

EPICS QT Framework

1.1.5

Generated by Doxygen 1.7.3

Mon Jul 18 2011 12:12:49

Contents

1	QCa - EPICS aware Qt Widgets	1
1.1	Documentation	1
1.2	License	1
1.3	Platforms	2
1.4	Screenshots	2
1.5	Downloads	2
1.6	Installation	2
1.7	Support	2
1.8	Related Projects	2
1.9	Credits:	3
2	GNU General Public License	5
3	INSTALL	7
4	ASgui screen shots	9
5	other applications using epicsqt widgets	15
6	Qt Designer	17
7	Qt Creator	19
8	Namespace Index	21
8.1	Namespace List	21
9	Class Index	23
9.1	Class Hierarchy	23
10	Class Index	27
10.1	Class List	27
11	Namespace Documentation	31
11.1	qcaobject Namespace Reference	31
11.1.1	Detailed Description	31
12	Class Documentation	33
12.1	ASguiForm Class Reference	33
12.1.1	Detailed Description	34
12.2	ASguiFormPlugin Class Reference	35

12.2.1 Detailed Description	35
12.2.2 Property Documentation	36
12.2.2.1 uiFile	36
12.3 ASguiFormPluginManager Class Reference	36
12.3.1 Detailed Description	36
12.4 CmdPushButtonPlugin Class Reference	37
12.4.1 Property Documentation	38
12.4.1.1 variableSubstitutions	38
12.5 CmdPushButtonPluginManager Class Reference	38
12.5.1 Detailed Description	38
12.6 qcastatemachine::ConnectionQCaStateMachine Class Reference	39
12.7 ContainerProfile Class Reference	39
12.7.1 Detailed Description	41
12.8 QCaPeriodic::elementInfoStruct Reference	41
12.9 GuiPushButtonPlugin Class Reference	42
12.9.1 Detailed Description	42
12.10GuiPushButtonPluginManager Class Reference	43
12.10.1 Detailed Description	44
12.11Link Class Reference	44
12.11.1 Detailed Description	46
12.12LinkPlugin Class Reference	46
12.12.1 Detailed Description	47
12.13LinkPluginManager Class Reference	48
12.13.1 Detailed Description	48
12.14localEnumerationItem Class Reference	49
12.15managePixmap Class Reference	49
12.16PeriodicDialog Class Reference	50
12.17PeriodicElementSetupForm Class Reference	50
12.18PeriodicSetupDialog Class Reference	50
12.19QAnalogProgressBar Class Reference	50
12.19.1 Detailed Description	51
12.20QAnalogProgressBarPluginManager Class Reference	52
12.20.1 Detailed Description	52
12.21QBitStatus Class Reference	52
12.21.1 Detailed Description	54
12.22QBitStatusPluginManager Class Reference	54
12.22.1 Detailed Description	55
12.23QCaAlarmInfo Class Reference	55
12.24QCaAnalogProgressBar Class Reference	56
12.24.1 Property Documentation	57
12.24.1.1 variable	57
12.25QCaAnalogProgressBarPluginManager Class Reference	58
12.25.1 Detailed Description	58
12.26QCaBitStatus Class Reference	58
12.26.1 Detailed Description	60
12.27QCaBitStatusPluginManager Class Reference	60
12.27.1 Detailed Description	61
12.28QCaComboBox Class Reference	61
12.28.1 Detailed Description	62
12.29QCaComboBoxPlugin Class Reference	63

