

GobChartsWidget

1.0

Generated by Doxygen 1.8.2

Mon Jan 14 2013 16:17:05

Contents

1	The GobChartsWidget Library	1
1.1	Introduction	1
1.2	Download	1
1.3	Using the Downloaded Libraries	1
1.3.1	Known Issues: Release vs Debug	1
1.3.2	Known Issues: Widget Style Sheets	2
2	Module Index	3
2.1	Modules	3
3	Namespace Index	5
3.1	Namespace List	5
4	Hierarchical Index	7
4.1	Class Hierarchy	7
5	Class Index	9
5.1	Class List	9
6	Module Documentation	11
6.1	Chart Views	11
6.1.1	Detailed Description	11
7	Namespace Documentation	13
7.1	GobChartsColours Namespace Reference	13
7.1.1	Detailed Description	13
7.1.2	Function Documentation	13
7.1.2.1	getNextColour	13
7.1.2.2	resetColourIndex	13
8	Class Documentation	15
8.1	GobChartsBarView Class Reference	15
8.1.1	Detailed Description	15
8.1.2	Member Function Documentation	16

8.1.2.1	generateGraphicsItems	16
8.1.2.2	needsGrid	16
8.1.2.3	typeInteger	16
8.2	GobChartsFactory Class Reference	16
8.2.1	Detailed Description	17
8.2.2	Member Function Documentation	17
8.2.2.1	createChart	17
8.2.2.2	getInstance	17
8.3	GobChartsGraphItems Class Reference	17
8.3.1	Detailed Description	18
8.3.2	Member Function Documentation	19
8.3.2.1	addItem	19
8.3.2.2	addItemToScene	19
8.3.2.3	clearSelection	19
8.3.2.4	deleteItems	19
8.3.2.5	getItemLegend	19
8.3.2.6	getItemRectF	19
8.3.2.7	getItemRectF	20
8.3.2.8	getModelIndex	20
8.3.2.9	getModelIndex	20
8.3.2.10	lastDebugLogMsg	20
8.3.2.11	removeItemsFromScene	20
8.3.2.12	setDebugLoggingOn	20
8.3.2.13	setSelected	20
8.4	GobChartsGrid Class Reference	21
8.4.1	Detailed Description	22
8.4.2	Member Function Documentation	22
8.4.2.1	addGridToScene	22
8.4.2.2	constructGrid	22
8.4.2.3	gridWidth	22
8.4.2.4	lastDebugLogMsg	22
8.4.2.5	removeGridFromScene	22
8.4.2.6	setDebugLoggingOn	23
8.4.2.7	setGridColour	23
8.4.2.8	setGridLineStyle	23
8.4.2.9	setGridRectF	23
8.4.2.10	setHorizontalGridLines	23
8.4.2.11	setVerticalGridLines	23
8.5	GobChartsLineView Class Reference	24
8.5.1	Detailed Description	24

8.5.2	Member Function Documentation	25
8.5.2.1	generateGraphicsItems	25
8.5.2.2	needsGrid	25
8.5.2.3	typeInteger	25
8.6	GobChartsNoCopy Class Reference	25
8.6.1	Detailed Description	26
8.7	GobChartsPieView Class Reference	26
8.7.1	Detailed Description	27
8.7.2	Member Function Documentation	27
8.7.2.1	generateGraphicsItems	27
8.7.2.2	needsGrid	27
8.7.2.3	typeInteger	27
8.8	GobChartsTextItem Class Reference	28
8.8.1	Detailed Description	29
8.8.2	Member Function Documentation	29
8.8.2.1	alignment	29
8.8.2.2	getStateXML	29
8.8.2.3	identity	29
8.8.2.4	mousePressEvent	29
8.8.2.5	receiveKeyEvent	29
8.8.2.6	setAlignment	29
8.8.2.7	setFont	30
8.8.2.8	setMaxFontSize	30
8.8.2.9	setPlainText	30
8.8.2.10	setRectF	30
8.8.2.11	setStateXML	30
8.9	GobChartsToolsWidget Class Reference	31
8.9.1	Detailed Description	32
8.9.2	Member Function Documentation	32
8.9.2.1	clearChart	32
8.9.2.2	createChart	32
8.9.2.3	emitStateSignals	32
8.9.2.4	getStateXML	32
8.9.2.5	loadChart	33
8.9.2.6	receiveLabelDetails	33
8.9.2.7	requestLabelDetails	33
8.9.2.8	saveChart	33
8.9.2.9	setAllowedDataRange	33
8.9.2.10	setFixedColour	34
8.9.2.11	setGridColour	34

8.9.2.12	setGridLineStyle	34
8.9.2.13	setHorizontalGridLines	34
8.9.2.14	setLabelDetails	34
8.9.2.15	setRandomColours	34
8.9.2.16	setShowTotalRange	34
8.9.2.17	setStateXML	35
8.9.2.18	setVerticalGridLines	35
8.10	GobChartsValidItems Class Reference	35
8.10.1	Detailed Description	36
8.10.2	Member Function Documentation	36
8.10.2.1	addValidItem	36
8.10.2.2	category	36
8.10.2.3	clear	36
8.10.2.4	data	36
8.10.2.5	size	36
8.10.2.6	validRows	36
8.11	GobChartsView Class Reference	36
8.11.1	Detailed Description	40
8.11.2	Member Enumeration Documentation	40
8.11.2.1	GobChartsColumn	40
8.11.3	Constructor & Destructor Documentation	40
8.11.3.1	~GobChartsView	40
8.11.4	Member Function Documentation	40
8.11.4.1	addToGraphItemsContainer	40
8.11.4.2	category	41
8.11.4.3	clearLegend	41
8.11.4.4	createLegendItem	41
8.11.4.5	dataChanged	41
8.11.4.6	debugLog	41
8.11.4.7	drawChart	41
8.11.4.8	emitLabelDetails	41
8.11.4.9	fixedColour	42
8.11.4.10	generateGraphicsItems	42
8.11.4.11	getStateXML	42
8.11.4.12	gridWidth	42
8.11.4.13	highLightLegendItem	42
8.11.4.14	horizontalOffset	42
8.11.4.15	indexAt	43
8.11.4.16	innerSceneRectF	43
8.11.4.17	isIndexHidden	43

8.11.4.18 isWithinAllowedRange	43
8.11.4.19 keyPressEvent	43
8.11.4.20 lastDebugLogMsg	43
8.11.4.21 legendItemSelected	43
8.11.4.22 mousePressEvent	44
8.11.4.23 moveCursor	44
8.11.4.24 needsGrid	44
8.11.4.25 nrValidItems	44
8.11.4.26 requestLabelDetails	44
8.11.4.27 rowsAboutToBeRemoved	44
8.11.4.28 rowsInserted	45
8.11.4.29 scrollTo	45
8.11.4.30 setAllowedDataRange	45
8.11.4.31 setDebugLoggingOn	45
8.11.4.32 setFixedColour	45
8.11.4.33 setGridColour	46
8.11.4.34 setGridLineStyle	46
8.11.4.35 setHorizontalGridLines	46
8.11.4.36 setLabelDetails	46
8.11.4.37 setModel	46
8.11.4.38 setRandomColours	47
8.11.4.39 setSelection	47
8.11.4.40 setShowTotalRange	47
8.11.4.41 setStateXML	47
8.11.4.42 setVerticalGridLines	47
8.11.4.43 stripSpace	48
8.11.4.44 totalValue	48
8.11.4.45 typeInteger	48
8.11.4.46 useFixedColour	48
8.11.4.47 validRowList	48
8.11.4.48 value	48
8.11.4.49 verticalOffset	49
8.11.4.50 visualRect	49
8.11.4.51 visualRegionForSelection	49
8.11.5 Friends And Related Function Documentation	49
8.11.5.1 GobChartsViewPrivate	49
8.12 GobChartsWidget Class Reference	49
8.12.1 Detailed Description	50
8.12.2 Member Function Documentation	50
8.12.2.1 lastDebugLogMsg	50

8.12.2.2	setDebugLoggingOn	50
8.12.2.3	setModel	51
8.12.2.4	setSelectionModel	51

Index	51
--------------	-----------

Chapter 1

The GobChartsWidget Library

1.1 Introduction

If you are looking for a complex, end-user charting library for use with Qt, this is probably not it. However, if your application is in need of some basic charting functionality and your primary data consists of a one to one mapping of category to value, then I believe that this widget may just be what you need (please see [GobChartsWidget](#) for important information regarding usage).

If you want to know more, the [official site](#) contains all the information about features and limitations as well as why this library is the way it is.

Please also feel free to [contact me](#) for any reason whatsoever.

1.2 Download

If you haven't yet, please see the [download page](#) for a list of possible download options and **ensure that you download the correct version for your compiler.**

1.3 Using the Downloaded Libraries

Simply add

```
LIBS += -L$$PWD/../path/to/lib -lgobchartswidget
INCLUDEPATH += $$PWD/../path/to/headers
DEPENDPATH += $$PWD/../path/to/headers
```

to your Qt project file and make sure you `#include <gobchartswidget.h>` where you want to instantiate the widget (the above will work for static linking, if you wish to load the library at runtime, have a look at the API documentation for QLibrary).

Alternatively, if you're using QtCreator, simply right-click on your Qt project, select "Add Library" and follow the prompts.

1.3.1 Known Issues: Release vs Debug

If you run into the following error when compiling your application with [GobChartsWidget](#):

"QWidget: Must construct a QApplication before a QPaintDevice"

then there is most probably a discrepancy between the compile mode you are using and that of the library you are linking to, i.e. if you are compiling in "release", then make sure you're using the "release" version of the library and,

obviously, if you're compiling in "debug" mode, use the "debug" version.

1.3.2 Known Issues: Widget Style Sheets

I developed this widget on Windows 7 and subsequently tested the widget on an Ubuntu (12.04 LTS) virtual machine where I noticed that some of the widget's elements looked downright rubbish.

It seems that the only workaround is to set specific style sheets on the widget for a more polished look. Since I do not wish to make the call with regards to what the widget should or should not look like, I leave this up to you (having said that, I *did* set a very basic style sheet on the Ubuntu distribution to ensure that the widget layout is correct, the style sheet is distributed with the source - see the "toolswidget" folder for the .txt file).

If you do come up with nice style sheets, I would appreciate it greatly if you could simply copy and paste your style sheet string into this [contact form](#) along with the OS name and version you implemented it on so that I may start distributing them along with the widget's source.

