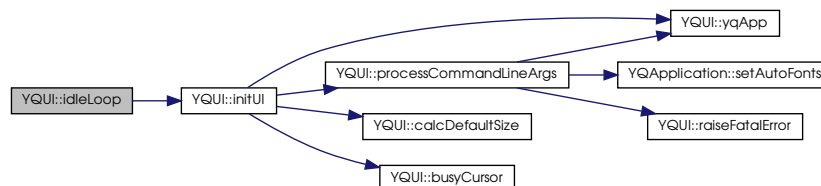
Idle around until fd\_ycp is readable and handle repaints.

This is only used when a separate ui thread is running.

Reimplemented from YUI.

Definition at line 430 of file [YQUI.cc](#).

Here is the call graph for this function:





3.73.2.11 `initUI()`

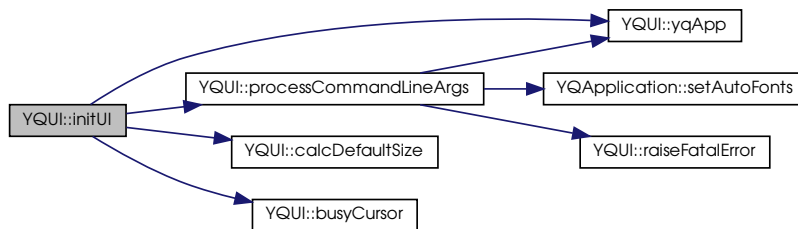
```
void YQUI::initUI ( )
```

Post-constructor initialization.

If running with threads, this has to be called in the UI thread. Any subsequent calls will do nothing.

Definition at line 122 of file [YQUI.cc](#).

Here is the call graph for this function:

3.73.2.12 `makeScreenShot()`

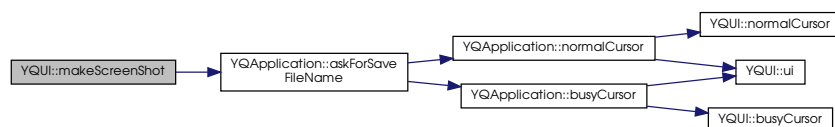
```
void YQUI::makeScreenShot (
    std::string filename )
```

Make a screen shot in .png format and save it to 'filename'.

Opens a file selection box if 'filename' is empty.

Definition at line 92 of file [YQUI\\_builtins.cc](#).

Here is the call graph for this function:



### 3.73.2.13 pendingEvent()

```
YEvent* YQUI::pendingEvent ( ) const [inline]
```

Returns the last event that isn't processed yet or 0 if there is none.

The Qt UI keeps track of only one single (the last one) event.

Definition at line 148 of file [YQUI.h](#).

### 3.73.2.14 probeX11Display()

```
void YQUI::probeX11Display (
    const YCommandLine & cmdLine ) [protected]
```

Probe the X11 display.

Throw exception upon failure. A "-display" command line argument is taken into account.

Definition at line 582 of file [YQUI.cc](#).

### 3.73.2.15 raiseFatalError()

```
void YQUI::raiseFatalError ( ) [inline]
```

Raise a fatal UI error.

It will be delivered when it is safe to do so. The caller should make sure it can continue for some time until the error is delivered.

Definition at line 185 of file [YQUI.h](#).

### 3.73.2.16 runPkgSelection()

```
YEvent * YQUI::runPkgSelection (
    YWidget * packageSelector ) [virtual]
```

UI-specific runPkgSelecion method: Start the package selection.

This implementation does the same as UserInput().

Reimplemented from YUI.

Definition at line 62 of file [YQUI\\_builtins.cc](#).

### 3.73.2.17 sendEvent()

```
void YQUI::sendEvent (
    YEvent * event )
```

Widget event handlers (slots) call this when an event occurred that should be the answer to a `UserInput()` / `PollInput()` (etc.) call.

The UI assumes ownership of the event object that 'event' points to. In particular, it takes care to delete that object.

It is an error to pass 0 for 'event'.

Definition at line 469 of file [YQUI.cc](#).

Here is the call graph for this function:



### 3.73.2.18 uiThreadDestructor()

```
void YQUI::uiThreadDestructor ( ) [protected], [virtual]
```

Destroy whatever needs to be destroyed within the UI thread.

Reimplemented from YUI.

Definition at line 330 of file [YQUI.cc](#).

### 3.73.2.19 yqApp()

```
YQApplication * YQUI::yqApp ( ) [static]
```

Return the global YApplication object as [YQApplication](#).

This will create the Y(Q)Application upon the first call and return a pointer to the one and only (singleton) Y(Q)Application upon each subsequent call. This may throw exceptions if the Y(Q)Application cannot be created.

Definition at line 257 of file [YQUI.cc](#).

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

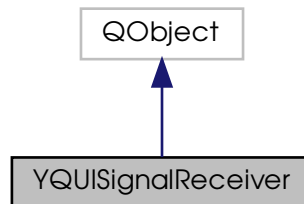
- `/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQUI.h`
- `/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQUI.cc`
- `/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQUI_builtins.cc`

### 3.74 YQUISignalReceiver Class Reference

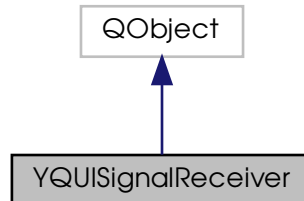
Helper class that acts as a Qt signal receiver for [YQUI](#).

```
#include <YQUI.h>
```

Inheritance diagram for YQUISignalReceiver:



Collaboration diagram for YQUISignalReceiver:



#### Public Slots

- void **slotBusyCursor** ()
- void **slotReceivedYCPCommand** ()

#### 3.74.1 Detailed Description

Helper class that acts as a Qt signal receiver for [YQUI](#).

[YQUI](#) itself cannot be a QObject to avoid problems with starting up the UI with multiple threads.

Definition at line [377](#) of file [YQUI.h](#).

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

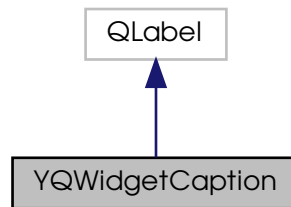
- /usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQUI.h
- /usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQUI.cc

## 3.75 YQWidgetCaption Class Reference

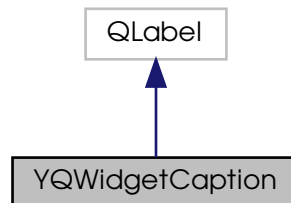
Helper class for captions (labels) above a widget: Takes care of hiding itself when its text is empty and showing when its text becomes non-empty.

```
#include <YQWidgetCaption.h>
```

Inheritance diagram for YQWidgetCaption:



Collaboration diagram for YQWidgetCaption:



### Public Member Functions

- [YQWidgetCaption](#) (QWidget \*parent, const std::string &text)

*Constructors.*

- **YQWidgetCaption** (QWidget \*parent, const QString &text)
- virtual [~YQWidgetCaption](#) ()

*Destructor.*

- virtual void [setText](#) (const std::string &newText)

*Change the text and handle visibility: If the new text is empty, hide this widget.*

- virtual void **setText** (const QString &newText)

### 3.75.1 Detailed Description

Helper class for captions (labels) above a widget: Takes care of hiding itself when its text is empty and showing when its text becomes non-empty.

Definition at line 38 of file [YQWidgetCaption.h](#).

### 3.75.2 Constructor & Destructor Documentation

#### 3.75.2.1 YQWidgetCaption()

```
YQWidgetCaption::YQWidgetCaption (
    QWidget * parent,
    const std::string & text )
```

Constructors.

If the text is empty, the widget will be created, but hidden right away. It can later be made visible again with [setText\(\)](#) with a non-empty text.

Definition at line 33 of file [YQWidgetCaption.cc](#).

### 3.75.3 Member Function Documentation

#### 3.75.3.1 setText()

```
void YQWidgetCaption::setText (
    const std::string & newText ) [virtual]
```

Change the text and handle visibility: If the new text is empty, hide this widget.

If the new text is non-empty, show this widget.

Definition at line 56 of file [YQWidgetCaption.cc](#).

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

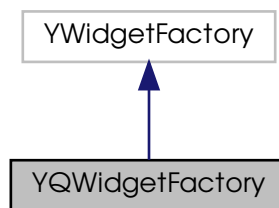
- [/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQWidgetCaption.h](#)
- [/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQWidgetCaption.cc](#)

## 3.76 YQWidgetFactory Class Reference

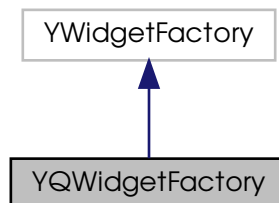
Concrete widget factory for mandatory widgets.

```
#include <YQWidgetFactory.h>
```

Inheritance diagram for YQWidgetFactory:



Collaboration diagram for YQWidgetFactory:



### Public Member Functions

- virtual [YQDialog](#) \* **createDialog** (YDialogType dialogType, YDialogColorMode colorMode=YDialogNormalColor)
- virtual [YQLayoutBox](#) \* **createLayoutBox** (YWidget \*parent, YUIDimension dim)
- virtual [YQButtonBox](#) \* **createButtonBox** (YWidget \*parent)
- virtual [YQPushButton](#) \* **createPushButton** (YWidget \*parent, const std::string &label)
- virtual [YQLabel](#) \* **createLabel** (YWidget \*parent, const std::string &text, bool isHeading=false, bool isOutput←Field=false)
- virtual [YQInputField](#) \* **createInputField** (YWidget \*parent, const std::string &label, bool passwordMode=false)
- virtual [YQCheckBox](#) \* **createCheckBox** (YWidget \*parent, const std::string &label, bool isChecked=false)