12.29.1 Detailed Description	64
12.30QCaComboBoxPluginManager Class Reference	64
12.30.1 Detailed Description	65
12.31QCaConnectionInfo Class Reference	65
12.32QCaDateTime Class Reference	66
12.33QCaDesignerPlugin Class Reference	66
12.33.1 Detailed Description	66
12.34QCaEventFilter Class Reference	67
12.34.1 Detailed Description	67
12.35QCaEventItem Class Reference	68
12.35.1 Detailed Description	69
12.36QCaEventUpdate Class Reference	69
12.36.1 Detailed Description	69
12.37QCaFloating Class Reference	70
12.37.1 Detailed Description	71
12.38QCaFloatingFormatting Class Reference	71
12.38.1 Detailed Description	72
12.39QCaInstalledFiltersListItem Class Reference	72
12.40QCaInteger Class Reference	73
12.40.1 Detailed Description	74
12.41QCaIntegerFormatting Class Reference	74
12.41.1 Detailed Description	74
12.42QCaLabel Class Reference	75
12.42.1 Detailed Description	77
12.43QCaLabelPlugin Class Reference	77
12.43.1 Detailed Description	80
12.43.2 Property Documentation	80
12.43.2.1 variable	80
12.44QCaLabelPluginManager Class Reference	80
12.44.1 Detailed Description	81
12.45QCaLineEdit Class Reference	81
12.45.1 Detailed Description	83
12.46QCaLineEditPlugin Class Reference	84
12.46.1 Detailed Description	85
12.46.2 Property Documentation	86
12.46.2.1 variable	86
12.47QCaLineEditPluginManager Class Reference	86
12.47.1 Detailed Description	86
12.48QCaObject Class Reference	87
12.48.1 Detailed Description	87
12.49qcaobject::QCaObject Class Reference	88
12.50QCaPeriodic Class Reference	89
12.51QCaPeriodicComponentData Class Reference	91
12.52QCaPeriodicPlugin Class Reference	92
12.52.1 Property Documentation	94
12.52.1.1 writeButtonVariable1	94
12.53QCaPeriodicPluginManager Class Reference	94
12.53.1 Detailed Description	94
12.54QCaPeriodicPluginTaskMenu Class Reference	94
12.55QCaPeriodicPluginTaskMenuFactory Class Reference	95

12.56QCaPlot Class Reference	95
12.56.1 Detailed Description	98
12.57QCaPlotPlugin Class Reference	98
12.57.1 Detailed Description	100
12.58QCaPlotPluginManager Class Reference	101
12.58.1 Detailed Description	101
12.59QCaPushButton Class Reference	102
12.60QCaPushButtonPlugin Class Reference	104
12.60.1 Property Documentation	107
12.60.1.1 variable	107
12.61QCaPushButtonPluginManager Class Reference	107
12.61.1 Detailed Description	108
12.62QCaShape Class Reference	108
12.62.1 Detailed Description	111
12.63QCaShapePlugin Class Reference	111
12.63.1 Detailed Description	116
12.63.2 Property Documentation	116
12.63.2.1 variable1	116
12.64QCaShapePluginManager Class Reference	117
12.64.1 Detailed Description	117
12.65QCaSlider Class Reference	118
12.65.1 Detailed Description	119
12.66QCaSliderPlugin Class Reference	119
12.66.1 Detailed Description	120
12.66.2 Property Documentation	121
12.66.2.1 variable	121
12.67QCaSliderPluginManager Class Reference	121
12.67.1 Detailed Description	122
12.68QCaSpinBox Class Reference	122
12.68.1 Detailed Description	123
12.69QCaSpinBoxPlugin Class Reference	124
12.69.1 Detailed Description	125
12.69.2 Property Documentation	125
12.69.2.1 variable	125
12.70QCaSpinBoxPluginManager Class Reference	125
12.70.1 Detailed Description	126
12.71QCaStateMachine Class Reference	126
12.71.1 Detailed Description	127
12.72qcastatemachine::QCaStateMachine Class Reference	127
12.73QCaString Class Reference	128
12.73.1 Detailed Description	128
12.74QCaStringFormatting Class Reference	129
12.74.1 Detailed Description	130
12.75QCaToolTip Class Reference	131
12.75.1 Detailed Description	131
12.76QCaVariableNamePropertyManager Class Reference	132
12.76.1 Detailed Description	132
12.77QCaWidget Class Reference	133
12.77.1 Detailed Description	135
12.78QCaWidgets Class Reference	136