Chapter 2

Module Index

2.1 Modules

Here is a list of all modules:

Chart Views	11
-----------------------	----

Chapter 3

Namespace Index

3.1 Namespace List

Here is a list of all documented namespaces with brief descriptions:

GobChartsColours	
Manages a fixed list of colours used for colouring the graphics items	13

Chapter 4

Hierarchical Index

4.1 Class Hierarchy

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

GobChartsNoCopy	25
GobChartsFactory	16
GobChartsGraphItems	17
GobChartsGrid	21
GobChartsTextItem	28
GobChartsToolsWidget	31
GobChartsValidItems	35
GobChartsView	36
GobChartsBarView	15
GobChartsLineView	24
GobChartsPieView	26
GobChartsWidget	49
QAbstractItemView	
GobChartsView	36
QFrame	
GobChartsToolsWidget	31
GobChartsWidget	49
QGraphicsTextItem	
GobChartsTextItem	28
QObject	
GobChartsGraphItems	17
GobChartsGrid	21
GobChartsValidItems	35

Chapter 5

Class Index

5.1 Class List

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

GobChartsBarView	Responsible for drawing BAR charts	15
GobChartsFactory	Factory class responsible for the creation of charts at run-time	16
GobChartsGraphItems	Manages associations between graph items and model indices	17
GobChartsGrid	Generates and manages grid and axes lines	21
GobChartsLineView	Responsible for drawing LINE charts	24
GobChartsNoCopy	Inheriting from this class disables copying	25
GobChartsPieView	Responsible for drawing PIE charts	26
GobChartsTextItem	A convenience text item class that can resize itself	28
GobChartsToolsWidget	Provides the end-user with the tools to select and customise charts	31
GobChartsValidItems	Controls and manages valid model rows	35
GobChartsView	36
GobChartsWidget	Main widget class	49

Chapter 6

Module Documentation

6.1 Chart Views

Abstract base class from which all chart type (view) classes must inherit.

Classes

- class [GobChartsBarView](#)
Responsible for drawing BAR charts.
- class [GobChartsLineView](#)
Responsible for drawing LINE charts.
- class [GobChartsPieView](#)
Responsible for drawing PIE charts.

6.1.1 Detailed Description

Abstract base class from which all chart type (view) classes must inherit. Each view class represents a specific chart type (with the exception of [GobChartsView](#), the abstract base class from which all these views derive).

The three supported chart types and their corresponding view classes are:

- BAR - [GobChartsBarView](#)
- PIE - [GobChartsPieView](#)
- LINE - [GobChartsLineView](#)

Chapter 7

Namespace Documentation

7.1 GobChartsColours Namespace Reference

Manages a fixed list of colours used for colouring the graphics items.

Functions

- QColor [getNextColour](#) ()
Returns the next available colour in the list.
- void [resetColourIndex](#) ()
Resets the colour index to the beginning of the list.

7.1.1 Detailed Description

Manages a fixed list of colours used for colouring the graphics items.

7.1.2 Function Documentation

7.1.2.1 QColor GobChartsColours::getNextColour ()

Returns the next available colour in the list.

Definition at line 48 of file gobchartscolours.cpp.

7.1.2.2 void GobChartsColours::resetColourIndex ()

Resets the colour index to the beginning of the list.

Definition at line 64 of file gobchartscolours.cpp.

Chapter 8

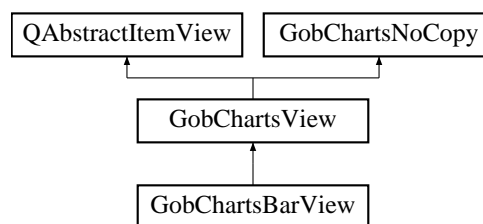
Class Documentation

8.1 GobChartsBarView Class Reference

Responsible for drawing BAR charts.

```
#include <gobchartsbarview.h>
```

Inheritance diagram for GobChartsBarView:



Public Member Functions

- [GobChartsBarView](#) (QWidget *parent=0)
Constructor.
- virtual [~GobChartsBarView](#) ()
Destructor.

Protected Member Functions

- void [generateGraphicsItems](#) ()
Graphics (chart) items.
- bool [needsGrid](#) () const
Grid required (returns "true").
- QString [typeInteger](#) () const
Type integer is "0".

Additional Inherited Members

8.1.1 Detailed Description

Responsible for drawing BAR charts.

This class implements the following [GobChartsView](#) pure virtual functions:

1. [generateGraphicsItems\(\)](#)
2. [needsGrid\(\)](#)
3. [typeInteger\(\)](#)

Definition at line 39 of file gobchartsbarview.h.

8.1.2 Member Function Documentation

8.1.2.1 void GobChartsBarView::generateGraphicsItems () [protected], [virtual]

Graphics (chart) items.

This function generates the chart's bar columns and uses the various categories' values to determine each column's relative size with respect to each other and the confines of the available space.

Implements [GobChartsView](#).

Definition at line 49 of file gobchartsbarview.cpp.

8.1.2.2 bool GobChartsBarView::needsGrid () const [protected], [virtual]

Grid required (returns "true").

Implements [GobChartsView](#).

Definition at line 138 of file gobchartsbarview.cpp.

8.1.2.3 QString GobChartsBarView::typeInteger () const [protected], [virtual]

Type integer is "0".

Implements [GobChartsView](#).

Definition at line 145 of file gobchartsbarview.cpp.

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

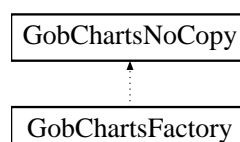
- gobchartsbarview.h
- gobchartsbarview.cpp

8.2 GobChartsFactory Class Reference

Factory class responsible for the creation of charts at run-time.

```
#include <gobchartsfactory.h>
```

Inheritance diagram for GobChartsFactory:



Public Member Functions

- [GobChartsView](#) * [createChart](#) (GobChartsType type, QWidget *parent)
Creates a chart view of type "type".

Static Public Member Functions

- static [GobChartsFactory](#) * [getInstance](#) ()
Returns the singleton instance.

8.2.1 Detailed Description

Factory class responsible for the creation of charts at run-time.

This factory is implemented as a singleton and is responsible for the creation of the user-selected chart types at run-time. Care must be taken when using this factory as it does not maintain ownership of the chart views it creates and clean-up becomes the responsibility of the calling object when no parent is specified.

Definition at line 39 of file gobchartsfactory.h.

8.2.2 Member Function Documentation

8.2.2.1 [GobChartsView](#) * [GobChartsFactory::createChart](#) ([GobChartsType](#) type, [QWidget](#) * parent)

Creates a chart view of type "type".

If no parent is provided, clean-up becomes the responsibility of the calling object.

Definition at line 54 of file gobchartsfactory.cpp.

8.2.2.2 [GobChartsFactory](#) * [GobChartsFactory::getInstance](#) () [static]

Returns the singleton instance.

Definition at line 35 of file gobchartsfactory.cpp.

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

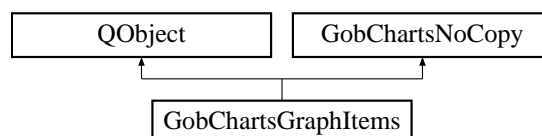
- gobchartsfactory.h
- gobchartsfactory.cpp

8.3 GobChartsGraphItems Class Reference

Manages associations between graph items and model indices.

```
#include <gobchartsgraphitems.h>
```

Inheritance diagram for GobChartsGraphItems:



Signals

- void [lastDebugLogMsg](#) (QString)

Broadcast debug messages.

Public Member Functions

- [GobChartsGraphItems](#) (QObject *parent=0)

Constructor.

- virtual [~GobChartsGraphItems](#) ()

Destructor.

- void [addItemToScene](#) (QGraphicsScene *scene)

Adds all the mapped QGraphicsItem's to "scene".

- void [removeItemsFromScene](#) (QGraphicsScene *scene)

Removes all the mapped QGraphicsItem's from "scene".

- void [setSelected](#) (int categoryRow)

Manipulates the QGraphicsItem corresponding to the given category row (if mapped and found) by changing it's opacity.

- void [clearSelection](#) ()

Clears the current selection.

- void [addItem](#) (const QModelIndex &valueIndex, QGraphicsItem *item, const QString &legendText)

Adds a QModelIndex/QGraphicsItem pair to the map.

- void [deleteItems](#) ()

Delete all the mapped QGraphicsItem's and clear the map.

- QModelIndex [getModelIndex](#) (QGraphicsItem *item) const

Returns the model index corresponding to "item".

- QModelIndex [getModelIndex](#) (const QString &text) const

Returns the model index of the item with legend "text".

- QRectF [getItemRectF](#) (const QModelIndex &index) const

Returns the rectangle of the graphics item corresponding to "index".

- QRectF [getItemRectF](#) (const QString &text) const

Returns the rectangle of the graphics item corresponding to the legend with text "text".

- QString [getItemLegend](#) (QGraphicsItem *item) const

Returns the legend text associated with the graphics item.

- void [setDebugLoggingOn](#) (bool logging)

Turn debug logging "on" or "off" (default "off").

8.3.1 Detailed Description

Manages associations between graph items and model indices.

[GobChartsGraphItems](#) is a container class that manages associations between graph items (QGraphicsItem's) and their respective indices (QModelIndex's). The class furthermore provides the functionality to add or remove the mapped items to or from a QGraphicsScene and to highlight items currently selected in the view by changing the item's opacity to create the visual distinction.

Definition at line 42 of file gobchartsgraphitems.h.

8.3.2 Member Function Documentation

8.3.2.1 void GobChartsGraphItems::addItem (const QModelIndex & *valueIndex*, QGraphicsItem * *item*, const QString & *legendText*)

Adds a QModelIndex/QGraphicsItem pair to the map.

This function furthermore ensures that the added item can be retrieved based on its corresponding legend text.

See Also

[deleteItems\(\)](#)

Definition at line 128 of file gobchartsgraphitems.cpp.

8.3.2.2 void GobChartsGraphItems::addItemToScene (QGraphicsScene * *scene*)

Adds all the mapped QGraphicsItem's to "scene".

See Also

[removeItemsFromScene\(\)](#)

Definition at line 54 of file gobchartsgraphitems.cpp.

8.3.2.3 void GobChartsGraphItems::clearSelection ()

Clears the current selection.

This function will return all the graphic items to their original opacity.

Definition at line 115 of file gobchartsgraphitems.cpp.

8.3.2.4 void GobChartsGraphItems::deleteItems ()

Delete all the mapped QGraphicsItem's and clear the map.

See Also

[addItem\(\)](#)

Definition at line 139 of file gobchartsgraphitems.cpp.

8.3.2.5 QString GobChartsGraphItems::getItemLegend (QGraphicsItem * *item*) const

Returns the legend text associated with the graphics item.

Definition at line 216 of file gobchartsgraphitems.cpp.

8.3.2.6 QRectF GobChartsGraphItems::getItemRectF (const QModelIndex & *index*) const

Returns the rectangle of the graphics item corresponding to "index".

Definition at line 184 of file gobchartsgraphitems.cpp.

8.3.2.7 `QRectF GobChartsGraphItems::getItemRectF (const QString & text) const`

Returns the rectangle of the graphics item corresponding to the legend with text "text".

Definition at line 200 of file gobchartsgraphitems.cpp.

8.3.2.8 `QModelIndex GobChartsGraphItems::getModelIndex (QGraphicsItem * item) const`

Returns the model index corresponding to "item".