- virtual [YQRadioButton](#) \* **createRadioButton** (YWidget \*parent, const std::string &label, bool isChecked=false)
- virtual [YQComboBox](#) \* **createComboBox** (YWidget \*parent, const std::string &label, bool editable=false)
- virtual [YQSelectionBox](#) \* **createSelectionBox** (YWidget \*parent, const std::string &label)
- virtual [YQTree](#) \* **createTree** (YWidget \*parent, const std::string &label, bool multiselection=false, bool recursiveSelection=false)
- virtual [YQTable](#) \* **createTable** (YWidget \*parent, YTableHeader \*header, bool multiSelection=false)
- virtual [YQProgressBar](#) \* **createProgressBar** (YWidget \*parent, const std::string &label, int maxValue=100)
- virtual [YQRichText](#) \* **createRichText** (YWidget \*parent, const std::string &text=std::string(), bool plainText↵ Mode=false)
- virtual [YQIntField](#) \* **createIntField** (YWidget \*parent, const std::string &label, int minVal, int maxVal, int initialVal)
- virtual [YQMenuButton](#) \* **createMenuButton** (YWidget \*parent, const std::string &label)
- virtual [YQMultiLineEdit](#) \* **createMultiLineEdit** (YWidget \*parent, const std::string &label)
- virtual [YQImage](#) \* **createImage** (YWidget \*parent, const std::string &imageFileName, bool animated=false)
- virtual [YQLogView](#) \* **createLogView** (YWidget \*parent, const std::string &label, int visibleLines, int stored↵ Lines=0)
- virtual [YQMultiSelectionBox](#) \* **createMultiSelectionBox** (YWidget \*parent, const std::string &label)
- virtual [YQBusyIndicator](#) \* **createBusyIndicator** (YWidget \*parent, const std::string &label, int maxValue=100)
- virtual YPackageSelector \* **createPackageSelector** (YWidget \*parent, long modeFlags=0)
- virtual YWidget \* **createPkgSpecial** (YWidget \*parent, const std::string &name)
- virtual [YQSpacing](#) \* **createSpacing** (YWidget \*parent, YUIDimension dim, bool stretchable=false, YLayout↵ Size\_t size=0.0)
- virtual [YQEmpty](#) \* **createEmpty** (YWidget \*parent)
- virtual [YQAlignment](#) \* **createAlignment** (YWidget \*parent, YAlignmentType horAlignment, YAlignmentType vertAlignment)
- virtual [YQSquash](#) \* **createSquash** (YWidget \*parent, bool horSquash, bool vertSquash)
- virtual [YQFrame](#) \* **createFrame** (YWidget \*parent, const std::string &label)
- virtual [YQCheckBoxFrame](#) \* **createCheckBoxFrame** (YWidget \*parent, const std::string &label, bool checked)
- virtual [YQRadioButtonGroup](#) \* **createRadioButtonGroup** (YWidget \*parent)
- virtual [YQReplacePoint](#) \* **createReplacePoint** (YWidget \*parent)

## Protected Member Functions

- [YQWidgetFactory](#) ()  
*Constructor.*
- virtual [~YQWidgetFactory](#) ()  
*Destructor.*

## Friends

- class **YQUI**

### 3.76.1 Detailed Description

Concrete widget factory for mandatory widgets.

Definition at line 68 of file [YQWidgetFactory.h](#).



### 3.76.2 Constructor & Destructor Documentation

#### 3.76.2.1 YQWidgetFactory()

```
YQWidgetFactory::YQWidgetFactory ( ) [protected]
```

Constructor.

Use `YUI::widgetFactory()` to get the singleton for this class.

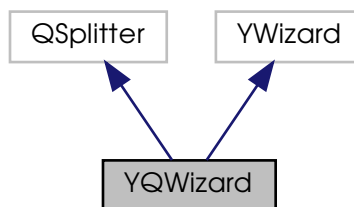
Definition at line 39 of file [YQWidgetFactory.cc](#).

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

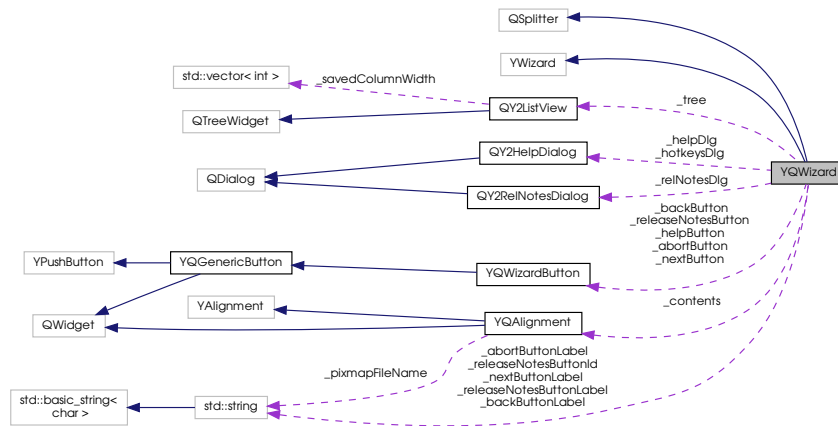
- `/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQWidgetFactory.h`
- `/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQWidgetFactory.cc`

## 3.77 YQWizard Class Reference

Inheritance diagram for YQWizard:



Collaboration diagram for YQWizard:



## Classes

- class [Step](#)  
*Helper class to represent a wizard step internally.*
- class [StepHeading](#)  
*Helper class to represent a wizard step heading internally.*
- class [TreeItem](#)  
*Helper class for wizard tree item.*

## Public Types

- enum **Direction** { **Forward**, **Backward** }

## Public Slots

- void [resizeClientArea](#) ()  
*Adapt the size of the client area (the `ReplacePoint(id (contents))`) to fit in its current space.*
- void [showHelp](#) ()  
*Show the current help text.*
- void [showHotkeys](#) ()  
*Show an overview of the power-user hotkeys.*
- void [showSteps](#) ()  
*Show the current wizard steps, if there are any.*
- void [showTree](#) ()  
*Show the current selection tree in the side panel, if there is any.*

## Signals

- void `backClicked` ()  
*Emitted when the "Back" or "Cancel" button is clicked.*
- void `abortClicked` ()  
*Emitted when the "Abort" button is clicked.*
- void `nextClicked` ()  
*Emitted when the "Next" or "OK" button is clicked.*

## Public Member Functions

- `YQWizard` (YWidget \*parent, const std::string &backButtonLabel, const std::string &abortButtonLabel, const std::string &nextButtonLabel, YWizardMode wizardMode=YWizardMode\_Standard)  
*Constructor.*
- virtual `~YQWizard` ()  
*Destructor.*
- virtual std::string `debugLabel` () const  
*Returns a descriptive label of this dialog instance for debugging.*
- Direction `direction` () const  
*Returns the current direction of wizard operations - going forward or going backward.*
- virtual `YQWizardButton` \* `backButton` () const  
*Return internal widgets.*
- virtual `YQWizardButton` \* `abortButton` () const
- virtual `YQWizardButton` \* `nextButton` () const
- virtual YReplacePoint \* `contentsReplacePoint` () const
- virtual void `setButtonLabel` (YPushButton \*button, const std::string &newLabel)  
*Set the label of one of the wizard buttons (`backButton()`, `abortButton()`, `nextButton()`) if that button is non-null.*
- virtual void `setHelpText` (const std::string &helpText)  
*Set the help text.*
- virtual void `setDialogIcon` (const std::string &iconName)  
*Set the dialog icon.*
- virtual void `setDialogTitle` (const std::string &titleText)  
*Set the dialog title shown in window manager's title bar.*
- virtual void `setDialogHeading` (const std::string &headingText)  
*Set the dialog heading.*
- virtual void `addStep` (const std::string &text, const std::string &id)  
*Add a step for the steps panel on the side bar.*
- virtual void `addStepHeading` (const std::string &text)  
*Add a step heading for the steps panel on the side bar.*
- QList< `YQWizard::Step` \* > `stepsList` ()  
*Return list of pointers to steps.*
- void `copySteps` (`YQWizard` \*wizard)  
*Create a copy of given wizard's steps set (names & IDs) Populates `_stepsList` structure of current wizard.*
- virtual void `deleteSteps` ()  
*Delete all steps and step headings from the internal lists.*
- virtual void `setCurrentStep` (const std::string &id)  
*Set the current step.*

- QString [currentStep](#) ()  
*Return QString ID of currently active step.*
- virtual void [updateSteps](#) ()  
*Update the steps display: Reflect the internal steps and heading lists in the layout.*
- virtual void [addTreeItem](#) (const std::string &parentID, const std::string &text, const std::string &id)  
*Add a tree item.*
- virtual void [selectTreeItem](#) (const std::string &id)  
*Select the tree item with the specified ID, if such an item exists.*
- virtual std::string [currentTreeSelection](#) ()  
*Returns the current tree selection or an empty std::string if nothing is selected or there is no tree.*
- virtual void [deleteTreeItems](#) ()  
*Delete all tree items.*
- virtual void [addMenu](#) (const std::string &text, const std::string &id)  
*Add a menu to the menu bar.*
- virtual void [addSubMenu](#) (const std::string &parentMenuID, const std::string &text, const std::string &id)  
*Add a submenu to the menu with ID 'parentMenuID'.*
- virtual void [addMenuEntry](#) (const std::string &parentMenuID, const std::string &text, const std::string &id)  
*Add a menu entry to the menu with ID 'parentMenuID'.*
- virtual void [addMenuSeparator](#) (const std::string &parentMenuID)  
*Add a menu separator to a menu.*
- virtual void [deleteMenus](#) ()  
*Delete all menus and hide the menu bar.*
- virtual void [showReleaseNotesButton](#) (const std::string &label, const std::string &id)  
*Show a "Release Notes" button above the "Help" button in the steps panel with the specified label that will return the specified id to UI::UserInput() when clicked.*
- virtual void [hideReleaseNotesButton](#) ()  
*Hide an existing "Release Notes" button.*
- virtual void [retranslateInternalButtons](#) ()  
*Retranslate internal buttons that are not accessible from the outside:*
- virtual bool [eventFilter](#) (QObject \*obj, QEvent \*ev)  
*Event filter.*
- QWidget \* [workArea](#) () const  
*Return this wizard's work area (the pane right of the side bar).*
- virtual int [preferredWidth](#) ()  
*Preferred width of the widget.*
- virtual int [preferredHeight](#) ()  
*Preferred height of the widget.*
- virtual void [setSize](#) (int newWidth, int newHeight)  
*Set the new size of the widget.*
- bool [isSecondary](#) () const  
*Returns true if the wizard should follow the first wizard with steps.*