12.79QEpicsPV Class Reference	136
12.79.1 Constructor & Destructor Documentation	137
12.79.1.1 QEpicsPV	137
12.79.1.2 QEpicsPV	137
12.79.2 Member Function Documentation	138
12.79.2.1 connectionChanged	138
12.79.2.2 get	138
12.79.2.3 get	138
12.79.2.4 getEnum	138
12.79.2.5 getReady	139
12.79.2.6 getUpdated	139
12.79.2.7 isConnected	139
12.79.2.8 needUpdated	139
12.79.2.9 pv	140
12.79.2.10set	140
12.79.2.11set	140
12.79.2.12setPV	141
12.79.2.13valueChanged	141
12.79.2.14valueUpdated	141
12.80QSubstitutedLabel Class Reference	141
12.80.1 Detailed Description	142
12.81QSubstitutedLabelPlugin Class Reference	142
12.81.1 Property Documentation	143
12.81.1.1 initialText	143
12.81.1.2 textSubstitutions	143
12.82QSubstitutedLabelPluginManager Class Reference	143
12.82.1 Detailed Description	144
12.83qcastatemachine::ReadQCaStateMachine Class Reference	144
12.84StateMachineTemplate Class Reference	144
12.85qcastatemachine::SubscriptionQCaStateMachine Class Reference	145
12.86trace Class Reference	145
12.87userInfoStruct Class Reference	146
12.88QCaPeriodic::userInfoStructArray Struct Reference	146
12.89UserMessage Class Reference	146
12.89.1 Detailed Description	147
12.90VariableNameManager Class Reference	147
12.90.1 Detailed Description	149
12.91WidgetRef Class Reference	149
12.92qcastatemachine::WriteQCaStateMachine Class Reference	150

Chapter 1

QCa - EPICS aware Qt Widgets

- QCa is a layered software framework for accessing EPICS data using Channel Access on a range of platforms.
- The QCa framework provides object oriented C++ access to control systems using EPICS (Experimental Physics and Industrial Control System). It is based on Qt, a widely used cross-platform application development framework.
- GUI or console based applications can be written that use QCa at several levels. QCa includes Qt plugin libraries, EPICS aware widgets, data formatting classes, and classes for accessing raw EPICS data in a Qt friendly way.
- QCa also includes an application - ASgui - for displaying forms produced by the Qt development tool 'Designer'. Using this application a complete EPICS GUI system can be generated without writing any code. A GUI system produced in this way can interact with existing EPICS display tools such as EDM.
- QCa handles much of the complexities of Channel Access including initiating and managing a channel. Applications using QCa can interact with Channel Access using Qt based classes and data types. Channel Access updates are delivered using Qt's signals and slots mechanism.

1.1 Documentation

Support documents can be found in the [documentation](#) section of the epicsqt sourceforge project. The framework download (available on the [epicsqt sourceforge homepage](#)) also includes this documentation as well as full Doxygen generated documentation of all the epicsqt classes and widgets.

1.2 License

epicsqt is distributed under the terms of the [GNU General Public License](#).

1.3 Platforms

epicsqt might be usable in all environments where you find [Qt](#). It is compatible with Qt >= 4.4.

1.4 Screenshots

- [ASgui screen shots](#)
- [other applications using epicsqt widgets](#)
- [Qt Designer](#)
- [Qt Creator](#)

Screenshots are only available in the HTML docs.

1.5 Downloads

Stable releases and development snapshots are available at the [epicsqt project page](#). For getting a development snapshot from the SVN repository:

```
svn svn co https://epicsqt.svn.sourceforge.net/svnroot/epicsqt epicsqt
```

1.6 Installation

Read [QCa_GettingStarted.pdf](#) in the documentation for setting up an environment for building or using the epicsqt framework.

To build the framework, open epicsqt.pro in QtCreator, ensure shadow build is turned off, and hit build.

Symbolic links to the resultant library libQCaPlugin.so will need to set up according to how it is to be used - see [QCa_GettingStarted.pdf](#) for details.

Any Qt specific queries? start at [Qt developer documentation](#)

1.7 Support

Visit the sourceforge epicsqt [support page](#) for assistance.

1.8 Related Projects

[Qwt](#), The core of a Channel Access aware plotting widget.

1.9 Credits:**Authors:**

Andrew Rhyder, Anthony Owen, Glenn Jackson

Project admin:

Andrew Rhyder <andrew.rhyder@synchrotron.org.au>

Chapter 2

GNU General Public License

The EPICS QT Framework is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

The EPICS QT Framework is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.

See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with the EPICS QT Framework.