Definition at line 156 of file gobchartsgraphitems.cpp.

8.3.2.9 `QModelIndex GobChartsGraphItems::getModelIndex (const QString & text) const`

Returns the model index of the item with legend "text".

Definition at line 170 of file gobchartsgraphitems.cpp.

8.3.2.10 `void GobChartsGraphItems::lastDebugLogMsg (QString) [signal]`

Broadcast debug messages.

This signal broadcasts the last debug log message and will only be emitted when debug logging is turned "on".

See Also

[setDebugLoggingOn\(\)](#)

8.3.2.11 `void GobChartsGraphItems::removeItemsFromScene (QGraphicsScene * scene)`

Removes all the mapped QGraphicsItem's from "scene".

See Also

[addItemToScene\(\)](#)

Definition at line 74 of file gobchartsgraphitems.cpp.

8.3.2.12 `void GobChartsGraphItems::setDebugLoggingOn (bool logging)`

Turn debug logging "on" or "off" (default "off").

See Also

[lastDebugLogMsg\(\)](#)

Definition at line 228 of file gobchartsgraphitems.cpp.

8.3.2.13 `void GobChartsGraphItems::setSelected (int categoryRow)`

Manipulates the QGraphicsItem corresponding to the given category row (if mapped and found) by changing it's opacity.

When selected the first time, this function lessens the item's opacity (makes it more transparent), otherwise it resets the item to being completely opaque.

See Also

[clearSelection\(\)](#)

Definition at line 94 of file gobchartsgraphitems.cpp.

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

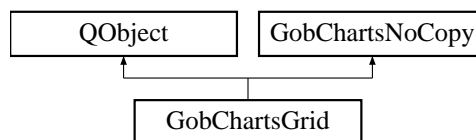
- gobchartsgraphitems.h
- gobchartsgraphitems.cpp

8.4 GobChartsGrid Class Reference

Generates and manages grid and axes lines.

```
#include <gobchartsgrid.h>
```

Inheritance diagram for GobChartsGrid:



Signals

- void [lastDebugLogMsg](#) (QString)
Broadcast debug messages.

Public Member Functions

- [GobChartsGrid](#) (QObject *parent=0)
Constructor.
- virtual [~GobChartsGrid](#) ()
Destructor.
- void [addGridToScene](#) (QGraphicsScene *scene)
Adds the axes and grid lines (if any) to "scene".
- void [removeGridFromScene](#) (QGraphicsScene *scene)
Removes the axes and grid lines (if any) from "scene".
- void [constructGrid](#) ()
Constructs grid and axes lines.
- void [setGridRectF](#) (const QRectF &rect)
Sets the grid's spatial dimensions.
- void [setGridLineStyle](#) (Qt::PenStyle style)
Set the pen style used to draw the grid lines.
- void [setHorizontalGridLines](#) (bool set, int number)
Set horizontal grid lines.
- void [setVerticalGridLines](#) (bool set, int number)
Set horizontal grid lines.
- void [setGridColour](#) (QColor colour)
Sets the grid line's colour to "colour".
- qreal [gridWidth](#) () const

Returns the grid's width.

- void [setDebugLoggingOn](#) (bool logging)

Turn debug logging "on" or "off" (default "off").

8.4.1 Detailed Description

Generates and manages grid and axes lines.

[GobChartsGrid](#) maintains a list of all grid lines as well as the chart's x and y axes. Line lengths and positions are recalculated when [constructGrid\(\)](#) is called, are determined by the dimensions and restricted to the boundaries of the grid's rectangle. The class furthermore provides the functionality to add or remove the lines to a QGraphicsScene.

Definition at line 40 of file gobchartsgrid.h.

8.4.2 Member Function Documentation

8.4.2.1 void GobChartsGrid::addGridToScene (QGraphicsScene * scene)

Adds the axes and grid lines (if any) to "scene".

See Also

[removeGridFromScene\(\)](#)

Definition at line 69 of file gobchartsgrid.cpp.

8.4.2.2 void GobChartsGrid::constructGrid ()

Constructs grid and axes lines.

Deletes all existing line items, recalculates the grid's dimensions and creates the new axes and grid lines according to the geometric settings and custom grid specifications.

Definition at line 109 of file gobchartsgrid.cpp.

8.4.2.3 qreal GobChartsGrid::gridWidth () const

Returns the grid's width.

Definition at line 187 of file gobchartsgrid.cpp.

8.4.2.4 void GobChartsGrid::lastDebugLogMsg (QString) [signal]

Broadcast debug messages.

This signal broadcasts the last debug log message and will only be emitted when debug logging is turned "on".

See Also

[setDebugLoggingOn\(\)](#)

8.4.2.5 void GobChartsGrid::removeGridFromScene (QGraphicsScene * scene)

Removes the axes and grid lines (if any) from "scene".

See Also

[addGridToScene\(\)](#)

Definition at line 89 of file gobchartsgrid.cpp.

8.4.2.6 void GobChartsGrid::setDebugLoggingOn (bool *logging*)

Turn debug logging "on" or "off" (default "off").

See Also

[lastDebugLogMsg\(\)](#)

Definition at line 224 of file gobchartsgrid.cpp.

8.4.2.7 void GobChartsGrid::setGridColour (QColor *colour*)

Sets the grid line's colour to "colour".

Definition at line 210 of file gobchartsgrid.cpp.

8.4.2.8 void GobChartsGrid::setGridLineStyle (Qt::PenStyle *style*)

Set the pen style used to draw the grid lines.

Parameters

<i>style</i>	- dashed, dotted, solid, etc. (see Qt API documentation for details on the enum).
--------------	---

Definition at line 217 of file gobchartsgrid.cpp.

8.4.2.9 void GobChartsGrid::setGridRectF (const QRectF & *rect*)

Sets the grid's spatial dimensions.

The grid will be confined to the dimensions of the given rectangle.

Definition at line 180 of file gobchartsgrid.cpp.

8.4.2.10 void GobChartsGrid::setHorizontalGridLines (bool *set*, int *number*)

Set horizontal grid lines.

Turn horizontal grid lines "on" or "off" and also sets the number of grid lines.

Definition at line 194 of file gobchartsgrid.cpp.

8.4.2.11 void GobChartsGrid::setVerticalGridLines (bool *set*, int *number*)

Set horizontal grid lines.

Turn vertical grid lines "on" or "off" and also sets the number of grid lines.

Definition at line 202 of file gobchartsgrid.cpp.

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

- gobchartsgrid.h

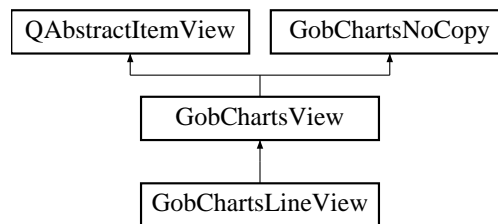
- gobchartsgrid.cpp

8.5 GobChartsLineView Class Reference

Responsible for drawing LINE charts.

```
#include <gobchartslineview.h>
```

Inheritance diagram for GobChartsLineView:



Public Member Functions

- [GobChartsLineView](#) (QWidget *parent=0)
Constructor.
- virtual [~GobChartsLineView](#) ()
Destructor.

Protected Member Functions

- void [generateGraphicsItems](#) ()
Graphics (chart) items.
- bool [needsGrid](#) () const
Grid required (returns "true").
- QString [typeInteger](#) () const
Type integer is "2".

Additional Inherited Members

8.5.1 Detailed Description

Responsible for drawing LINE charts.

This class implements the following [GobChartsView](#) pure virtual functions:

1. [generateGraphicsItems\(\)](#)
2. [needsGrid\(\)](#)
3. [typeInteger\(\)](#)

Definition at line 39 of file gobchartslineview.h.

8.5.2 Member Function Documentation

8.5.2.1 void GobChartsLineView::generateGraphicsItems () [protected],[virtual]

Graphics (chart) items.

This function generates the chart's line segments and uses the various categories' values to determine each segment's relative size with respect to each other and the confines of the available space.

Implements [GobChartsView](#).

Definition at line 51 of file gobchartslineview.cpp.

8.5.2.2 bool GobChartsLineView::needsGrid () const [protected],[virtual]

Grid required (returns "true").

Implements [GobChartsView](#).

Definition at line 126 of file gobchartslineview.cpp.

8.5.2.3 QString GobChartsLineView::typeInteger () const [protected],[virtual]

Type integer is "2".

Implements [GobChartsView](#).

Definition at line 133 of file gobchartslineview.cpp.

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

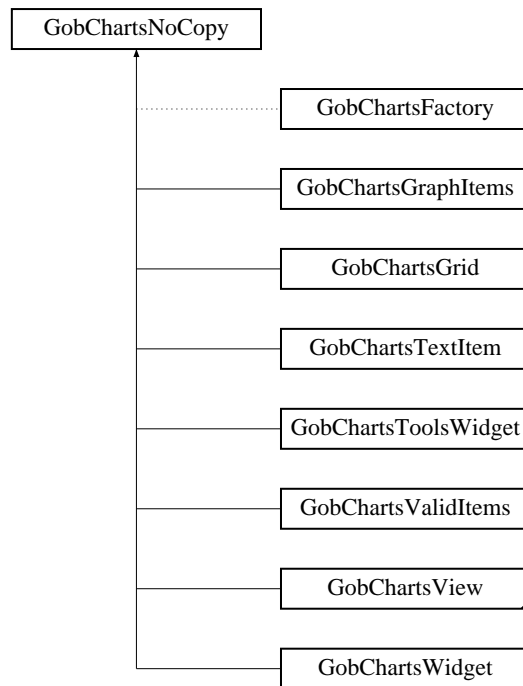
- gobchartslineview.h
- gobchartslineview.cpp

8.6 GobChartsNoCopy Class Reference

Inheriting from this class disables copying.

```
#include <gobchartsnocopy.h>
```

Inheritance diagram for GobChartsNoCopy:



8.6.1 Detailed Description

Inheriting from this class disables copying.

Inheriting from this class disables copying via the copy constructor and assignment operator. The advantage of doing it this way (as opposed to simply implementing copy constructors and assignment operators as "private" in the child classes) is that this moves the error detection up from link time to compile time, which I prefer.

Definition at line 33 of file gobchartsnocopy.h.

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

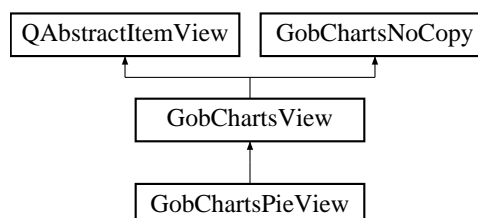
- gobchartsnocopy.h

8.7 GobChartsPieView Class Reference

Responsible for drawing PIE charts.

```
#include <gobchartspieview.h>
```

Inheritance diagram for GobChartsPieView:



Public Member Functions

- [GobChartsPieView](#) (QWidget *parent=0)

Constructor.