## Protected Slots

- void [slotBackClicked](#) ()  
*Internal notification that the "Back" button has been clicked.*
- void [slotAbortClicked](#) ()  
*Internal notification that the "Abort" button has been clicked.*
- void [slotNextClicked](#) ()  
*Internal notification that the "Next" button has been clicked.*
- void [showReleaseNotes](#) ()  
*Propagate button clicked event of release notes button to the application.*
- void [sendTreeEvent](#) (QTreeWidgetItem \*item)  
*Internal notification that [Space] or [Return] has been pressed on a tree item.*
- void [treeSelectionChanged](#) ()  
*Internal notification that the tree selection has changed.*
- void [sendMenuEvent](#) (QAction \*action)  
*Internal notification that a menu item with numeric ID 'numID' has been activated.*

## Protected Member Functions

- void [layoutTitleBar](#) (QWidget \*parent)
- QLayout \* [layoutSideBar](#) (QWidget \*parent)
- void [layoutSideBarButtonBox](#) (QWidget \*parent, QPushButton \*button)
- void [layoutStepsPanel](#) ()
- void [layoutTreePanel](#) ()
- QWidget \* [layoutWorkArea](#) (QWidget \*parent)
- void [layoutClientArea](#) (QWidget \*parent)
- QLayout \* [layoutButtonBox](#) (QWidget \*parent)
- bool [titlesOnTheLeft](#) ()
- void [destroyButtons](#) ()  
*Destroy the button box's buttons.*
- void [updateStepStates](#) ()  
*Update all step - use appropriate icons and colors.*
- void [sendEvent](#) (const std::string &id)  
*Send a wizard event with the specified ID.*
- void [connectNotify](#) (const char \*signal)  
*Notification that a signal is being connected.*
- void [disconnectNotify](#) (const char \*signal)  
*Notification that a signal is being disconnected.*
- void [setButtonLabel](#) (YQWizardButton \*button, const QString &newLabel)  
*Set a button's label.*
- void [enableButton](#) (YQWizardButton \*button, bool enabled)  
*Enable or disable a button.*
- void [setButtonFocus](#) (YQWizardButton \*button)  
*Set the keyboard focus to a button.*
- YQWizard::Step \* [findStep](#) (const QString &id)  
*Find a step with the specified ID.*
- YQWizard::TreeWidgetItem \* [findTreeWidgetItem](#) (const std::string &id)  
*Find a tree item with the specified ID.*

## Protected Attributes

- std::string **\_backButtonLabel**
- std::string **\_abortButtonLabel**
- std::string **\_nextButtonLabel**
- bool **\_stepsEnabled**
- bool **\_stepsRegistered**
- bool **\_treeEnabled**
- bool **\_protectNextButton**
- bool **\_stepsDirty**
- bool **\_sendButtonEvents**
- Direction **\_direction**
- QString **\_currentStepID**
- QString **\_qHelpText**
- QString **\_qHotkeysText**
- [QY2HelpDialog](#) \* **\_helpDlg**
- [QY2HelpDialog](#) \* **\_hotkeysDlg**
- [QY2RelNotesDialog](#) \* **\_relNotesDlg**
- QStackedWidget \* **\_sideBar**
- QWidget \* **\_stepsPanel**
- [YQWizardButton](#) \* **\_releaseNotesButton**
- [YQWizardButton](#) \* **\_helpButton**
- QAction \* **\_helpAction**
- QAction \* **\_hotkeysAction**
- QPushButton \* **\_stepsButton**
- QPushButton \* **\_treeButton**
- QFrame \* **\_treePanel**
- [QY2ListView](#) \* **\_tree**
- QFrame \* **\_workArea**
- QWidget \* **\_clientArea**
- QMenuBar \* **\_menuBar**
- QLabel \* **\_dialogIcon**
- QLabel \* **\_dialogLogo**
- QLabel \* **\_dialogHeading**
- [YQAlignment](#) \* **\_contents**
- [YQWizardButton](#) \* **\_backButton**
- [YQWizardButton](#) \* **\_abortButton**
- [YQWizardButton](#) \* **\_nextButton**
- YReplacePoint \* **\_contentsReplacePoint**
- QList< [YQWizard::Step](#) \* > **\_stepsList**
- QHash< QString, [YQWizard::Step](#) \* > **\_stepsIDs**
- QHash< QString, [YQWizard::TreeItem](#) \* > **\_treeIDs**
- QHash< QString, QMenu \* > **\_menuIDs**
- QHash< QAction \*, std::string > **\_menuEntryIDs**
- QIcon **\_previousWindowIcon**

## Static Protected Attributes

- static std::string **\_releaseNotesButtonId** = ""
- static std::string **\_releaseNotesButtonLabel** = ""

### 3.77.1 Detailed Description

Definition at line 64 of file [YQWizard.h](#).

### 3.77.2 Member Function Documentation

#### 3.77.2.1 addMenu()

```
void YQWizard::addMenu (
    const std::string & text,
    const std::string & id ) [virtual]
```

Add a menu to the menu bar.

If the menu bar is not visible yet, it will be made visible. 'text' is the user-visible text for the menu bar (including keyboard shortcuts marked with '&'), 'id' is the menu ID for later [addMenuEntry\(\)](#) etc. calls.

Implemented from YWizard.

Definition at line 1200 of file [YQWizard.cc](#).

Here is the call graph for this function:



#### 3.77.2.2 addMenuEntry()

```
void YQWizard::addMenuEntry (
    const std::string & parentMenuID,
    const std::string & text,
    const std::string & id ) [virtual]
```

Add a menu entry to the menu with ID 'parentMenuID'.

'id' is what will be returned by `UI::UserInput()` etc. when a user activates this menu entry.

Implemented from YWizard.

Definition at line 1244 of file [YQWizard.cc](#).

### 3.77.2.3 addMenuSeparator()

```
void YQWizard::addMenuSeparator (
    const std::string & parentMenuID ) [virtual]
```

Add a menu separator to a menu.

Implemented from YWizard.

Definition at line [1267](#) of file [YQWizard.cc](#).

### 3.77.2.4 addStep()

```
void YQWizard::addStep (
    const std::string & text,
    const std::string & id ) [virtual]
```

Add a step for the steps panel on the side bar.

This only adds the step to the internal list of steps. The display is only updated upon calling [updateSteps\(\)](#).

Implemented from YWizard.

Definition at line [293](#) of file [YQWizard.cc](#).

### 3.77.2.5 addStepHeading()

```
void YQWizard::addStepHeading (
    const std::string & text ) [virtual]
```

Add a step heading for the steps panel on the side bar.

This only adds the heading to the internal list of steps. The display is only updated upon calling [updateSteps\(\)](#).

Implemented from YWizard.

Definition at line [329](#) of file [YQWizard.cc](#).



### 3.77.2.6 addSubMenu()

```
void YQWizard::addSubMenu (
    const std::string & parentMenuID,
    const std::string & text,
    const std::string & id ) [virtual]
```

Add a submenu to the menu with ID 'parentMenuID'.

Implemented from YWizard.

Definition at line 1220 of file [YQWizard.cc](#).

Here is the call graph for this function:



### 3.77.2.7 addTreeItem()

```
void YQWizard::addTreeItem (
    const std::string & parentID,
    const std::string & text,
    const std::string & id ) [virtual]
```

Add a tree item.

If "parentID" is an empty `std::string`, it will be a root item. 'text' is the text that will be displayed in the tree, 'id' the ID with which this newly created item can be referenced - and that will be returned when the user clicks on a tree item.

Implemented from YWizard.

Definition at line 584 of file [YQWizard.cc](#).

Here is the call graph for this function:



### 3.77.2.8 backButton()

```
virtual YQWizardButton* YQWizard::backButton ( ) const [inline], [virtual]
```

Return internal widgets.

Implemented from YWizard.

Definition at line 112 of file [YQWizard.h](#).

### 3.77.2.9 connectNotify()

```
void YQWizard::connectNotify (
    const char * signal ) [protected]
```

Notification that a signal is being connected.

Reimplemented from QObject.

Definition at line 975 of file [YQWizard.cc](#).

### 3.77.2.10 copySteps()

```
void YQWizard::copySteps (
    YQWizard * wizard )
```

Create a copy of given wizard's steps set (names & IDs) Populates `_stepsList` structure of current wizard.

Definition at line 496 of file [YQWizard.cc](#).

Here is the call graph for this function:



#### 3.77.2.11 currentStep()

```
QString YQWizard::currentStep ( ) [inline]
```

Return QString ID of currently active step.

Definition at line 212 of file [YQWizard.h](#).

#### 3.77.2.12 currentTreeSelection()

```
string YQWizard::currentTreeSelection ( ) [virtual]
```

Returns the current tree selection or an empty std::string if nothing is selected or there is no tree.

Implemented from YWizard.

Definition at line 674 of file [YQWizard.cc](#).

#### 3.77.2.13 debugLabel()

```
string YQWizard::debugLabel ( ) const [virtual]
```

Returns a descriptive label of this dialog instance for debugging.

Reimplemented from YWidget.

Definition at line 1047 of file [YQWizard.cc](#).

#### 3.77.2.14 deleteMenus()

```
void YQWizard::deleteMenus ( ) [virtual]
```

Delete all menus and hide the menu bar.

Implemented from YWizard.

Definition at line 1282 of file [YQWizard.cc](#).

### 3.77.2.15 deleteSteps()

```
void YQWizard::deleteSteps ( ) [virtual]
```

Delete all steps and step headings from the internal lists.

The display is only updated upon calling [updateSteps\(\)](#).

Implemented from YWizard.

Definition at line [528](#) of file [YQWizard.cc](#).

### 3.77.2.16 deleteTreeItems()

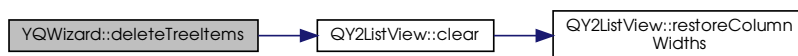
```
void YQWizard::deleteTreeItems ( ) [virtual]
```

Delete all tree items.

Implemented from YWizard.

Definition at line [619](#) of file [YQWizard.cc](#).

Here is the call graph for this function:



### 3.77.2.17 direction()

```
Direction YQWizard::direction ( ) const [inline]
```

Returns the current direction of wizard operations - going forward or going backward.

This can be used to maintain a consistent direction when assigning default buttons to a dialog.