If not, see "<http://www.gnu.org/licenses/>

Chapter 3

INSTALL

Chapter 4

ASgui screen shots

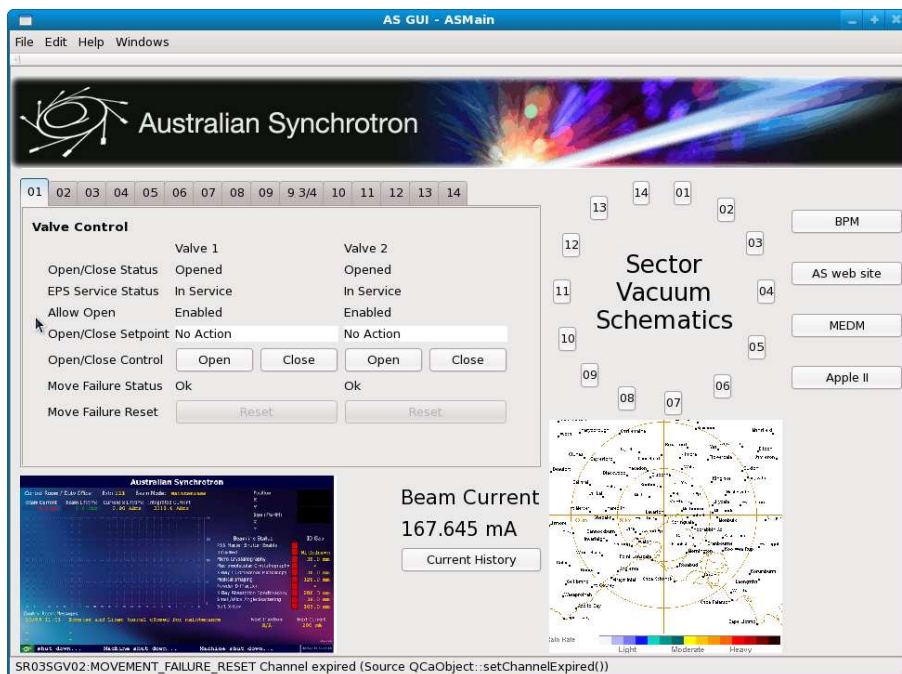


Figure 4.1: Australian Synchrotron mock up

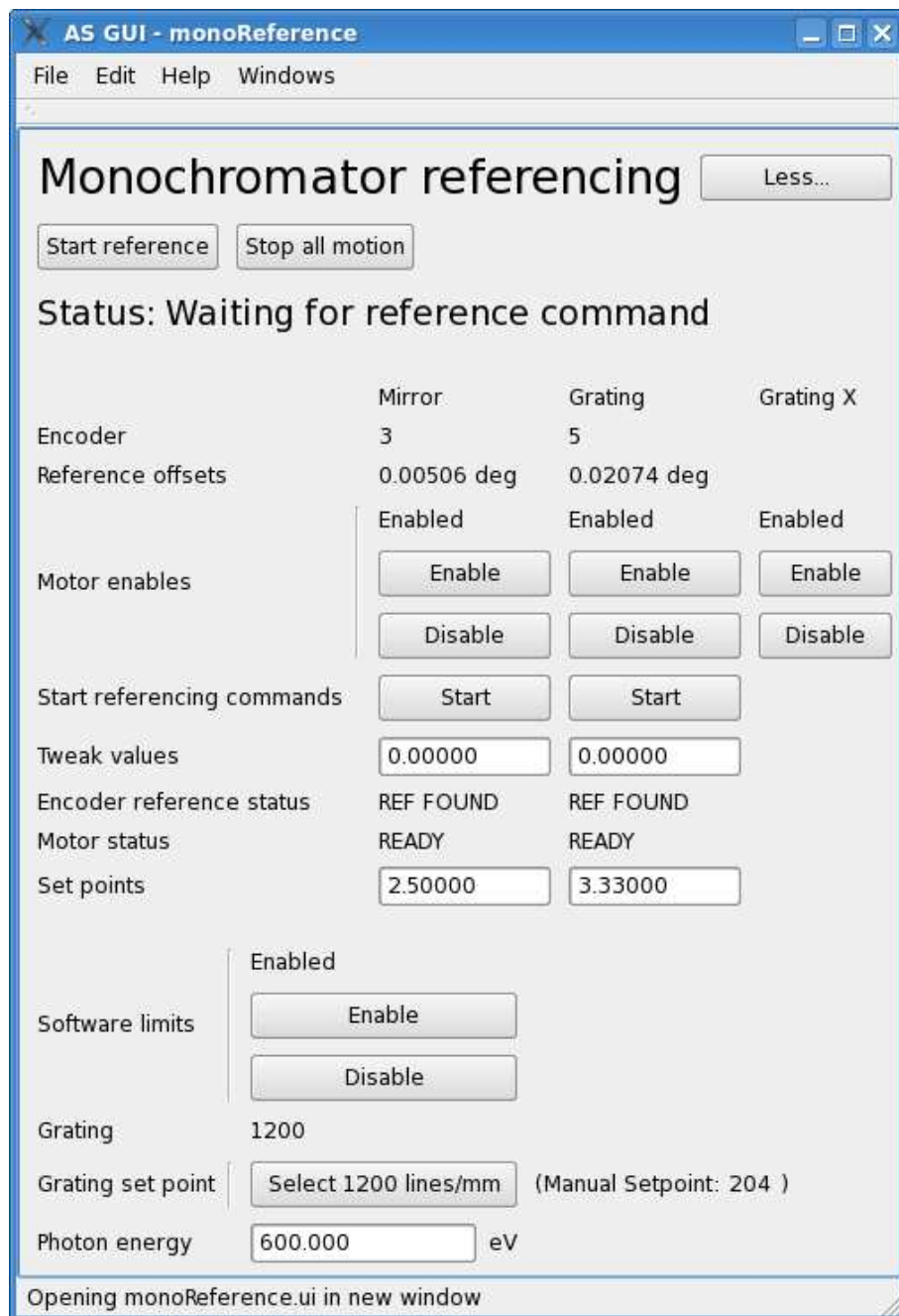


Figure 4.2: Monochromator referencing

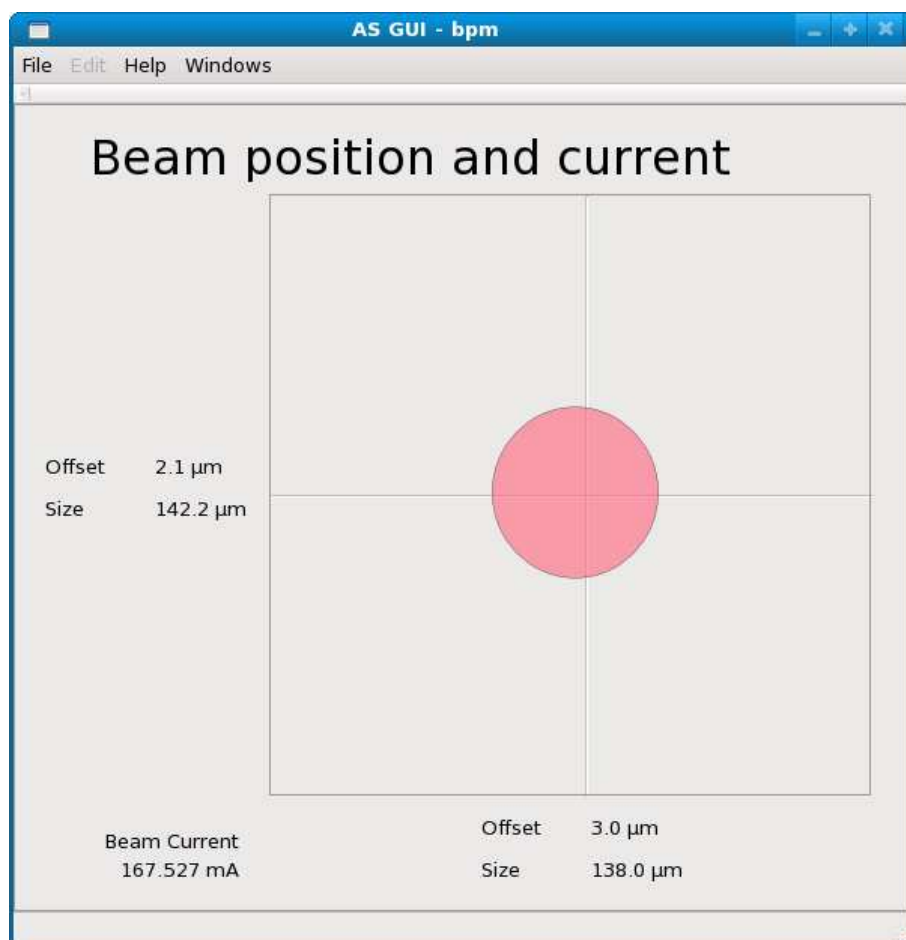


Figure 4.3: Beam position monitor

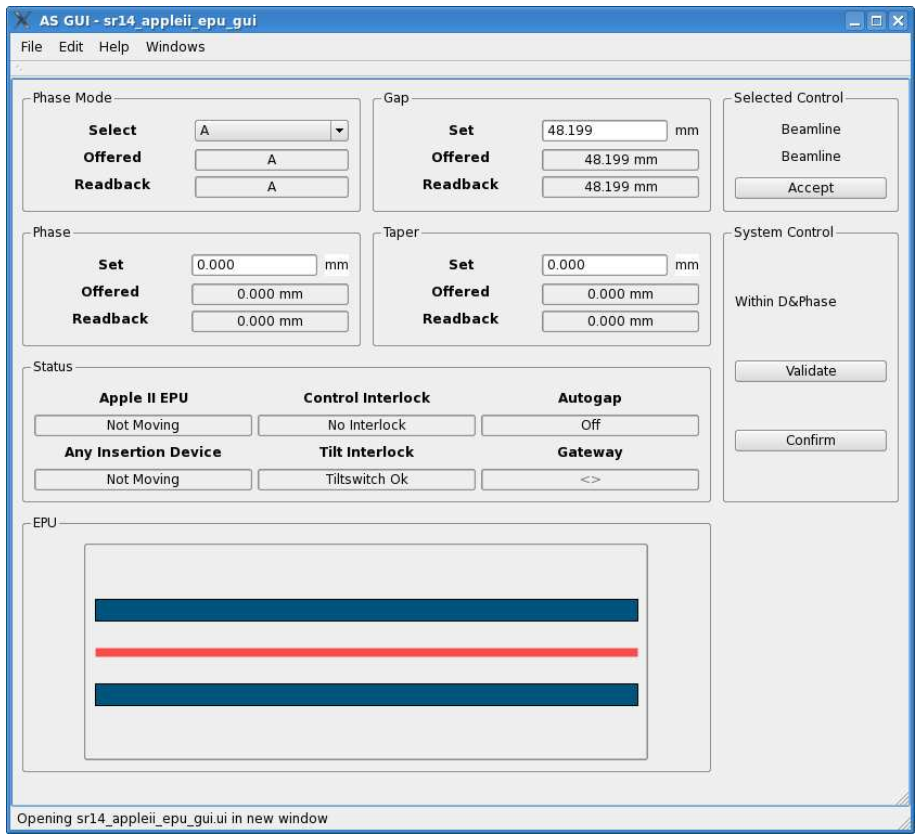