- virtual [~GobChartsPieView](#) ()

Destructor.

Protected Member Functions

- void [generateGraphicsItems](#) ()
Graphics (chart) items.
- bool [needsGrid](#) () const
No grid required (returns "false").
- QString [typeInteger](#) () const
Type integer is "1".

Additional Inherited Members

8.7.1 Detailed Description

Responsible for drawing PIE charts.

This class implements the following [GobChartsView](#) pure virtual functions:

1. [generateGraphicsItems\(\)](#)
2. [needsGrid\(\)](#)
3. [typeInteger\(\)](#)

Definition at line 39 of file gobchartspieview.h.

8.7.2 Member Function Documentation

8.7.2.1 void GobChartsPieView::generateGraphicsItems () [protected], [virtual]

Graphics (chart) items.

This function generates the chart's pie segments and uses the various categories' values to determine each segment's relative size with respect to each other and the confines of the available space.

Implements [GobChartsView](#).

Definition at line 47 of file gobchartspieview.cpp.

8.7.2.2 bool GobChartsPieView::needsGrid () const [protected], [virtual]

No grid required (returns "false").

Implements [GobChartsView](#).

Definition at line 115 of file gobchartspieview.cpp.

8.7.2.3 QString GobChartsPieView::typeInteger () const [protected], [virtual]

Type integer is "1".

Implements [GobChartsView](#).

Definition at line 122 of file gobchartspieview.cpp.

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

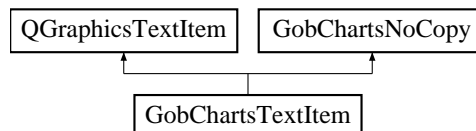
- gobchartspieview.h
- gobchartspieview.cpp

8.8 GobChartsTextItem Class Reference

A convenience text item class that can resize itself.

```
#include <gobchartstextitem.h>
```

Inheritance diagram for GobChartsTextItem:



Public Slots

- void [setRectF](#) (const QRectF &rect)
Sets the dimensions within which the text item's bounding rectangle must fit.

Signals

- void [identity](#) (const QString &)
Broadcast the item's identity.

Public Member Functions

- [GobChartsTextItem](#) (Qt::Orientation orientation, QString uniqueID, QGraphicsItem *parent=0)
Constructor.
- void [setMaxFontSize](#) (const QFont &font)
Set maximum font size.
- void [setAlignment](#) (Qt::Alignment alignment)
Sets the text alignment option (the default is Qt::AlignHCenter).
- Qt::Alignment [alignment](#) () const
Returns the current text alignment.
- void [receiveKeyEvent](#) (QKeyEvent *event)
Receives forwarded keyboard input events.
- QString [getStateXML](#) () const
Returns the label's content and settings as an XML QString.
- bool [setStateXML](#) (const QDomNode &node)
Set the label's values when loading from file.
- void [setPlainText](#) (const QString &text)
Re-implemented from QGraphicsTextItem.

Protected Member Functions

- void [mousePressEvent](#) (QGraphicsSceneMouseEvent *event)
Re-implemented from QGraphicsTextItem.
- void [setFont](#) (const QFont &font)
Re-implemented from QGraphicsTextItem to call `resize()` whenever the font is changed.

8.8.1 Detailed Description

A convenience text item class that can resize itself.

[GobChartsTextItem](#) automatically resizes itself to fit into whichever space is allocated to it and allows for the text alignment to be specified.

Definition at line 38 of file `gobchartstextitem.h`.

8.8.2 Member Function Documentation

8.8.2.1 `Qt::Alignment GobChartsTextItem::alignment () const`

Returns the current text alignment.

See Also

[setAlignment\(\)](#)

Definition at line 190 of file `gobchartstextitem.cpp`.

8.8.2.2 `QString GobChartsTextItem::getStateXML () const`

Returns the label's content and settings as an XML QString.

Charts are saved to file as XML, this function takes all the label's settings, converts it to XML and returns the XML string.

See Also

[setStateXML\(\)](#)

Definition at line 262 of file `gobchartstextitem.cpp`.

8.8.2.3 `void GobChartsTextItem::identity (const QString &) [signal]`

Broadcast the item's identity.

8.8.2.4 `void GobChartsTextItem::mousePressEvent (QGraphicsSceneMouseEvent * event) [protected]`

Re-implemented from `QGraphicsTextItem`.

See Qt API documentation for details.

Definition at line 349 of file `gobchartstextitem.cpp`.

8.8.2.5 `void GobChartsTextItem::receiveKeyEvent (QKeyEvent * event)`

Receives forwarded keyboard input events.

Definition at line 255 of file `gobchartstextitem.cpp`.

8.8.2.6 `void GobChartsTextItem::setAlignment (Qt::Alignment alignment)`

Sets the text alignment option (the default is `Qt::AlignHCenter`).

See Also

[alignment\(\)](#)

Definition at line 179 of file gobchartstextitem.cpp.

8.8.2.7 void GobChartsTextItem::setFont (const QFont & font) [protected]

Re-implemented from QGraphicsTextItem to call resize() whenever the font is changed.

Definition at line 160 of file gobchartstextitem.cpp.

8.8.2.8 void GobChartsTextItem::setMaxFontSize (const QFont & font)

Set maximum font size.

Updates the item's font, but also keeps track of the user selected font size to ensure that, when automatically increasing the text size, the user preference is respected.

Definition at line 152 of file gobchartstextitem.cpp.

8.8.2.9 void GobChartsTextItem::setPlainText (const QString & text)

Re-implemented from QGraphicsTextItem.

See Qt API documentation for details.

Definition at line 168 of file gobchartstextitem.cpp.

8.8.2.10 void GobChartsTextItem::setRectF (const QRectF & rect) [slot]

Sets the dimensions within which the text item's bounding rectangle must fit.

The item will reposition itself to be centred within this given rectangular space (it's position will be calculated in terms of this provided rectangle).

Definition at line 119 of file gobchartstextitem.cpp.

8.8.2.11 bool GobChartsTextItem::setStateXML (const QDomNode & node)

Set the label's values when loading from file.

When loading a label from file, the content of the file is parsed from XML and the QDomNodes containing the necessary information for building the label is provided to this function as parameter.

See Also

[getStateXML\(\)](#)

Definition at line 308 of file gobchartstextitem.cpp.

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

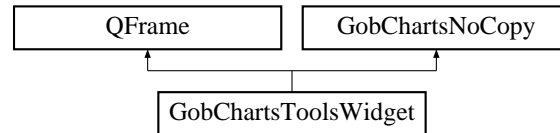
- gobchartstextitem.h
- gobchartstextitem.cpp

8.9 GobChartsToolsWidget Class Reference

Provides the end-user with the tools to select and customise charts.

```
#include <gobchartstoolswidget.h>
```

Inheritance diagram for GobChartsToolsWidget:



Public Slots

- void [receiveLabelDetails](#) (GobChartsLabel label, QString text, QFont font, QColor colour, Qt::Alignment align)
Receives label details.

Signals

- void [clearChart](#) ()
Emitted when the user clicks the "Clear" button.
- void [saveChart](#) ()
Emitted when the user clicks the "Save" button.
- void [loadChart](#) ()
Emitted when the user clicks the "Load" button.
- void [createChart](#) (GobChartsType)
Emitted when the user clicks on any of the chart tool buttons (BAR, PIE or LINE).
- void [setFixedColour](#) (QColor)
Emitted when the user selects a fixed chart colour.
- void [setRandomColours](#) ()
Emitted when the user wishes random colour selection to be re-enabled.
- void [setAllowedDataRange](#) (qreal, qreal)
Emitted when the user chooses to restrict the chart's data range.
- void [setShowTotalRange](#) ()
Emitted when the user chooses to display the entire data range (normally after a previous call to setAllowedDataRange).
- void [setGridLineStyle](#) (Qt::PenStyle)
Emitted when the pen style used to draw the grid lines changes.
- void [setVerticalGridLines](#) (bool, int)
Emitted when the user turns the display of the vertical grid lines "on" or "off" or when the number of vertical grid lines changes.
- void [setHorizontalGridLines](#) (bool, int)
Emitted when the user turns the display of the horizontal grid lines "on" or "off" or when the number of horizontal grid lines changes.
- void [setGridColour](#) (QColor)
Emits a user-selected grid colour.
- void [setLabelDetails](#) (GobChartsLabel, QString, QFont, QColor, Qt::Alignment)
Emitted when a label's details changes.
- void [requestLabelDetails](#) (GobChartsLabel)
Emitted when an update to a label's details is required.

Public Member Functions

- [GobChartsToolsWidget](#) (QWidget *parent=0)
Constructor.
- [~GobChartsToolsWidget](#) ()
Destructor.
- QString [getStateXML](#) () const
Returns the tool widget's content and settings as an XML QString.
- void [setStateXML](#) (GobChartsType type, const QDomNode &toolNode)
Used when saving or loading a chart to or from file.
- void [emitStateSignals](#) ()
Emits all the tool's state signals after loading data from XML.

Friends

- class **LabelDetails**

8.9.1 Detailed Description

Provides the end-user with the tools to select and customise charts.

[GobChartsToolsWidget](#) keeps track of user preferences, emits signals relating to user decisions and updates itself on occasions where the user manipulates the chart directly.

Definition at line 43 of file gobchartstoolswidget.h.

8.9.2 Member Function Documentation

8.9.2.1 void GobChartsToolsWidget::clearChart () [signal]

Emitted when the user clicks the "Clear" button.

8.9.2.2 void GobChartsToolsWidget::createChart (GobChartsType) [signal]

Emitted when the user clicks on any of the chart tool buttons (BAR, PIE or LINE).

8.9.2.3 void GobChartsToolsWidget::emitStateSignals ()

Emits all the tool's state signals after loading data from XML.

See Also

[getStateXML\(\)](#) and [setStateXML\(\)](#)

Definition at line 774 of file gobchartstoolswidget.cpp.

8.9.2.4 QString GobChartsToolsWidget::getStateXML () const

Returns the tool widget's content and settings as an XML QString.

This function takes all the tool settings, converts them to XML and returns the XML string.

See Also

[setStateXML\(\)](#) and [emitStateSignals\(\)](#)

Definition at line 521 of file gobchartstoolswidget.cpp.

8.9.2.5 void GobChartsToolsWidget::loadChart () [signal]

Emitted when the user clicks the "Load" button.

See Also

[saveChart\(\)](#)

8.9.2.6 void GobChartsToolsWidget::receiveLabelDetails (GobChartsLabel *label*, QString *text*, QFont *font*, QColor *colour*, Qt::Alignment *align*) [slot]

Receives label details.

Changing the label that is currently in focus (i.e when the user clicks on any of the three label buttons "H", "X" or "Y"), we wish to update the font in the dropdown, the text alignment, italics, bold and underlined buttons, etc in line with the chosen label. This is how we get that information from the main widget and we request that information be sent via [requestLabelDetails\(\)](#).