Definition at line [101](#) of file [YQWizard.h](#).

## 3.77.2.18 disconnectNotify()

```
void YQWizard::disconnectNotify (
    const char * signal ) [protected]
```

Notification that a signal is being disconnected.

Reimplemented from QObject.

Definition at line 985 of file [YQWizard.cc](#).

## 3.77.2.19 eventFilter()

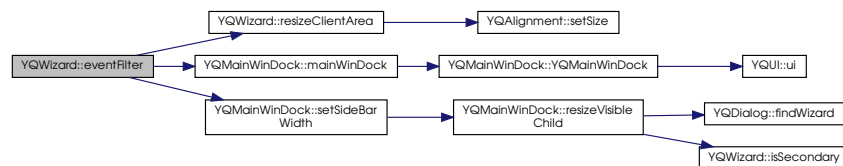
```
bool YQWizard::eventFilter (
    QObject * obj,
    QEvent * ev ) [virtual]
```

Event filter.

Reimplemented from QWidget.

Definition at line 1337 of file [YQWizard.cc](#).

Here is the call graph for this function:



## 3.77.2.20 findStep()

```
YQWizard::Step * YQWizard::findStep (
    const QString & id ) [protected]
```

Find a step with the specified ID.

Returns 0 if there is no such step.

Definition at line 543 of file [YQWizard.cc](#).

### 3.77.2.21 findTreeItem()

```
YQWizard::TreeItem * YQWizard::findTreeItem (
    const std::string & id ) [protected]
```

Find a tree item with the specified ID.

Tree items without IDs cannot be found at all. Returns the item or 0 if no such item found.

Definition at line 629 of file [YQWizard.cc](#).

### 3.77.2.22 hideReleaseNotesButton()

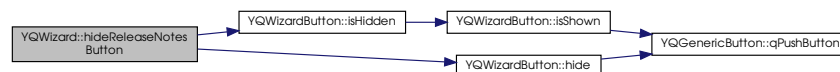
```
void YQWizard::hideReleaseNotesButton ( ) [virtual]
```

Hide an existing "Release Notes" button.

Implemented from YWizard.

Definition at line 1393 of file [YQWizard.cc](#).

Here is the call graph for this function:



### 3.77.2.23 nextClicked

```
void YQWizard::nextClicked ( ) [signal]
```

Emitted when the "Next" or "OK" button is clicked.

Notice: As long as this signal is connected, the wizard will no longer send button events to the UI. Rather, the connected QObject has to take care to propagate those events. This is used in YQPatternSelector, for example.

#### 3.77.2.24 preferredHeight()

```
int YQWizard::preferredHeight ( ) [virtual]
```

Preferred height of the widget.

Reimplemented from YWidget.

Definition at line 1319 of file [YQWizard.cc](#).

#### 3.77.2.25 preferredWidth()

```
int YQWizard::preferredWidth ( ) [virtual]
```

Preferred width of the widget.

Reimplemented from YWidget.

Definition at line 1313 of file [YQWizard.cc](#).

#### 3.77.2.26 retranslateInternalButtons()

```
void YQWizard::retranslateInternalButtons ( ) [virtual]
```

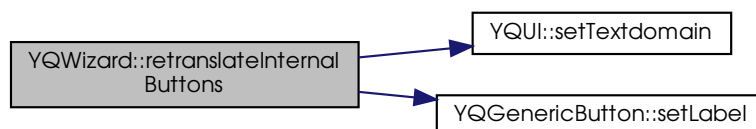
Retranslate internal buttons that are not accessible from the outside:

- [Help]
- [Steps]
- [Tree]

Implemented from YWizard.

Definition at line 1404 of file [YQWizard.cc](#).

Here is the call graph for this function:



### 3.77.2.27 selectTreeItem()

```
void YQWizard::selectTreeItem (
    const std::string & id ) [virtual]
```

Select the tree item with the specified ID, if such an item exists.

Implemented from YWizard.

Definition at line 638 of file [YQWizard.cc](#).

Here is the call graph for this function:



### 3.77.2.28 sendTreeEvent

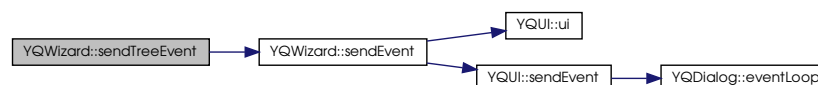
```
void YQWizard::sendTreeEvent (
    QTreeWidgetItem * item ) [protected], [slot]
```

Internal notification that [Space] or [Return] has been pressed on a tree item.

If the item has an ID, that ID will be returned to `UI::UserInput()`.

Definition at line 655 of file [YQWizard.cc](#).

Here is the call graph for this function:





3.77.2.29 `setButtonLabel()`

```
void YQWizard::setButtonLabel (
    YPushButton * button,
    const std::string & newLabel ) [virtual]
```

Set the label of one of the wizard buttons (`backButton()`, `abortButton()`, `nextButton()` ) if that button is non-null.

Implemented from YWizard.

Definition at line 1355 of file [YQWizard.cc](#).

Here is the call graph for this function:

3.77.2.30 `setCurrentStep()`

```
void YQWizard::setCurrentStep (
    const std::string & id ) [virtual]
```

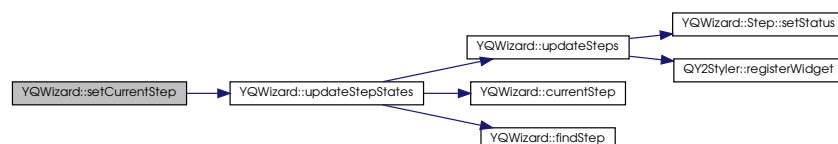
Set the current step.

This also triggers `updateSteps()` if necessary.

Implemented from YWizard.

Definition at line 488 of file [YQWizard.cc](#).

Here is the call graph for this function:



### 3.77.2.31 setDialogHeading()

```
void YQWizard::setDialogHeading (
    const std::string & headingText ) [virtual]
```

Set the dialog heading.

Implemented from YWizard.

Definition at line 1036 of file [YQWizard.cc](#).

### 3.77.2.32 setDialogIcon()

```
void YQWizard::setDialogIcon (
    const std::string & iconName ) [virtual]
```

Set the dialog icon.

An empty icon name clears the current icon.

Implemented from YWizard.

Definition at line 995 of file [YQWizard.cc](#).

### 3.77.2.33 setDialogTitle()

```
void YQWizard::setDialogTitle (
    const std::string & titleText ) [virtual]
```

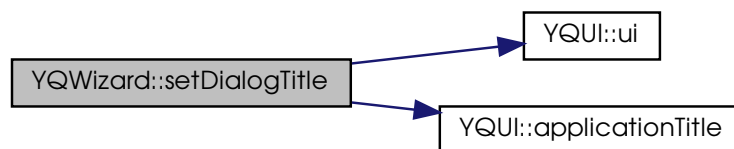
Set the dialog title shown in window manager's title bar.

An empty std::string clears the current text.

Implemented from YWizard.

Definition at line 1025 of file [YQWizard.cc](#).

Here is the call graph for this function:



#### 3.77.2.34 setHelpText()

```
void YQWizard::setHelpText (
    const std::string & helpText ) [virtual]
```

Set the help text.

Implemented from YWizard.

Definition at line 1062 of file [YQWizard.cc](#).

#### 3.77.2.35 setSize()

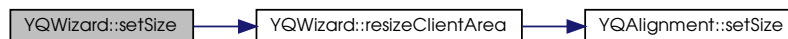
```
void YQWizard::setSize (
    int newWidth,
    int newHeight ) [virtual]
```

Set the new size of the widget.

Reimplemented from YWidget.

Definition at line 1325 of file [YQWizard.cc](#).

Here is the call graph for this function:



#### 3.77.2.36 showHelp

```
void YQWizard::showHelp ( ) [slot]
```

Show the current help text.

This is useful only if it is obscured by any wizard steps, but it can safely be called at any time.

Definition at line 1100 of file [YQWizard.cc](#).

### 3.77.2.37 showHotkeys

```
void YQWizard::showHotkeys ( ) [slot]
```

Show an overview of the power-user hotkeys.

Help text to be shown after pressing Shift-F1 listing the advanced keyboard shortcuts available in the Qt-UI

Definition at line 1116 of file [YQWizard.cc](#).

### 3.77.2.38 showReleaseNotesButton()

```
void YQWizard::showReleaseNotesButton (
    const std::string & label,
    const std::string & id ) [virtual]
```

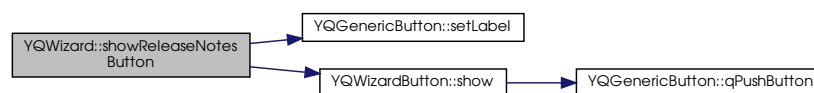
Show a "Release Notes" button above the "Help" button in the steps panel with the specified label that will return the specified id to UI::UserInput() when clicked.

The button (or the wizard) will assume ownership of the id and delete it in the destructor.

Implemented from YWizard.

Definition at line 1372 of file [YQWizard.cc](#).

Here is the call graph for this function:



### 3.77.2.39 showSteps

```
void YQWizard::showSteps ( ) [slot]
```

Show the current wizard steps, if there are any.

If there are none, nothing happens.

Definition at line 1182 of file [YQWizard.cc](#).

#### 3.77.2.40 showTree

```
void YQWizard::showTree ( ) [slot]
```

Show the current selection tree in the side panel, if there is any.

If there is none, nothing happens.

Definition at line 1191 of file [YQWizard.cc](#).

#### 3.77.2.41 stepsList()

```
QList<YQWizard::Step*> YQWizard::stepsList ( ) [inline]
```

Return list of pointers to steps.

Not needed outside [copySteps\(\)](#) function

Definition at line 184 of file [YQWizard.h](#).

#### 3.77.2.42 treeSelectionChanged

```
void YQWizard::treeSelectionChanged ( ) [protected], [slot]
```

Internal notification that the tree selection has changed.

If the currently selected item has an ID, that ID will be returned to UI::UserInput().

Definition at line 667 of file [YQWizard.cc](#).