Figure 4.4: Insertion device

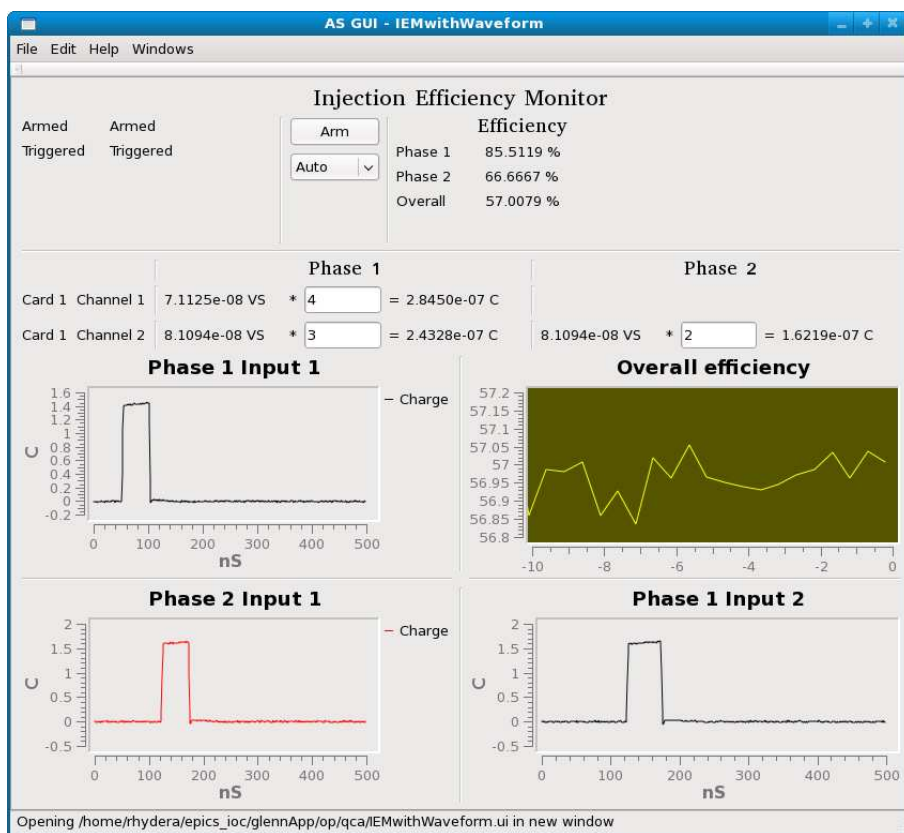


Figure 4.5: Injection efficiency monitor

Chapter 5

other applications using epicsqt widgets

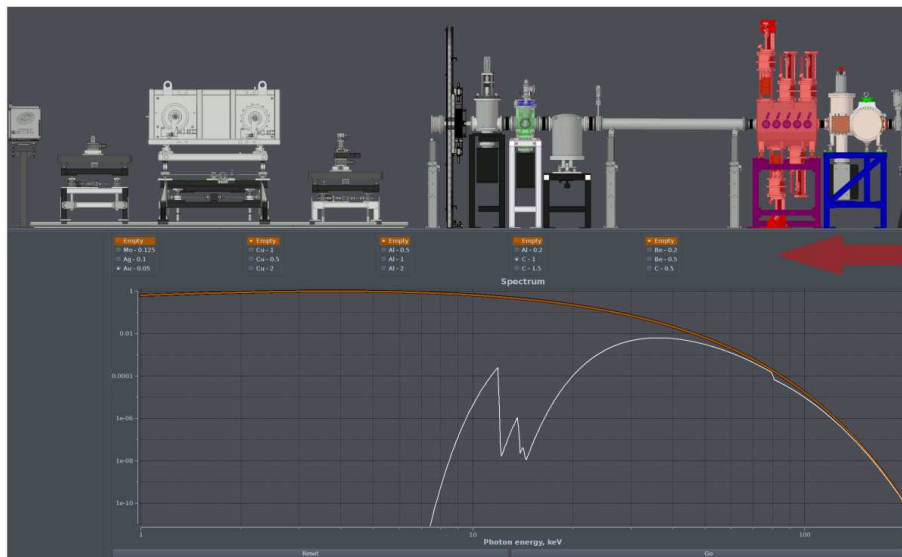


Figure 5.1: Medical Imaging beamline

2B SampleTable Z <@SR08ID01OPI01>

PV name: SR08ID01:MTR32B View mode: Macro

Description: 2B SampleTable Z

Precision: 5 Units: mm