Definition at line 468 of file gobchartstoolswidget.cpp.

8.9.2.7 void GobChartsToolsWidget::requestLabelDetails (GobChartsLabel) [signal]

Emitted when an update to a label's details is required.

See Also

[setLabelDetails\(\)](#) and [receiveLabelDetails\(\)](#)

8.9.2.8 void GobChartsToolsWidget::saveChart () [signal]

Emitted when the user clicks the "Save" button.

See Also

[loadChart\(\)](#)

8.9.2.9 void GobChartsToolsWidget::setAllowedDataRange (qreal , qreal) [signal]

Emitted when the user chooses to restrict the chart's data range.

Setting the allowed data range determines which categories will be shown in the final chart (i.e. only those data categories with values within the range will be shown). By default the entire value range is drawn. The first parameter contains the lowest allowed value to be displayed and the second the highest.

See Also

[setShowTotalRange\(\)](#)

8.9.2.10 void GobChartsToolsWidget::setFixedColour (QColor) [signal]

Emitted when the user selects a fixed chart colour.

Fix the chart colour to "colour". This means that the different data categories will all be displayed in "colour" as opposed to having randomly generated (different) colours.

See Also

[setRandomColours\(\)](#)

8.9.2.11 void GobChartsToolsWidget::setGridColour (QColor) [signal]

Emits a user-selected grid colour.

8.9.2.12 void GobChartsToolsWidget::setGridLineStyle (Qt::PenStyle) [signal]

Emitted when the pen style used to draw the grid lines changes.

See Qt API documentation for details on the enum.

8.9.2.13 void GobChartsToolsWidget::setHorizontalGridLines (bool , int) [signal]

Emitted when the user turns the display of the horizontal grid lines "on" or "off" or when the number of horizontal grid lines changes.

8.9.2.14 void GobChartsToolsWidget::setLabelDetails (GobChartsLabel , QString , QFont , QColor , Qt::Alignment) [signal]

Emitted when a label's details changes.

See Also

[requestLabelDetails\(\)](#) and [receiveLabelDetails\(\)](#)

8.9.2.15 void GobChartsToolsWidget::setRandomColours () [signal]

Emitted when the user wishes random colour selection to be re-enabled.

Re-enables random colour generation (the default) for chart items after a call to [setFixedColour\(\)](#).

See Also

[setFixedColour\(\)](#)

8.9.2.16 void GobChartsToolsWidget::setShowTotalRange () [signal]

Emitted when the user chooses to display the entire data range (normally after a previous call to [setAllowedDataRange\(\)](#)).

See Also

[setAllowedDataRange\(\)](#)

8.9.2.17 void GobChartsToolsWidget::setStateXML (GobChartsType type, const QDomNode & toolNode)

Used when saving or loading a chart to or from file.

After reading in the relevant data from the QDomNode, all the relevant state signals are emitted in succession to update the listener with the changes just loaded.

See Also

[getStateXML\(\)](#) and [emitStateSignals\(\)](#)

Definition at line 625 of file gobchartstoolswidget.cpp.

8.9.2.18 void GobChartsToolsWidget::setVerticalGridLines (bool, int) [signal]

Emitted when the user turns the display of the vertical grid lines "on" or "off" or when the number of vertical grid lines changes.

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

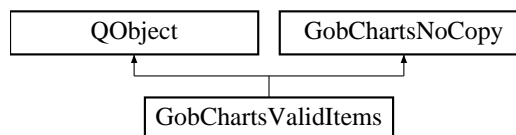
- gobchartstoolswidget.h
- gobchartstoolswidget.cpp

8.10 GobChartsValidItems Class Reference

Controls and manages valid model rows.

```
#include <gobchartsvaliditems.h>
```

Inheritance diagram for GobChartsValidItems:



Public Member Functions

- [GobChartsValidItems](#) (QObject *parent=0)
Constructor.
- virtual [~GobChartsValidItems](#) ()
Destructor.
- void [addValidItem](#) (int row, QString category, qreal data)
Creates and maps a category/value pair to its corresponding row.
- void [clear](#) ()
Delete all valid items from the map.
- const QList< int > [validRows](#) () const
Returns a list of all rows containing valid items.
- QString [category](#) (int row) const
Returns the category corresponding to a specific row.
- qreal [data](#) (int row) const
Returns the value corresponding to a specific row.
- int [size](#) () const
Returns the number of valid items in the map.

8.10.1 Detailed Description

Controls and manages valid model rows.

[GobChartsValidItems](#) keeps track of which data rows and columns contain valid entries (non-empty, legal types, etc) and provides access to the valid values and category names based on the row number.

Definition at line 39 of file gobchartsvaliditems.h.

8.10.2 Member Function Documentation

8.10.2.1 void GobChartsValidItems::addValidItem (int row, QString category, qreal data)

Creates and maps a category/value pair to its corresponding row.

Definition at line 41 of file gobchartsvaliditems.cpp.

8.10.2.2 QString GobChartsValidItems::category (int row) const

Returns the category corresponding to a specific row.

Definition at line 65 of file gobchartsvaliditems.cpp.

8.10.2.3 void GobChartsValidItems::clear ()

Delete all valid items from the map.

Definition at line 51 of file gobchartsvaliditems.cpp.

8.10.2.4 qreal GobChartsValidItems::data (int row) const

Returns the value corresponding to a specific row.

Definition at line 77 of file gobchartsvaliditems.cpp.

8.10.2.5 int GobChartsValidItems::size () const

Returns the number of valid items in the map.

Definition at line 89 of file gobchartsvaliditems.cpp.

8.10.2.6 const QList< int > GobChartsValidItems::validRows () const

Returns a list of all rows containing valid items.

Definition at line 58 of file gobchartsvaliditems.cpp.

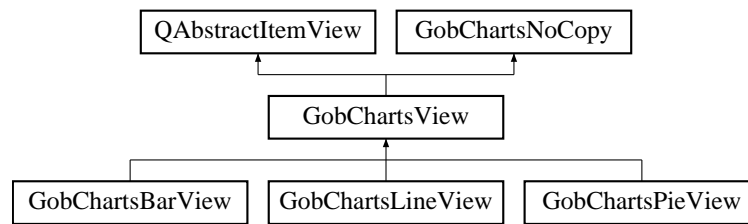
The documentation for this class was generated from the following files:

- gobchartsvaliditems.h
- gobchartsvaliditems.cpp

8.11 GobChartsView Class Reference

```
#include <gobchartsview.h>
```

Inheritance diagram for GobChartsView:



Public Slots

- void [setLabelDetails](#) (GobChartsLabel label, const QString &text, const QFont &font, const QColor &colour, Qt::Alignment align)
Set chart label.
- void [requestLabelDetails](#) (GobChartsLabel label)
Request a specific label's settings.
- void [setGridLineStyle](#) (Qt::PenStyle style)
Grid line style.
- void [setHorizontalGridLines](#) (bool set, int number)
Set horizontal grid lines.
- void [setVerticalGridLines](#) (bool set, int number)
Set horizontal grid lines.
- void [setGridColour](#) (QColor colour)
Set grid colour.
- void [setFixedColour](#) (QColor colour)
Fix the chart colour.
- void [setRandomColours](#) ()
Use different colours.
- void [setAllowedDataRange](#) (qreal lowerBoundary, qreal upperBoundary)
Limit data range.
- void [setShowTotalRange](#) ()
Show total data range.
- void [setDebugLoggingOn](#) (bool logging)
Debug log.

Signals

- void [lastDebugLogMsg](#) (QString)
Debug messages.
- void [emitLabelDetails](#) (GobChartsLabel label, const QString &text, const QFont &font, const QColor &colour, Qt::Alignment align)
Obtain label details.
- void [createLegendItem](#) (const QColor &colour, const QString &text)
Emitted when a new graphics item was created.
- void [clearLegend](#) ()
Emitted when a chart is redrawn and its graphics items are reset.
- void [highLightLegendItem](#) (const QString &text)
Emitted when a graphics item is selected.

Public Member Functions

- virtual [~GobChartsView](#) ()
Constructor.
- void [drawChart](#) ()
Displays the chosen chart type.
- void [legendItemSelected](#) (const QString &text)
Informs the view that a legend of name "text" has been selected.
- void [setStateXML](#) (const QDomNode &viewNode, const QDomNode &dataNode, bool includeData=true)
Set the chart values when loading from file.
- QString [getStateXML](#) (bool includeData=true) const
Returns the chart content and settings as an XML QString.
- QRect [visualRect](#) (const QModelIndex &index) const
Re-implemented from QAbstractItemView.
- QModelIndex [indexAt](#) (const QPoint &point) const
Re-implemented from QAbstractItemView.
- void [scrollTo](#) (const QModelIndex &index, ScrollHint hint=EnsureVisible)
Re-implemented from QAbstractItemView.
- void [setModel](#) (QAbstractItemModel *model)
Re-implemented from QAbstractItemView.

Protected Types

- enum [GobChartsColumn](#) { **CATEGORY**, **VALUE** }
Expected data model columns.

Protected Slots

- void [debugLog](#) (QString msg)
Log debug messages.
- void [dataChanged](#) (const QModelIndex &topLeft, const QModelIndex &bottomRight)
Re-implemented from QAbstractItemView.
- void [rowsInserted](#) (const QModelIndex &parent, int start, int end)
Re-implemented from QAbstractItemView.
- void [rowsAboutToBeRemoved](#) (const QModelIndex &parent, int start, int end)
Re-implemented from QAbstractItemView.
- void [mousePressEvent](#) (QMouseEvent *p_event)
Re-implemented from QAbstractItemView.