Here is the call graph for this function:



### 3.77.2.43 updateSteps()

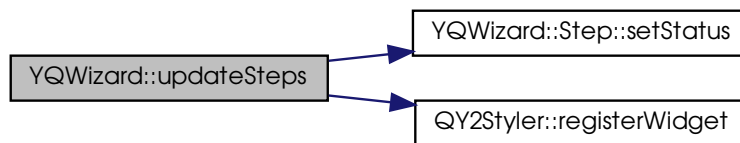
```
void YQWizard::updateSteps ( ) [virtual]
```

Update the steps display: Reflect the internal steps and heading lists in the layout.

Implemented from YWizard.

Definition at line [336](#) of file [YQWizard.cc](#).

Here is the call graph for this function:



### 3.77.2.44 workArea()

```
QWidget* YQWizard::workArea ( ) const [inline]
```

Return this wizard's work area (the pane right of the side bar).

This should not be needed outside of [YQMainWinDock](#).

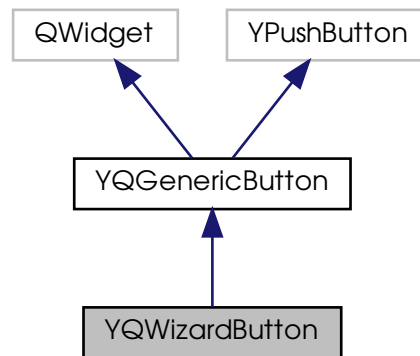
Definition at line [356](#) of file [YQWizard.h](#).

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

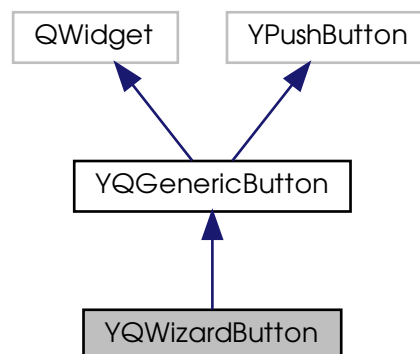
- `/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQWizard.h`
- `/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQWizard.cc`

## 3.78 YQWizardButton Class Reference

Inheritance diagram for YQWizardButton:



Collaboration diagram for YQWizardButton:



### Signals

- void [clicked](#) ()  
*Emitted when the button is clicked or otherwise activated.*

## Public Member Functions

- [YQWizardButton](#) ([YQWizard](#) \*[wizard](#), [QWidget](#) \*[buttonParent](#), const std::string &[label](#))  
*Constructor.*
- virtual [~YQWizardButton](#) ()  
*Destructor.*
- virtual const char \* [widgetClass](#) () const  
*Returns a descriptive name of this widget class for logging, debugging etc.*
- void [hide](#) ()  
*Hide the associated QPushButton.*
- void [show](#) ()  
*Show the associated QPushButton - not this widget itself (!).*
- bool [isShown](#) () const  
*Returns 'true' if the associated QPushButton (!) is shown.*
- bool [isHidden](#) () const  
*Returns 'true' if the associated QPushButton (!) is hidden.*
- virtual int [preferredWidth](#) ()  
*Preferred width of the widget.*
- virtual int [preferredHeight](#) ()  
*Preferred height of the widget.*
- virtual void [setSize](#) (int [newWidth](#), int [newHeight](#))  
*Set the new size of the widget.*
- [YQWizard](#) \* [wizard](#) () const  
*Returns the wizard this wizard button belongs to.*

## Additional Inherited Members

### 3.78.1 Detailed Description

Definition at line 37 of file [YQWizardButton.h](#).

### 3.78.2 Member Function Documentation

#### 3.78.2.1 preferredHeight()

```
int YQWizardButton::preferredHeight ( ) [virtual]
```

Preferred height of the widget.

Reimplemented from [YWidget](#).

Definition at line 104 of file [YQWizardButton.cc](#).



### 3.78.2.2 preferredWidth()

```
int YQWizardButton::preferredWidth ( ) [virtual]
```

Preferred width of the widget.

Reimplemented from YWidget.

Definition at line 95 of file [YQWizardButton.cc](#).

### 3.78.2.3 setSize()

```
void YQWizardButton::setSize (
    int newWidth,
    int newHeight ) [virtual]
```

Set the new size of the widget.

Reimplemented from YWidget.

Definition at line 113 of file [YQWizardButton.cc](#).

### 3.78.2.4 show()

```
void YQWizardButton::show ( )
```

Show the associated QPushButton - not this widget itself (!).

This widget itself will never become visible.

Definition at line 73 of file [YQWizardButton.cc](#).

Here is the call graph for this function:



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

- `/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQWizardButton.h`
- `/usr/src/RPM/BUILD/libyui-qt-2.49.11/src/YQWizardButton.cc`



# Index

- ~Step
  - YQWizard::Step, [57](#)
- ~YQDialog
  - YQDialog, [112](#)
- ~YQSignalBlocker
  - YQSignalBlocker, [232](#)
- activate
  - YQDialog, [113](#)
- activateDefaultButton
  - YQDialog, [113](#)
- add
  - YQMainWinDock, [165](#)
- addItem
  - YQComboBox, [97](#)
  - YQDumbTab, [126](#)
  - YQMultiSelectionBox, [187](#)
  - YQSelectionBox, [225](#)
  - YQTable, [243](#)
- addItems
  - YQSelectionBox, [225](#)
  - YQTable, [243](#)
- addMenu
  - YQWizard, [291](#)
- addMenuEntry
  - YQWizard, [291](#)
- addMenuSeparator
  - YQWizard, [291](#)
- addPage
  - QY2ComboTabWidget, [21](#)
- addRadioButton
  - YQRadioButtonGroup, [214](#)
- addStep
  - YQWizard, [292](#)
- addStepHeading
  - YQWizard, [292](#)
- addSubMenu
  - YQWizard, [292](#)
- addTreeWidgetItem
  - YQWizard, [293](#)
- applicationTitle
  - YQUI, [273](#)
- askConfigureLogging
  - YQUI, [273](#)
- askForExistingDirectory
  - YQApplication, [67](#)
- askForExistingFile
  - YQApplication, [68](#)
- askForSaveFileName
  - YQApplication, [69](#)
- askSaveLogs
  - YQUI, [273](#)
- autoFonts
  - YQApplication, [69](#)
- backButton
  - YQWizard, [293](#)
- beep
  - YQApplication, [70](#)
- blockEvents
  - YQUI, [274](#)
- buildStyleSheet
  - QY2Styler, [49](#)
- buildStyleSheetFromFile
  - QY2Styler, [49](#)
- BusyBar, [9](#)
- busyCursor
  - YQApplication, [70](#)
- cellChanged
  - YQTable, [244](#)
- center
  - YQDialog, [114](#)
- changed
  - YQInputField, [145](#)
- closeEvent
  - YQMainWinDock, [166](#)
- columnWidthChanged
  - QY2ListView, [37](#)
- connectNotify
  - YQWizard, [294](#)
- consumePendingEvent
  - YQUI, [274](#)
- copySteps
  - YQWizard, [294](#)
- createGraph
  - YQGraphPluginStub, [139](#)
- createOptionalWidgetFactory
  - YQUI, [275](#)
- createPackageSelector
  - YQPackageSelectorPluginStub, [197](#)
- createWidgetFactory

- YQUI, 275
- currentChanged
  - QY2ComboTabWidget, 21
- currentFont
  - YQApplication, 70
- currentItem
  - YQMultiSelectionBox, 187
  - YQTree, 261
- currentStep
  - YQWizard, 294
- currentTreeSelection
  - YQWizard, 295
- debugLabel
  - YQWizard, 295
- defaultTextColor
  - YQBarGraph, 79
- deleteAllItems
  - YQComboBox, 97
  - YQDumbTab, 127
  - YQMultiSelectionBox, 187
  - YQSelectionBox, 226
  - YQTable, 244
  - YQTree, 261
- deleteMenus
  - YQWizard, 295
- deleteNotify
  - YQUI, 275
- deleteSteps
  - YQWizard, 295
- deleteTreeItems
  - YQWizard, 296
- deselectAllItems
  - YQDumbTab, 127
  - YQMultiSelectionBox, 188
  - YQSelectionBox, 226
  - YQTable, 245
  - YQTree, 261
- deviceName
  - QY2DiskUsageListItem, 27
- deviceUnits
  - YQApplication, 71
- dialog
  - YQGenericButton, 135
- direction
  - YQWizard, 296
- disconnectNotify
  - YQWizard, 296
- displayLogText
  - YQLogView, 161
- doUpdate
  - YQBarGraph, 79
  - YQMultiProgressMeter, 179
- drawSegment
  - YQMultiProgressMeter, 179
- enforceMaxInputLength
  - YQMultiLineEdit, 174
- ensureOnlyOneDefaultButton
  - YQDialog, 114
- eventFilter
  - YQGenericButton, 135
  - YQRichText, 220
  - YQSelectionBox, 226
  - YQWizard, 297
- eventsBlocked
  - YQUI, 275
- findItem
  - YQMultiSelectionBox, 188
- findStep
  - YQWizard, 297
- findTreeItem
  - YQWizard, 297
- findWizard
  - YQDialog, 115
- freeSize
  - QY2DiskUsageListItem, 27
- gettingFocus
  - YQDialog, 115
- glyph
  - YQApplication, 71
- headingFont
  - YQApplication, 72
- hideReleaseNotesButton
  - YQWizard, 298
- highlight
  - YQDialog, 115
- hit
  - YQPushButton, 207
- idleLoop
  - YQUI, 276
- initUI
  - YQUI, 276
- isShownAsDefault
  - YQGenericButton, 136
- key
  - YQTreeItem, 268
- keyPressEvent
  - YQDialog, 116
- layoutUnits
  - YQApplication, 72
- loadAlternateStyleSheet
  - QY2Styler, 50
- loadDefaultStyleSheet

- QY2Styler, [50](#)
- loadStyleSheet
  - QY2Styler, [51](#)
- losingFocus
  - YQDialog, [116](#)
- mainWinDock
  - YQMainWinDock, [166](#)
- makeScreenShot
  - YQApplication, [73](#)
  - YQUI, [277](#)
- maybeLeftHandedUser
  - YQApplication, [73](#)
- minimumSizeHint
  - QY2ListView, [38](#)
- mouseDoubleClickEvent
  - QY2ListView, [38](#)
- mousePressEvent
  - QY2ListView, [38](#)
- mouseReleaseEvent
  - QY2ListView, [38](#)
- moveChild
  - YQAlignment, [62](#)
  - YQButtonBox, [86](#)
  - YQLayoutBox, [158](#)
- MyTextEdit, [11](#)
- name
  - QY2DiskUsageListItem, [28](#)
- nextClicked
  - YQWizard, [298](#)
- normalCursor
  - YQApplication, [74](#)
- openContextMenu
  - YQApplication, [74](#)
- openInternal
  - YQDialog, [117](#)
- operator<
  - QY2DiskUsageListItem, [28](#)
  - QY2ListViewItem, [42](#)
- paintEvent
  - YQBarGraph, [79](#)
  - YQMainWinDock, [167](#)
  - YQMultiProgressMeter, [180](#)
- paintPercentageBar
  - QY2DiskUsageListItem, [29](#)
- pendingEvent
  - YQUI, [277](#)
- pickAutoFonts
  - YQApplication, [74](#)
- pollEventInternal
  - YQDialog, [117](#)
- preferredHeight
  - YQBarGraph, [79](#)
  - YQBusyIndicator, [82](#)
  - YQCheckBox, [88](#)
  - YQCheckBoxFrame, [92](#)
  - YQComboBox, [97](#)
  - YQContextMenu, [103](#)
  - YQDateField, [106](#)
  - YQDialog, [118](#)
  - YQDownloadProgress, [123](#)
  - YQDumbTab, [127](#)
  - YQFrame, [132](#)
  - YQImage, [142](#)
  - YQInputField, [145](#)
  - YQIntField, [151](#)
  - YQLabel, [155](#)
  - YQLogView, [161](#)
  - YQMenuButton, [170](#)
  - YQMultiLineEdit, [174](#)
  - YQMultiProgressMeter, [181](#)
  - YQMultiSelectionBox, [188](#)
  - YQPartitionSplitter, [199](#)
  - YQProgressBar, [203](#)
  - YQPushButton, [207](#)
  - YQRadioButton, [210](#)
  - YQRichText, [221](#)
  - YQSelectionBox, [227](#)
  - YQSlider, [235](#)
  - YQTable, [245](#)
  - YQTimeField, [253](#)
  - YQTimezoneSelector, [257](#)
  - YQTree, [262](#)
  - YQWizard, [298](#)
  - YQWizardButton, [308](#)
- preferredWidth
  - YQBarGraph, [80](#)
  - YQBusyIndicator, [82](#)
  - YQCheckBox, [88](#)
  - YQCheckBoxFrame, [92](#)
  - YQComboBox, [98](#)
  - YQContextMenu, [103](#)
  - YQDateField, [106](#)
  - YQDialog, [118](#)
  - YQDownloadProgress, [123](#)
  - YQDumbTab, [127](#)
  - YQFrame, [132](#)
  - YQImage, [142](#)
  - YQInputField, [146](#)
  - YQIntField, [151](#)
  - YQLabel, [155](#)
  - YQLogView, [162](#)
  - YQMenuButton, [170](#)
  - YQMultiLineEdit, [174](#)
  - YQMultiProgressMeter, [182](#)
  - YQMultiSelectionBox, [189](#)

- YQPartitionSplitter, 199
- YQProgressBar, 203
- YQPushButton, 207
- YQRadioButton, 210
- YQRichText, 221
- YQSelectionBox, 227
- YQSlider, 235
- YQTable, 245
- YQTimeField, 253
- YQTimezoneSelector, 257
- YQTree, 262
- YQWizard, 299
- YQWizardButton, 308
- probeX11Display
  - YQUI, 278
- processUrls
  - QY2Styler, 52
- QY2CharValidator, 12
  - QY2CharValidator, 13
  - setValidChars, 13
  - validate, 14
- QY2CheckListItem, 15
  - serial, 16
  - setBackgroundColor, 16
  - setTextColor, 17
  - toolTip, 17
  - updateData, 17
  - updateStatus, 18
- QY2ComboTabWidget, 18
  - addPage, 21
  - currentChanged, 21
  - QY2ComboTabWidget, 20
  - showPage, 21
  - showPageIndex, 21
- QY2DiskUsageList, 22
  - QY2DiskUsageList, 24
- QY2DiskUsageListItem, 25
  - deviceName, 27
  - freeSize, 27
  - name, 28
  - operator<, 28
  - paintPercentageBar, 29
  - QY2DiskUsageListItem, 27
  - totalSize, 30
  - updateData, 30
  - updateStatus, 30
  - usedPercent, 31
  - usedSize, 32
- QY2DiskUsagePercentageItem, 32
- QY2HelpDialog, 33
- QY2ListView, 35
  - columnWidthChanged, 37
  - minimumSizeHint, 38
  - mouseDoubleClickEvent, 38
  - mousePressEvent, 38
  - mouseReleaseEvent, 38
  - selectSomething, 39
  - toolTip, 39
  - updateItemData, 39
  - updateItemStates, 40
- QY2ListViewItem, 41
  - operator<, 42
  - serial, 42
  - toolTip, 43
  - updateData, 43
  - updateStatus, 43
- QY2RelNotesDialog, 44
- QY2StyleEditor, 45
- QY2Styler, 46
  - buildStyleSheet, 49
  - buildStyleSheetFromFile, 49
  - loadAlternateStyleSheet, 50
  - loadDefaultStyleSheet, 50
  - loadStyleSheet, 51
  - processUrls, 52
  - QY2Styler, 48
  - registerChildWidget, 52
  - registerWidget, 53
  - setAlternateStyleSheet, 53
  - setDefaultStyleSheet, 53
  - setStyleSheet, 54
  - styleSheetExists, 55
  - unregisterWidget, 55
- raiseFatalError
  - YQUI, 278
- rebuildMenuTree
  - YQContextMenu, 104
  - YQMenuButton, 171
- rebuildTree
  - YQTree, 262
- registerChildWidget
  - QY2Styler, 52
- registerWidget
  - QY2Styler, 53
- remove
  - YQMainWinDock, 167
- resizeEvent
  - YQMainWinDock, 168
- retranslateInternalButtons
  - YQWizard, 299
- returnDelayed
  - YQSelectionBox, 228
- returnImmediately
  - YQSelectionBox, 228
- runPkgSelection
  - YQUI, 278

- selectItem
  - YQDumbTab, 128
  - YQMultiSelectionBox, 189
  - YQSelectionBox, 228
  - YQTable, 246
  - YQTree, 263
- selectSomething
  - QY2ListView, 39
- selectTreeItem
  - YQWizard, 299
- sendEvent
  - YQUI, 278
- sendTreeEvent
  - YQWizard, 300
- serial
  - QY2CheckListItem, 16
  - QY2ListViewItem, 42
- setAlive
  - YQBusyIndicator, 83
- setAlternateStyleSheet
  - QY2Styler, 53
- setApplicationIcon
  - YQApplication, 75
- setApplicationTitle
  - YQApplication, 75
- setAutoScale
  - YQImage, 142
- setAutoScrollDown
  - YQRichText, 221
- setBackgroundColor
  - QY2CheckListItem, 16
- setBackgroundPixmap
  - YQAlignment, 63
- setButtonLabel
  - YQWizard, 300
- setCurrentItem
  - YQMultiSelectionBox, 189
- setCurrentStep
  - YQWizard, 301
- setDefaultButton
  - YQDialog, 119
- setDefaultStyleSheet
  - QY2Styler, 53
- setDialogHeading
  - YQWizard, 301
- setDialogIcon
  - YQWizard, 302
- setDialogTitle
  - YQWizard, 302
- setEnabled
  - YQAlignment, 63
  - YQBarGraph, 80
  - YQBusyIndicator, 83
  - YQButtonBox, 86
  - YQCheckBox, 89
  - YQCheckBoxFrame, 93
  - YQComboBox, 98
  - YQDateField, 107
  - YQDialog, 119
  - YQDownloadProgress, 123
  - YQDumbTab, 128
  - YQFrame, 132
  - YQGenericButton, 136
  - YQInputField, 146
  - YQIntField, 151
  - YQLabel, 155
  - YQLayoutBox, 158
  - YQLogView, 162
  - YQMenuButton, 171
  - YQMultiLineEdit, 175
  - YQMultiProgressMeter, 182
  - YQMultiSelectionBox, 190
  - YQPartitionSplitter, 200
  - YQProgressBar, 203
  - YQRadioButton, 210
  - YQRadioButtonGroup, 214
  - YQReplacePoint, 218
  - YQRichText, 222
  - YQSelectionBox, 229
  - YQSlider, 235
  - YQSquash, 240
  - YQTable, 246
  - YQTimeField, 253
  - YQTree, 263
- setExpectedSize
  - YQDownloadProgress, 123
- setFilename
  - YQDownloadProgress, 124
- setHelpText
  - YQWizard, 302
- setIcon
  - YQGenericButton, 136
- setImage
  - YQImage, 142
- setInputMaxLength
  - YQComboBox, 98
  - YQInputField, 146
  - YQMultiLineEdit, 175
- setKeepSorting
  - YQTable, 246
- setKeyboardFocus
  - YQCheckBox, 89
  - YQCheckBoxFrame, 93
  - YQDateField, 107
  - YQInputField, 146
  - YQIntField, 151
  - YQRadioButton, 211
  - YQRichText, 222