Protected Member Functions

- [GobChartsView](#) (QWidget *parent=0)
Destructor.
- virtual void [generateGraphicsItems](#) ()=0
Derived classes must implement this function.
- virtual bool [needsGrid](#) () const =0
Derived classes must implement this function.
- virtual QString [typeInteger](#) () const =0
Derived classes must implement this function.
- bool [useFixedColour](#) () const

- Returns "true" if the chart colour is fixed or "false" if random colours must be generated.*
- QColor [fixedColour](#) () const
 - If the chart colour is fixed, this function returns the colour used to draw the chart items.*
- void [addToGraphItemsContainer](#) (const QModelIndex &valueIndex, QGraphicsItem *item, const QString &legendText)
 - Stores dynamically created graphics items.*
- int [nrValidItems](#) () const
 - Returns the number of valid rows in the data model.*
- QList< int > [validRowList](#) () const
 - Returns a list of valid data row numbers.*
- QString [category](#) (int row) const
 - Returns the category name corresponding to "row" in [validRowList\(\)](#)*
- qreal [value](#) (int row) const
 - Returns the data value corresponding to "row" in [validRowList\(\)](#)*
- const QRectF & [innerSceneRectF](#) () const
 - Returns the rectangle corresponding to the "inner scene".*
- bool [isWithinAllowedRange](#) (qreal [value](#)) const
 - Determines whether or not a data value falls within the allowed data range.*
- qreal [gridWidth](#) () const
 - Returns the grid width.*
- qreal [stripSpace](#) (qreal perc) const
 - Calculates the amount of space at the top of the view that is "extra".*
- qreal [totalValue](#) () const
 - Returns the total, consolidated value of all category values.*
- QModelIndex [moveCursor](#) (QAbstractItemView::CursorAction cursorAction, Qt::KeyboardModifiers modifiers)
 - Re-implemented from QAbstractItemView.*
- QRegion [visualRegionForSelection](#) (const QItemSelection &selection) const
 - Re-implemented from QAbstractItemView.*
- int [horizontalOffset](#) () const
 - Re-implemented from QAbstractItemView.*
- int [verticalOffset](#) () const
 - Re-implemented from QAbstractItemView.*
- bool [isIndexHidden](#) (const QModelIndex &index) const
 - Re-implemented from QAbstractItemView.*
- void [setSelection](#) (const QRect &rect, QItemSelectionModel::SelectionFlags command)
 - Re-implemented from QAbstractItemView.*
- void [keyPressEvent](#) (QKeyEvent *event)
 - Re-implemented from QAbstractItemView.*

Friends

- class [GobChartsViewPrivate](#)
 - The PIMPL idiom will help with future binary compatibility.*

8.11.1 Detailed Description

Derived classes must implement the following pure virtual functions:

1. [generateGraphicsItems\(\)](#) - creates the graphics items required for the specific chart type and positions them within the view.
1. [needsGrid\(\)](#) - if the chart type supports grids, this function must return "true" (e.g. PIE charts don't support chart grids whereas BAR charts do).
1. [typeInteger\(\)](#) - returns an integer value as a QString. The value thus returned must match one of the GobChartsType enum values.

Definition at line 60 of file gobchartsview.h.

8.11.2 Member Enumeration Documentation

8.11.2.1 enum GobChartsView::GobChartsColumn [protected]

Expected data model columns.

[GobChartsWidget](#) is only capable of displaying one to one mappings between categories and data values and expects the first column of the data model to contain the chart's category names and the second the corresponding data values.

Definition at line 193 of file gobchartsview.h.

8.11.3 Constructor & Destructor Documentation

8.11.3.1 GobChartsView::~GobChartsView () [virtual]

Constructor.

Public virtual constructor so that we may delete "new"-ed derived objects from pointer-to-base. [GobChartsFactory](#) and the underlying strategy pattern depend on this for run-time type polymorphism.

Definition at line 356 of file gobchartsview.cpp.

8.11.4 Member Function Documentation

8.11.4.1 void GobChartsView::addToGraphItemsContainer (const QModelIndex & *valueIndex*, QGraphicsItem * *item*, const QString & *legendText*) [protected]

Stores dynamically created graphics items.

This function adds newly created graphics items to a map so that they can be retrieved based on their corresponding model indices later on.

Parameters

<i>valueIndex</i>	- the QModelIndex that the graphics item represents.
<i>item</i>	- the newly created graphics item.

Definition at line 705 of file gobchartsview.cpp.

8.11.4.2 QString GobChartsView::category (int row) const [protected]

Returns the category name corresponding to "row" in [validRowList\(\)](#)

See Also

[nrValidItems\(\)](#) and [value\(\)](#)

Definition at line 733 of file gobchartsview.cpp.

8.11.4.3 void GobChartsView::clearLegend () [signal]

Emitted when a chart is redrawn and its graphics items are reset.

8.11.4.4 void GobChartsView::createLegendItem (const QColor & colour, const QString & text) [signal]

Emitted when a new graphics item was created.

This signal is emitted shortly after a graphics item is created and contains the details for the corresponding legend item.

8.11.4.5 void GobChartsView::dataChanged (const QModelIndex & topLeft, const QModelIndex & bottomRight) [protected], [slot]

Re-implemented from QAbstractItemView.

See the Qt API documentation for details.

Definition at line 778 of file gobchartsview.cpp.

8.11.4.6 void GobChartsView::debugLog (QString msg) [protected], [slot]

Log debug messages.

This slot receives logging information from privately owned objects.

Definition at line 761 of file gobchartsview.cpp.

8.11.4.7 void GobChartsView::drawChart ()

Displays the chosen chart type.

Generates and displays the chart within the confines of the QGraphicsView which contains it.

Definition at line 434 of file gobchartsview.cpp.

8.11.4.8 void GobChartsView::emitLabelDetails (GobChartsLabel label, const QString & text, const QFont & font, const QColor & colour, Qt::Alignment align) [signal]

Obtain label details.

This signal is emitted when [requestLabelDetails\(\)](#) is called and contains all the information relevant to the specified label.

See Also

[requestLabelDetails\(\)](#) and [setLabelDetails\(\)](#)

8.11.4.9 QColor GobChartsView::fixedColour () const [protected]

If the chart colour is fixed, this function returns the colour used to draw the chart items.

See Also

[setFixedColour\(\)](#), [setRandomColours](#) and [useFixedColour\(\)](#)

Definition at line 657 of file gobchartsview.cpp.

8.11.4.10 virtual void GobChartsView::generateGraphicsItems () [protected], [pure virtual]

Derived classes must implement this function.

This function must create the graphics items required for the specific chart type and position them within the confines of the view.

Implemented in [GobChartsBarView](#), [GobChartsLineView](#), and [GobChartsPieView](#).

8.11.4.11 QString GobChartsView::getStateXML (bool *includeData* = true) const

Returns the chart content and settings as an XML QString.

Charts are saved to file as XML, this function takes all the view settings, converts it to XML and returns the XML string.

Parameters

<i>includeData</i>	- specifies whether or not the chart data must also be saved (if "false", then only the view settings are provided in the XML QString).
--------------------	---

See Also

[setStateXML\(\)](#)

Definition at line 568 of file gobchartsview.cpp.

8.11.4.12 qreal GobChartsView::gridWidth () const [protected]

Returns the grid width.

Within the context of the allowed space, this is the width that the grid may occupy.

Definition at line 712 of file gobchartsview.cpp.

8.11.4.13 void GobChartsView::highLightLegendItem (const QString & *text*) [signal]

Emitted when a graphics item is selected.

8.11.4.14 int GobChartsView::horizontalOffset () const [protected]

Re-implemented from QAbstractItemView.

See the Qt API documentation for details.

Definition at line 902 of file gobchartsview.cpp.

8.11.4.15 `QModelIndex GobChartsView::indexAt (const QPoint & point) const`

Re-implemented from QAbstractItemView.

See the Qt API documentation for details.

Definition at line 847 of file gobchartsview.cpp.

8.11.4.16 `const QRectF & GobChartsView::innerSceneRectF () const` `[protected]`

Returns the rectangle corresponding to the "inner scene".

The "inner scene" is the rectangle defined as the space within which the chart items must fit. Within the context of the entire view, this means that the space taken up by labels is excluded.

Definition at line 747 of file gobchartsview.cpp.

8.11.4.17 `bool GobChartsView::isIndexHidden (const QModelIndex & index) const` `[protected]`

Re-implemented from QAbstractItemView.

See the Qt API documentation for details.

Definition at line 916 of file gobchartsview.cpp.

8.11.4.18 `bool GobChartsView::isWithinAllowedRange (qreal value) const` `[protected]`

Determines whether or not a data value falls within the allowed data range.

See Also

[setAllowedDataRange\(\)](#) and [setShowTotalRange\(\)](#)

Definition at line 691 of file gobchartsview.cpp.

8.11.4.19 `void GobChartsView::keyPressEvent (QKeyEvent * event)` `[protected]`

Re-implemented from QAbstractItemView.

See the Qt API documentation for details.

Definition at line 987 of file gobchartsview.cpp.

8.11.4.20 `void GobChartsView::lastDebugLogMsg (QString)` `[signal]`

Debug messages.

This signal broadcasts the last debug log message and will only be emitted when debug logging is turned "on".

See Also

[setDebugLoggingOn\(\)](#)

8.11.4.21 `void GobChartsView::legendItemSelected (const QString & text)`

Informs the view that a legend of name "text" has been selected.

Definition at line 463 of file gobchartsview.cpp.

8.11.4.22 `void GobChartsView::mousePressEvent (QMouseEvent * p_event)` [protected],[slot]

Re-implemented from QAbstractItemView.

See the Qt API documentation for details.

Definition at line 979 of file gobchartsview.cpp.

8.11.4.23 `QModelIndex GobChartsView::moveCursor (QAbstractItemView::CursorAction cursorAction, Qt::KeyboardModifiers modifiers)` [protected]

Re-implemented from QAbstractItemView.

See the Qt API documentation for details.

Definition at line 859 of file gobchartsview.cpp.

8.11.4.24 `virtual bool GobChartsView::needsGrid () const` [protected],[pure virtual]

Derived classes must implement this function.

This function must return "true" when the chart type supports grid lines and "false" otherwise (e.g. PIE charts don't support chart grids whereas BAR charts do).

Implemented in [GobChartsBarView](#), [GobChartsLineView](#), and [GobChartsPieView](#).

8.11.4.25 `int GobChartsView::nrValidItems () const` [protected]

Returns the number of valid rows in the data model.

"Valid rows" are rows defined as those containing chart categories or data values and excludes empty rows).

See Also

[validRowList\(\)](#), [category\(\)](#) and [value\(\)](#)

Definition at line 719 of file gobchartsview.cpp.

8.11.4.26 `void GobChartsView::requestLabelDetails (GobChartsLabel label)` [slot]

Request a specific label's settings.

Calling this function will result in the [emitLabelDetails\(\)](#) signal being emitted. The signal's parameters contain all the information for the relevant label - text, font, colour and alignment.

Parameters

<i>label</i>	- the header, x-axis label or y-axis label.
--------------	---

See Also

[setLabelDetails\(\)](#) and [emitLabelDetails\(\)](#)

Definition at line 377 of file gobchartsview.cpp.

8.11.4.27 `void GobChartsView::rowsAboutToBeRemoved (const QModelIndex & parent, int start, int end)` [protected],[slot]

Re-implemented from QAbstractItemView.

See the Qt API documentation for details.

Definition at line 801 of file gobchartsview.cpp.

8.11.4.28 void GobChartsView::rowsInserted (const QModelIndex & *parent*, int *start*, int *end*) [protected], [slot]

Re-implemented from QAbstractItemView.

See the Qt API documentation for details.

Definition at line 792 of file gobchartsview.cpp.

8.11.4.29 void GobChartsView::scrollTo (const QModelIndex & *index*, ScrollHint *hint* = EnsureVisible)

Re-implemented from QAbstractItemView.

See the Qt API documentation for details.

Definition at line 823 of file gobchartsview.cpp.

8.11.4.30 void GobChartsView::setAllowedDataRange (qreal *lowerBoundary*, qreal *upperBoundary*) [slot]

Limit data range.