- YQSelectionBox, [229](#)
- YQTable, [247](#)
- YQTimeField, [253](#)
- setLabel
  - YQBusyIndicator, [83](#)
  - YQCheckBox, [89](#)
  - YQCheckBoxFrame, [93](#)
  - YQComboBox, [98](#)
  - YQDateField, [107](#)
  - YQDownloadProgress, [124](#)
  - YQGenericButton, [137](#)
  - YQInputField, [147](#)
  - YQIntField, [151](#)
  - YQLogView, [162](#)
  - YQMenuButton, [171](#)
  - YQMultiLineEdit, [175](#)
  - YQMultiSelectionBox, [190](#)
  - YQProgressBar, [204](#)
  - YQRadioButton, [211](#)
  - YQSelectionBox, [229](#)
  - YQSlider, [235](#)
  - YQTimeField, [254](#)
  - YQTree, [264](#)
- setLangFonts
  - YQApplication, [75](#)
- setLanguage
  - YQApplication, [76](#)
- setMargin
  - YQMultiProgressMeter, [182](#)
- setOpen
  - YQTreeItem, [268](#)
- setPlainTextMode
  - YQRichText, [222](#)
- setShortcut
  - YQGenericButton, [137](#)
- setSize
  - YQAlignment, [63](#)
  - YQBarGraph, [80](#)
  - YQBusyIndicator, [84](#)
  - YQButtonBox, [86](#)
  - YQCheckBox, [89](#)
  - YQCheckBoxFrame, [94](#)
  - YQComboBox, [99](#)
  - YQContextMenu, [104](#)
  - YQDateField, [107](#)
  - YQDialog, [120](#)
  - YQDownloadProgress, [124](#)
  - YQDumbTab, [128](#)
  - YQEmpty, [130](#)
  - YQFrame, [132](#)
  - YQImage, [143](#)
  - YQInputField, [147](#)
  - YQIntField, [152](#)
  - YQLabel, [155](#)
  - YQLayoutBox, [159](#)
  - YQLogView, [162](#)
  - YQMenuButton, [172](#)
  - YQMultiLineEdit, [176](#)
  - YQMultiProgressMeter, [182](#)
  - YQMultiSelectionBox, [191](#)
  - YQPartitionSplitter, [200](#)
  - YQProgressBar, [204](#)
  - YQPushButton, [208](#)
  - YQRadioButton, [211](#)
  - YQRadioButtonGroup, [214](#)
  - YQReplacePoint, [218](#)
  - YQRichText, [222](#)
  - YQSelectionBox, [230](#)
  - YQSlider, [236](#)
  - YQSpacing, [239](#)
  - YQSquash, [240](#)
  - YQTable, [247](#)
  - YQTimeField, [254](#)
  - YQTimezoneSelector, [257](#)
  - YQTree, [264](#)
  - YQWizard, [303](#)
  - YQWizardButton, [309](#)
- setSource
  - YQTextBrowser, [251](#)
- setSpacing
  - YQMultiProgressMeter, [183](#)
- setStyleSheet
  - QY2Styler, [54](#)
- setText
  - YQComboBox, [99](#)
  - YQLabel, [156](#)
  - YQWidgetCaption, [282](#)
- setTextColor
  - QY2CheckListItem, [17](#)
- setTimeout
  - YQBusyIndicator, [84](#)
- setTriThickness
  - YQMultiProgressMeter, [183](#)
- setUseBoldFont
  - YQCheckBox, [90](#)
  - YQLabel, [156](#)
  - YQRadioButton, [211](#)
- setValidChars
  - QY2CharValidator, [13](#)
  - YQComboBox, [99](#)
  - YQInputField, [147](#)
- setValue
  - YQCheckBox, [90](#)
  - YQCheckBoxFrame, [94](#)
  - YQDateField, [108](#)
  - YQInputField, [148](#)
  - YQMultiLineEdit, [176](#)
  - YQPartitionSplitter, [200](#)



- YQProgressBar, 204
- YQRadioButton, 212
- YQRichText, 223
- YQTimeField, 254
- setValueInternal
  - YQIntField, 152
  - YQSlider, 236
- shortcutChanged
  - YQDumbTab, 128
- show
  - YQMainWinDock, 168
  - YQWizardButton, 309
- showAsDefault
  - YQGenericButton, 137
- showChild
  - YQReplacePoint, 218
- showHelp
  - YQWizard, 303
- showHotkeys
  - YQWizard, 303
- showPage
  - QY2ComboTabWidget, 21
- showPageIndex
  - QY2ComboTabWidget, 21
- showReleaseNotesButton
  - YQWizard, 304
- showSteps
  - YQWizard, 304
- showTree
  - YQWizard, 304
- slotActivated
  - YQTree, 264
- slotContextMenu
  - YQTable, 247
  - YQTree, 265
- slotItemChanged
  - YQTree, 265
- slotSelected
  - YQComboBox, 100
- slotSelectionChanged
  - YQSelectionBox, 230
  - YQTree, 266
- stepsList
  - YQWizard, 305
- styleSheetExists
  - QY2Styler, 55
- text
  - YQComboBox, 100
- textChanged
  - YQComboBox, 101
- timeout
  - YQBusyIndicator, 84
- toolTip
  - QY2CheckListItem, 17
  - QY2ListView, 39
  - QY2ListViewItem, 43
- totalSize
  - QY2DiskUsageListItem, 30
- treeSelectionChanged
  - YQWizard, 305
- triThickness
  - YQMultiProgressMeter, 184
- uiThreadDestructor
  - YQUI, 279
- unregisterWidget
  - QY2Styler, 55
- updateData
  - QY2CheckListItem, 17
  - QY2DiskUsageListItem, 30
  - QY2ListViewItem, 43
- updateItemData
  - QY2ListView, 39
- updateItemStates
  - QY2ListView, 40
- updateStatus
  - QY2CheckListItem, 18
  - QY2DiskUsageListItem, 30
  - QY2ListViewItem, 43
- updateSteps
  - YQWizard, 305
- usedPercent
  - QY2DiskUsageListItem, 31
- usedSize
  - QY2DiskUsageListItem, 32
- validate
  - QY2CharValidator, 14
- value
  - YQCheckBox, 90
  - YQCheckBoxFrame, 94
  - YQDateField, 108
  - YQInputField, 148
  - YQIntField, 152
  - YQMultiLineEdit, 176
  - YQPartitionSplitter, 201
  - YQRadioButton, 212
  - YQSlider, 236
  - YQTimeField, 255
- valueChangedSlot
  - YQIntField, 153
  - YQSlider, 237
- waitForEventInternal
  - YQDialog, 120
- wizardDefaultButton
  - YQDialog, 120
- workArea

- YQWizard, 306
- x11Event
  - YQRawLineEdit, 216
- YQAlignment, 61
  - moveChild, 62
  - setBackgroundPixmap, 63
  - setEnabled, 63
  - setSize, 63
  - YQAlignment, 62
- yqApp
  - YQUI, 279
- YQApplication, 64
  - askForExistingDirectory, 67
  - askForExistingFile, 68
  - askForSaveFileName, 69
  - autoFonts, 69
  - beep, 70
  - busyCursor, 70
  - currentFont, 70
  - deviceUnits, 71
  - glyph, 71
  - headingFont, 72
  - layoutUnits, 72
  - makeScreenShot, 73
  - maybeLeftHandedUser, 73
  - normalCursor, 74
  - openContextMenu, 74
  - pickAutoFonts, 74
  - setApplicationIcon, 75
  - setApplicationTitle, 75
  - setLangFonts, 75
  - setLanguage, 76
  - YQApplication, 67
- YQBarGraph, 77
  - defaultTextColor, 79
  - doUpdate, 79
  - paintEvent, 79
  - preferredHeight, 79
  - preferredWidth, 80
  - setEnabled, 80
  - setSize, 80
- YQBusyIndicator, 81
  - preferredHeight, 82
  - preferredWidth, 82
  - setAlive, 83
  - setEnabled, 83
  - setLabel, 83
  - setSize, 84
  - setTimeout, 84
  - timeout, 84
- YQButtonBox, 85
  - moveChild, 86
  - setEnabled, 86
  - setSize, 86
- YQCheckBox, 87
  - preferredHeight, 88
  - preferredWidth, 88
  - setEnabled, 89
  - setKeyboardFocus, 89
  - setLabel, 89
  - setSize, 89
  - setUseBoldFont, 90
  - setValue, 90
  - value, 90
- YQCheckBoxFrame, 91
  - preferredHeight, 92
  - preferredWidth, 92
  - setEnabled, 93
  - setKeyboardFocus, 93
  - setLabel, 93
  - setSize, 94
  - setValue, 94
  - value, 94
- YQComboBox, 95
  - addItem, 97
  - deleteAllItems, 97
  - preferredHeight, 97
  - preferredWidth, 98
  - setEnabled, 98
  - setInputMaxLength, 98
  - setLabel, 98
  - setSize, 99
  - setText, 99
  - setValidChars, 99
  - slotSelected, 100
  - text, 100
  - textChanged, 101
- YQContextMenu, 102
  - preferredHeight, 103
  - preferredWidth, 103
  - rebuildMenuTree, 104
  - setSize, 104
- YQDateField, 105
  - preferredHeight, 106
  - preferredWidth, 106
  - setEnabled, 107
  - setKeyboardFocus, 107
  - setLabel, 107
  - setSize, 107
  - setValue, 108
  - value, 108
- YQDialog, 109
  - ~YQDialog, 112
  - activate, 113
  - activateDefaultButton, 113
  - center, 114
  - ensureOnlyOneDefaultButton, 114

- findWizard, 115
- gettingFocus, 115
- highlight, 115
- keyPressEvent, 116
- losingFocus, 116
- openInternal, 117
- pollEventInternal, 117
- preferredHeight, 118
- preferredWidth, 118
- setDefaultButton, 119
- setEnabled, 119
- setSize, 120
- waitForEventInternal, 120
- wizardDefaultButton, 120
- YQDialog, 112
- YQDownloadProgress, 121
  - preferredHeight, 123
  - preferredWidth, 123
  - setEnabled, 123
  - setExpectedSize, 123
  - setFilename, 124
  - setLabel, 124
  - setSize, 124
- YQDumbTab, 125
  - addItem, 126
  - deleteAllItems, 127
  - deselectAllItems, 127
  - preferredHeight, 127
  - preferredWidth, 127
  - selectItem, 128
  - setEnabled, 128
  - setSize, 128
  - shortcutChanged, 128
- YQEmpty, 129
  - setSize, 130
- YQFrame, 131
  - preferredHeight, 132
  - preferredWidth, 132
  - setEnabled, 132
  - setSize, 132
- YQGenericButton, 133
  - dialog, 135
  - eventFilter, 135
  - isShownAsDefault, 136
  - setEnabled, 136
  - setIcon, 136
  - setLabel, 137
  - setShortcut, 137
  - showAsDefault, 137
- YQGraphPluginIf, 138
- YQGraphPluginStub, 138
  - createGraph, 139
- YQImage, 140
  - preferredHeight, 142
  - preferredWidth, 142
  - setAutoScale, 142
  - setImage, 142
  - setSize, 143
- YQImage, 141
- YQInputField, 143
  - changed, 145
  - preferredHeight, 145
  - preferredWidth, 146
  - setEnabled, 146
  - setInputMaxLength, 146
  - setKeyboardFocus, 146
  - setLabel, 147
  - setSize, 147
  - setValidChars, 147
  - setValue, 148
  - value, 148
- YQIntField, 149
  - preferredHeight, 151
  - preferredWidth, 151
  - setEnabled, 151
  - setKeyboardFocus, 151
  - setLabel, 151
  - setSize, 152
  - setValueInternal, 152
  - value, 152
  - valueChangedSlot, 153
- YQLabel, 154
  - preferredHeight, 155
  - preferredWidth, 155
  - setEnabled, 155
  - setSize, 155
  - setText, 156
  - setUseBoldFont, 156
- YQLayoutBox, 157
  - moveChild, 158
  - setEnabled, 158
  - setSize, 159
- YQLayoutBox, 158
- YQLogView, 160
  - displayLogText, 161
  - preferredHeight, 161
  - preferredWidth, 162
  - setEnabled, 162
  - setLabel, 162
  - setSize, 162
- YQMainWinDock, 163
  - add, 165
  - closeEvent, 166
  - mainWinDock, 166
  - paintEvent, 167
  - remove, 167
  - resizeEvent, 168
  - show, 168

- YQMainWinDock, 165
- YQMenuBar, 169
  - preferredHeight, 170
  - preferredWidth, 170
  - rebuildMenuTree, 171
  - setEnabled, 171
  - setLabel, 171
  - setSize, 172
- YQMultiLineEdit, 172
  - enforceMaxInputLength, 174
  - preferredHeight, 174
  - preferredWidth, 174
  - setEnabled, 175
  - setInputMaxLength, 175
  - setLabel, 175
  - setSize, 176
  - setValue, 176
  - value, 176
- YQMultiProgressMeter, 177
  - doUpdate, 179
  - drawSegment, 179
  - paintEvent, 180
  - preferredHeight, 181
  - preferredWidth, 182
  - setEnabled, 182
  - setMargin, 182
  - setSize, 182
  - setSpacing, 183
  - setTriThickness, 183
  - triThickness, 184
- YQMultiSelectionBox, 185
  - addItem, 187
  - currentItem, 187
  - deleteAllItems, 187
  - deselectAllItems, 188
  - findItem, 188
  - preferredHeight, 188
  - preferredWidth, 189
  - selectItem, 189
  - setCurrentItem, 189
  - setEnabled, 190
  - setLabel, 190
  - setSize, 191
- YQMultiSelectionBoxItem, 192
- YQOptionalWidgetFactory, 193
  - YQOptionalWidgetFactory, 195
- YQPackageSelectorPluginIf, 195
- YQPackageSelectorPluginStub, 196
  - createPackageSelector, 197
- YQPartitionSplitter, 198
  - preferredHeight, 199
  - preferredWidth, 199
  - setEnabled, 200
  - setSize, 200
  - setValue, 200
  - value, 201
- YQProgressBar, 202
  - preferredHeight, 203
  - preferredWidth, 203
  - setEnabled, 203
  - setLabel, 204
  - setSize, 204
  - setValue, 204
- YQPushButton, 205
  - hit, 207
  - preferredHeight, 207
  - preferredWidth, 207
  - setSize, 208
- YQRadioButton, 209
  - preferredHeight, 210
  - preferredWidth, 210
  - setEnabled, 210
  - setKeyboardFocus, 211
  - setLabel, 211
  - setSize, 211
  - setUseBoldFont, 211
  - setValue, 212
  - value, 212
- YQRadioButtonGroup, 213
  - addRadioButton, 214
  - setEnabled, 214
  - setSize, 214
- YQRawLineEdit, 215
  - x11Event, 216
- YQReplacePoint, 217
  - setEnabled, 218
  - setSize, 218
  - showChild, 218
- YQRichText, 219
  - eventFilter, 220
  - preferredHeight, 221
  - preferredWidth, 221
  - setAutoScrollDown, 221
  - setEnabled, 222
  - setKeyboardFocus, 222
  - setPlainTextMode, 222
  - setSize, 222
  - setValue, 223
- YQSelectionBox, 223
  - addItem, 225
  - addItems, 225
  - deleteAllItems, 226
  - deselectAllItems, 226
  - eventFilter, 226
  - preferredHeight, 227
  - preferredWidth, 227
  - returnDelayed, 228
  - returnImmediately, 228

- [selectItem](#), 228
  - [setEnabled](#), 229
  - [setKeyboardFocus](#), 229
  - [setLabel](#), 229
  - [setSize](#), 230
  - [slotSelectionChanged](#), 230
- [YQSignalBlocker](#), 231
  - [~YQSignalBlocker](#), 232
  - [YQSignalBlocker](#), 232
- [YQSlider](#), 232
  - [preferredHeight](#), 235
  - [preferredWidth](#), 235
  - [setEnabled](#), 235
  - [setLabel](#), 235
  - [setSize](#), 236
  - [setValueInternal](#), 236
  - [value](#), 236
  - [valueChangedSlot](#), 237
  - [YQSlider](#), 234
- [YQSpacing](#), 238
  - [setSize](#), 239
- [YQSquash](#), 239
  - [setEnabled](#), 240
  - [setSize](#), 240
- [YQTable](#), 241
  - [addItem](#), 243
  - [addItems](#), 243
  - [cellChanged](#), 244
  - [deleteAllItems](#), 244
  - [deselectAllItems](#), 245
  - [preferredHeight](#), 245
  - [preferredWidth](#), 245
  - [selectItem](#), 246
  - [setEnabled](#), 246
  - [setKeepSorting](#), 246
  - [setKeyboardFocus](#), 247
  - [setSize](#), 247
  - [slotContextMenu](#), 247
- [YQTableListViewItem](#), 248
- [YQTextBrowser](#), 250
  - [setSource](#), 251
- [YQTimeField](#), 251
  - [preferredHeight](#), 253
  - [preferredWidth](#), 253
  - [setEnabled](#), 253
  - [setKeyboardFocus](#), 253
  - [setLabel](#), 254
  - [setSize](#), 254
  - [setValue](#), 254
  - [value](#), 255
- [YQTimezoneSelector](#), 255
  - [preferredHeight](#), 257
  - [preferredWidth](#), 257
  - [setSize](#), 257
- [YQTimezoneSelectorPrivate](#), 258
- [YQTimezoneSelectorPrivate::Location](#), 10
- [YQTree](#), 259
  - [currentItem](#), 261
  - [deleteAllItems](#), 261
  - [deselectAllItems](#), 261
  - [preferredHeight](#), 262
  - [preferredWidth](#), 262
  - [rebuildTree](#), 262
  - [selectItem](#), 263
  - [setEnabled](#), 263
  - [setLabel](#), 264
  - [setSize](#), 264
  - [slotActivated](#), 264
  - [slotContextMenu](#), 265
  - [slotItemChanged](#), 265
  - [slotSelectionChanged](#), 266
- [YQTreeWidgetItem](#), 267
  - [key](#), 268
  - [setOpen](#), 268
- [YQUI](#), 269
  - [applicationTitle](#), 273
  - [askConfigureLogging](#), 273
  - [askSaveLogs](#), 273
  - [blockEvents](#), 274
  - [consumePendingEvent](#), 274
  - [createOptionalWidgetFactory](#), 275
  - [createWidgetFactory](#), 275
  - [deleteNotify](#), 275
  - [eventsBlocked](#), 275
  - [idleLoop](#), 276
  - [initUI](#), 276
  - [makeScreenShot](#), 277
  - [pendingEvent](#), 277
  - [probeX11Display](#), 278
  - [raiseFatalError](#), 278
  - [runPkgSelection](#), 278
  - [sendEvent](#), 278
  - [uiThreadDestructor](#), 279
  - [yqApp](#), 279
- [YQUISignalReceiver](#), 280
- [YQWidgetCaption](#), 281
  - [setText](#), 282
  - [YQWidgetCaption](#), 282
- [YQWidgetFactory](#), 283
  - [YQWidgetFactory](#), 285
- [YQWizard](#), 285
  - [addMenu](#), 291
  - [addMenuEntry](#), 291
  - [addMenuSeparator](#), 291
  - [addStep](#), 292
  - [addStepHeading](#), 292
  - [addSubMenu](#), 292
  - [addTreeWidgetItem](#), 293

- backButton, [293](#)
- connectNotify, [294](#)
- copySteps, [294](#)
- currentStep, [294](#)
- currentTreeSelection, [295](#)
- debugLabel, [295](#)
- deleteMenus, [295](#)
- deleteSteps, [295](#)
- deleteTreeItems, [296](#)
- direction, [296](#)
- disconnectNotify, [296](#)
- eventFilter, [297](#)
- findStep, [297](#)
- findTreeItem, [297](#)
- hideReleaseNotesButton, [298](#)
- nextClicked, [298](#)
- preferredHeight, [298](#)
- preferredWidth, [299](#)
- retranslateInternalButtons, [299](#)
- selectTreeItem, [299](#)
- sendTreeEvent, [300](#)
- setButtonLabel, [300](#)
- setCurrentStep, [301](#)
- setDialogHeading, [301](#)
- setDialogIcon, [302](#)
- setDialogTitle, [302](#)
- setHelpText, [302](#)
- setSize, [303](#)
- showHelp, [303](#)
- showHotkeys, [303](#)
- showReleaseNotesButton, [304](#)
- showSteps, [304](#)
- showTree, [304](#)
- stepsList, [305](#)
- treeSelectionChanged, [305](#)
- updateSteps, [305](#)
- workArea, [306](#)
- YQWizard::Step, [56](#)
- ~Step, [57](#)
- YQWizard::StepHeading, [58](#)
- YQWizard::TreeItem, [59](#)
- YQWizardButton, [307](#)
- preferredHeight, [308](#)
- preferredWidth, [308](#)
- setSize, [309](#)
- show, [309](#)