Setting the allowed data range determines which categories will be shown in the final chart (i.e. only those data categories with values within the range will be shown). By default the entire value range is drawn.

Parameters

<i>lowerBoundary</i>	- the lowest allowed value to be displayed.
<i>upperBoundary</i>	- the highest allowed value to be displayed.

See Also

[setShowTotalRange\(\)](#) and [isWithinAllowedRange\(\)](#)

Definition at line 673 of file gobchartsview.cpp.

8.11.4.31 void GobChartsView::setDebugLoggingOn (bool *logging*) [slot]

Debug log.

Turn debug logging messages "on" or "off" (default "off").

See Also

[lastDebugLogMsg\(\)](#)

Definition at line 664 of file gobchartsview.cpp.

8.11.4.32 void GobChartsView::setFixedColour (QColor *colour*) [slot]

Fix the chart colour.

Fix the chart colour to "colour". This means that the different data categories will all be displayed in "colour" as opposed to having randomly generated (different) colours.

See Also

[setRandomColours\(\)](#), [useFixedColour\(\)](#) and [fixedColour\(\)](#)

Definition at line 624 of file gobchartsview.cpp.

8.11.4.33 void GobChartsView::setGridColour (QColor *colour*) [slot]

Set grid colour.

Sets the grid line's colour to "colour".

Definition at line 426 of file gobchartsview.cpp.

8.11.4.34 void GobChartsView::setGridLineStyle (Qt::PenStyle *style*) [slot]

Grid line style.

Set the pen style used to draw the grid lines.

Parameters

<i>style</i>	- dashed, dotted, solid, etc. (see Qt API documentation for details on the enum).
--------------	---

Definition at line 402 of file gobchartsview.cpp.

8.11.4.35 void GobChartsView::setHorizontalGridLines (bool *set*, int *number*) [slot]

Set horizontal grid lines.

Turn horizontal grid lines "on" or "off" and also sets the number of grid lines.

Definition at line 410 of file gobchartsview.cpp.

8.11.4.36 void GobChartsView::setLabelDetails (GobChartsLabel *label*, const QString & *text*, const QFont & *font*, const QColor & *colour*, Qt::Alignment *align*) [slot]

Set chart label.

Sets the text, font, colour and alignment of the label specified.

Parameters

<i>label</i>	- the header, x-axis label or y-axis label.
--------------	---

See Also

[requestLabelDetails\(\)](#) and [emitLabelDetails](#)

Definition at line 363 of file gobchartsview.cpp.

8.11.4.37 void GobChartsView::setModel (QAbstractItemModel * *model*)

Re-implemented from QAbstractItemView.

See the Qt API documentation for details.

Definition at line 831 of file gobchartsview.cpp.

8.11.4.38 void GobChartsView::setRandomColours () [slot]

Use different colours.

Re-enables random colour generation (the default) for chart items after a call to [setFixedColour\(\)](#).

See Also

[setFixedColour\(\)](#), [useFixedColour\(\)](#) and [fixedColour\(\)](#)

Definition at line 642 of file gobchartsview.cpp.

8.11.4.39 void GobChartsView::setSelection (const QRect & rect, QItemSelectionModel::SelectionFlags command) [protected]

Re-implemented from QAbstractItemView.

See the Qt API documentation for details.

Definition at line 924 of file gobchartsview.cpp.

8.11.4.40 void GobChartsView::setShowTotalRange () [slot]

Show total data range.

Displays all categories (i.e. the entire data value range).

See Also

[setAllowedDataRange\(\)](#) and [isWithinAllowedRange\(\)](#)

Definition at line 683 of file gobchartsview.cpp.

8.11.4.41 void GobChartsView::setStateXML (const QDomNode & viewNode, const QDomNode & dataNode, bool includeData = true)

Set the chart values when loading from file.

When loading a chart from file, the content of the file is parsed from XML and the QDomNodes containing the necessary information for building the chart is provided to this function as parameters.

Parameters

<i>viewNode</i>	- the QDomNode containing settings specific to the chart view itself.
<i>dataNode</i>	- the QDomNode containing the data read from the saved file.
<i>includeData</i>	- whether or not the dataNode's content must be read (if "false", it will be ignored).

See Also

[getStateXML\(\)](#)

Definition at line 475 of file gobchartsview.cpp.

8.11.4.42 void GobChartsView::setVerticalGridLines (bool set, int number) [slot]

Set horizontal grid lines.

Turn vertical grid lines "on" or "off" and also sets the number of grid lines.

Definition at line 418 of file gobchartsview.cpp.

8.11.4.43 `qreal GobChartsView::stripSpace (qreal perc) const` [protected]

Calculates the amount of space at the top of the view that is "extra".

This extra space can be stripped from all item height calculations and is needed to ensure that as much space is used for the display of the chart as possible.

Definition at line 610 of file `gobchartsview.cpp`.

8.11.4.44 `qreal GobChartsView::totalValue () const` [protected]

Returns the total, consolidated value of all category values.

Definition at line 754 of file `gobchartsview.cpp`.

8.11.4.45 `virtual QString GobChartsView::typeInteger () const` [protected], [pure virtual]

Derived classes must implement this function.

This function must return an integer value that matches one of the `GobChartsType` enum values as a `QString` (this information is used when saving and loading charts to and from file).

Implemented in [GobChartsBarView](#), [GobChartsLineView](#), and [GobChartsPieView](#).

8.11.4.46 `bool GobChartsView::useFixedColour () const` [protected]

Returns "true" if the chart colour is fixed or "false" if random colours must be generated.

See Also

[setFixedColour\(\)](#), [setRandomColours](#) and [fixedColour\(\)](#)

Definition at line 650 of file `gobchartsview.cpp`.

8.11.4.47 `QList< int > GobChartsView::validRowList () const` [protected]

Returns a list of valid data row numbers.

See Also

[nrValidItems\(\)](#), [category\(\)](#) and [value\(\)](#)

Definition at line 726 of file `gobchartsview.cpp`.

8.11.4.48 `qreal GobChartsView::value (int row) const` [protected]

Returns the data value corresponding to "row" in [validRowList\(\)](#)

See Also

[nrValidItems\(\)](#) and [category\(\)](#)

Definition at line 740 of file `gobchartsview.cpp`.

8.11.4.49 `int GobChartsView::verticalOffset () const` `[protected]`

Re-implemented from `QAbstractItemView`.

See the Qt API documentation for details.

Definition at line 909 of file `gobchartsview.cpp`.

8.11.4.50 `QRect GobChartsView::visualRect (const QModelIndex & index) const`

Re-implemented from `QAbstractItemView`.

See the Qt API documentation for details.

Definition at line 810 of file `gobchartsview.cpp`.

8.11.4.51 `QRegion GobChartsView::visualRegionForSelection (const QItemSelection & selection) const` `[protected]`

Re-implemented from `QAbstractItemView`.

See the Qt API documentation for details.

Definition at line 1012 of file `gobchartsview.cpp`.

8.11.5 Friends And Related Function Documentation

8.11.5.1 `friend class GobChartsViewPrivate` `[friend]`

The PIMPL idiom will help with future binary compatibility.

Since the `GobChartsViewPrivate` class is completely restricted to this class, the "friend" status does not cause any adverse coupling (the class is basically a container of all the data members that would have been declared as "private" anyway, with one or two utility methods thrown in).

Definition at line 308 of file `gobchartsview.h`.

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

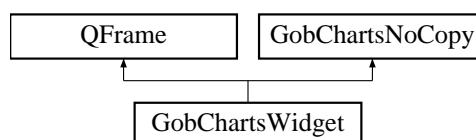
- `gobchartsview.h`
- `gobchartsview.cpp`

8.12 GobChartsWidget Class Reference

Main widget class.

```
#include <gobchartswidget.h>
```

Inheritance diagram for `GobChartsWidget`:



Signals

- void [lastDebugLogMsg](#) (`QString`)

Emits the last debug log message.

Public Member Functions

- [GobChartsWidget](#) (QWidget *parent=0)
Constructor.
- [~GobChartsWidget](#) ()
Destructor.
- void [setModel](#) (QAbstractItemModel *model)
Set the item model ([GobChartsWidget](#) does NOT take ownership).
- void [setSelectionModel](#) (QItemSelectionModel *selectionModel)
Set the item selection model ([GobChartsWidget](#) does NOT take ownership).
- void [setDebugLoggingOn](#) (bool log)
Turn debug logging messages "on" or "off".

8.12.1 Detailed Description

Main widget class.

As an application developer, this is the only class that really needs to concern you (unless, of course, you are interested in changing any of the underlying functionality).

[GobChartsWidget](#) will provide you with a final, usable widget that contains all the necessary GUI tools for an end-user to create and customise any of the available chart types (BAR, PIE or LINE). All you need to do is to create an instance of [GobChartsWidget](#) and inform it of your item and selection models via [setModel\(\)](#) and [setSelectionModel\(\)](#). It is important to note that [GobChartsWidget](#) does NOT take ownership of either of the models set on it.

VERY IMPORTANT: The data model you use **must** contain your data categories in its first column and the corresponding values in the second. Furthermore, [GobChartsWidget](#) can only handle one-to-one mappings of categories to data, i.e. you cannot draw charts with data values spanning multiple columns and, finally, all negative data values are automatically rounded up to zero. The reason for this is that pie charts do not lend themselves to the display of multiple data sets per category, nor the display of negative numbers and I decided to exclude these possibilities altogether in favour of a design allowing for greater flexibility at run-time and fewer complications with regards to the implementation.

Finally, if you would like to see what [GobChartsWidget](#) gets up to in the background, you can turn debug logging on via [setDebugLoggingOn\(\)](#) and connect to [GobChartsWidget](#)'s [lastDebugLogMsg\(\)](#) signal in order to receive the log output.

Definition at line 124 of file gobchartswidget.h.

8.12.2 Member Function Documentation

8.12.2.1 void GobChartsWidget::lastDebugLogMsg (QString) [signal]

Emits the last debug log message.

See Also

[setDebugLoggingOn\(\)](#)

8.12.2.2 void GobChartsWidget::setDebugLoggingOn (bool log)

Turn debug logging messages "on" or "off".

See Also

[lastDebugLogMsg\(\)](#)

Definition at line 161 of file gobchartswidget.cpp.

8.12.2.3 void GobChartsWidget::setModel (QAbstractItemModel * *model*)

Set the item model ([GobChartsWidget](#) does NOT take ownership).

Definition at line 174 of file gobchartswidget.cpp.

8.12.2.4 void GobChartsWidget::setSelectionModel (QItemSelectionModel * *selectionModel*)

Set the item selection model ([GobChartsWidget](#) does NOT take ownership).

Definition at line 184 of file gobchartswidget.cpp.

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

- gobchartswidget.h
- gobchartswidget.cpp

Index

- ~GobChartsView
 - GobChartsView, [40](#)
- addGridToScene
 - GobChartsGrid, [22](#)
- addItem
 - GobChartsGraphItems, [19](#)
- addItemToScene
 - GobChartsGraphItems, [19](#)
- addToGraphItemsContainer
 - GobChartsView, [40](#)
- addValidItem
 - GobChartsValidItems, [36](#)
- alignment
 - GobChartsTextItem, [29](#)
- category
 - GobChartsValidItems, [36](#)
 - GobChartsView, [40](#)
- Chart Views, [11](#)
- clear
 - GobChartsValidItems, [36](#)
- clearChart
 - GobChartsToolsWidget, [32](#)
- clearLegend
 - GobChartsView, [41](#)
- clearSelection
 - GobChartsGraphItems, [19](#)
- constructGrid
 - GobChartsGrid, [22](#)
- createChart
 - GobChartsFactory, [17](#)
 - GobChartsToolsWidget, [32](#)
- createLegendItem
 - GobChartsView, [41](#)
- data
 - GobChartsValidItems, [36](#)
- dataChanged
 - GobChartsView, [41](#)
- debugLog
 - GobChartsView, [41](#)
- deleteItems
 - GobChartsGraphItems, [19](#)
- drawChart
 - GobChartsView, [41](#)
- emitLabelDetails
 - GobChartsView, [41](#)
- emitStateSignals
 - GobChartsToolsWidget, [32](#)
- fixedColour
 - GobChartsView, [41](#)
- generateGraphicsItems
 - GobChartsBarView, [16](#)
 - GobChartsLineView, [25](#)
 - GobChartsPieView, [27](#)
 - GobChartsView, [42](#)
- getInstance
 - GobChartsFactory, [17](#)
- getItemLegend
 - GobChartsGraphItems, [19](#)
- getItemRectF
 - GobChartsGraphItems, [19](#)
- getModelIndex
 - GobChartsGraphItems, [20](#)
- getNextColour
 - GobChartsColours, [13](#)
- getStateXML
 - GobChartsTextItem, [29](#)
 - GobChartsToolsWidget, [32](#)
 - GobChartsView, [42](#)
- GobChartsBarView, [15](#)
 - generateGraphicsItems, [16](#)
 - needsGrid, [16](#)
 - typeInteger, [16](#)
- GobChartsColours, [13](#)
 - getNextColour, [13](#)
 - resetColourIndex, [13](#)
- GobChartsColumn
 - GobChartsView, [40](#)
- GobChartsFactory, [16](#)
 - createChart, [17](#)
 - getInstance, [17](#)
- GobChartsGraphItems, [17](#)
 - addItem, [19](#)
 - addItemToScene, [19](#)
 - clearSelection, [19](#)
 - deleteItems, [19](#)
 - getItemLegend, [19](#)
 - getItemRectF, [19](#)
 - getModelIndex, [20](#)
 - lastDebugLogMsg, [20](#)
 - removeItemsFromScene, [20](#)
 - setDebugLoggingOn, [20](#)
 - setSelected, [20](#)
- GobChartsGrid, [21](#)
 - addGridToScene, [22](#)

- constructGrid, 22
- gridWidth, 22
- lastDebugLogMsg, 22
- removeGridFromScene, 22
- setDebugLoggingOn, 23
- setGridColour, 23
- setGridLineStyle, 23
- setGridRectF, 23
- setHorizontalGridLines, 23
- setVerticalGridLines, 23
- GobChartsLineView, 24
 - generateGraphicsItems, 25
 - needsGrid, 25
 - typeInteger, 25
- GobChartsNoCopy, 25
- GobChartsPieView, 26
 - generateGraphicsItems, 27
 - needsGrid, 27
 - typeInteger, 27
- GobChartsTextItem, 28
 - alignment, 29
 - getStateXML, 29
 - identity, 29
 - mousePressEvent, 29
 - receiveKeyEvent, 29
 - setAlignment, 29
 - setFont, 30
 - setMaxFontSize, 30
 - setPlainText, 30
 - setRectF, 30
 - setStateXML, 30
- GobChartsToolsWidget, 31
 - clearChart, 32
 - createChart, 32
 - emitStateSignals, 32
 - getStateXML, 32
 - loadChart, 33
 - receiveLabelDetails, 33
 - requestLabelDetails, 33
 - saveChart, 33
 - setAllowedDataRange, 33
 - setFixedColour, 33
 - setGridColour, 34
 - setGridLineStyle, 34
 - setHorizontalGridLines, 34
 - setLabelDetails, 34
 - setRandomColours, 34
 - setShowTotalRange, 34
 - setStateXML, 34
 - setVerticalGridLines, 35
- GobChartsValidItems, 35
 - addValidItem, 36
 - category, 36
 - clear, 36
 - data, 36
 - size, 36
 - validRows, 36
- GobChartsView, 36
 - ~GobChartsView, 40
 - addToGraphItemsContainer, 40
 - category, 40
 - clearLegend, 41
 - createLegendItem, 41
 - dataChanged, 41
 - debugLog, 41
 - drawChart, 41
 - emitLabelDetails, 41
 - fixedColour, 41
 - generateGraphicsItems, 42
 - getStateXML, 42
 - GobChartsColumn, 40
 - GobChartsViewPrivate, 49
 - gridWidth, 42
 - highLightLegendItem, 42
 - horizontalOffset, 42
 - indexAt, 42
 - innerSceneRectF, 43
 - isIndexHidden, 43
 - isWithinAllowedRange, 43
 - keyPressEvent, 43
 - lastDebugLogMsg, 43
 - legendItemSelected, 43
 - mousePressEvent, 43
 - moveCursor, 44
 - needsGrid, 44
 - nrValidItems, 44
 - requestLabelDetails, 44
 - rowsAboutToBeRemoved, 44
 - rowsInserted, 45
 - scrollTo, 45
 - setAllowedDataRange, 45
 - setDebugLoggingOn, 45
 - setFixedColour, 45
 - setGridColour, 46
 - setGridLineStyle, 46
 - setHorizontalGridLines, 46
 - setLabelDetails, 46
 - setModel, 46
 - setRandomColours, 46
 - setSelection, 47
 - setShowTotalRange, 47
 - setStateXML, 47
 - setVerticalGridLines, 47
 - stripSpace, 47
 - totalValue, 48
 - typeInteger, 48
 - useFixedColour, 48
 - validRowList, 48
 - value, 48
 - verticalOffset, 48
 - visualRect, 49
 - visualRegionForSelection, 49
- GobChartsViewPrivate
 - GobChartsView, 49
- GobChartsWidget, 49
 - lastDebugLogMsg, 50

- setDebugLoggingOn, 50
 - setModel, 51
 - setSelectionModel, 51
- gridWidth
 - GobChartsGrid, 22
 - GobChartsView, 42
- highLightLegendItem
 - GobChartsView, 42
- horizontalOffset
 - GobChartsView, 42
- identity
 - GobChartsTextItem, 29
- indexAt
 - GobChartsView, 42
- innerSceneRectF
 - GobChartsView, 43
- isIndexHidden
 - GobChartsView, 43
- isWithinAllowedRange
 - GobChartsView, 43
- keyPressEvent
 - GobChartsView, 43
- lastDebugLogMsg
 - GobChartsGraphItems, 20
 - GobChartsGrid, 22
 - GobChartsView, 43
 - GobChartsWidget, 50
- legendItemSelected
 - GobChartsView, 43
- loadChart
 - GobChartsToolsWidget, 33
- mousePressEvent
 - GobChartsTextItem, 29
 - GobChartsView, 43
- moveCursor
 - GobChartsView, 44
- needsGrid
 - GobChartsBarView, 16
 - GobChartsLineView, 25
 - GobChartsPieView, 27
 - GobChartsView, 44
- nrValidItems
 - GobChartsView, 44
- receiveKeyEvent
 - GobChartsTextItem, 29
- receiveLabelDetails
 - GobChartsToolsWidget, 33
- removeGridFromScene
 - GobChartsGrid, 22
- removeItemsFromScene
 - GobChartsGraphItems, 20
- requestLabelDetails
 - GobChartsToolsWidget, 33
- GobChartsView, 44
- resetColourIndex
 - GobChartsColours, 13
- rowsAboutToBeRemoved
 - GobChartsView, 44
- rowsInserted
 - GobChartsView, 45
- saveChart
 - GobChartsToolsWidget, 33
- scrollTo
 - GobChartsView, 45
- setAlignment
 - GobChartsTextItem, 29
- setAllowedDataRange
 - GobChartsToolsWidget, 33
 - GobChartsView, 45
- setDebugLoggingOn
 - GobChartsGraphItems, 20
 - GobChartsGrid, 23
 - GobChartsView, 45
 - GobChartsWidget, 50
- setFixedColour
 - GobChartsToolsWidget, 33
 - GobChartsView, 45
- setFont
 - GobChartsTextItem, 30
- setGridColour
 - GobChartsGrid, 23
 - GobChartsToolsWidget, 34
 - GobChartsView, 46
- setGridLineStyle
 - GobChartsGrid, 23
 - GobChartsToolsWidget, 34
 - GobChartsView, 46
- setGridRectF
 - GobChartsGrid, 23
- setHorizontalGridLines
 - GobChartsGrid, 23
 - GobChartsToolsWidget, 34
 - GobChartsView, 46
- setLabelDetails
 - GobChartsToolsWidget, 34
 - GobChartsView, 46
- setMaxFontSize
 - GobChartsTextItem, 30
- setModel
 - GobChartsView, 46
 - GobChartsWidget, 51
- setPlainText
 - GobChartsTextItem, 30
- setRandomColours
 - GobChartsToolsWidget, 34
 - GobChartsView, 46
- setRectF
 - GobChartsTextItem, 30
- setSelected
 - GobChartsGraphItems, 20
- setSelection

- GobChartsView, [47](#)
- setSelectionMode
 - GobChartsWidget, [51](#)
- setShowTotalRange
 - GobChartsToolsWidget, [34](#)
 - GobChartsView, [47](#)
- setStateXML
 - GobChartsTextItem, [30](#)
 - GobChartsToolsWidget, [34](#)
 - GobChartsView, [47](#)
- setVerticalGridLines
 - GobChartsGrid, [23](#)
 - GobChartsToolsWidget, [35](#)
 - GobChartsView, [47](#)
- size
 - GobChartsValidItems, [36](#)
- stripSpace
 - GobChartsView, [47](#)
- totalValue
 - GobChartsView, [48](#)
- typeInteger
 - GobChartsBarView, [16](#)
 - GobChartsLineView, [25](#)
 - GobChartsPieView, [27](#)
 - GobChartsView, [48](#)
- useFixedColour
 - GobChartsView, [48](#)
- validRowList
 - GobChartsView, [48](#)
- validRows
 - GobChartsValidItems, [36](#)
- value
 - GobChartsView, [48](#)
- verticalOffset
 - GobChartsView, [48](#)
- visualRect
 - GobChartsView, [49](#)
- visualRegionForSelection
 - GobChartsView, [49](#)