Message: Connection established. clean

User: 6mm Move absolutely Raw: -80932

JOG< UNDO >JOG

LIMIT< Move relatively >LIMIT

< 1mm >

step/10 step/2 step*2 step*10

User: 6mm = Hi limit

Resolution: 0.0001mm/step * 42271.2068mm

Raw: -80932 + Lo Limit

Offset: -2.0932mm -42275.3932mm

Speed Acceleration

Maximum: 1.2mm/s

Normal: 0.8mm/s 1s

Backlash: 0mm/s 1s

log: 1mm/s 10s

Backlash: 0mm

Figure 5.2: Motor controller

MotorMx <@SR08ID01OPI01>

- ▲ ▼	DEI Theta Mono	109.5mm	<	0.1	>	UNDO
- ▲ ▼	DEI Mono Z	-0.3mm	<	0.1	>	UNDO
- ▲ ▼	2B Sample Table Y	0mm	<	1	>	UNDO
- ▲ ▼	2B SampleTable Z	6mm	<	1	>	UNDO
- ▲ ▼	2B Detector Table Z	42mm	<	5	>	UNDO
- ▲ ▼	2B Sample Rotate	-1deg	<	1	>	UNDO
- ▲ ▼	2B Detector Table Y	13.9025mm	<	1	>	UNDO
- ▲ ▼	SETUP	0	<	relative	>	STOP
- ▲ ▼	SLW01:LEFT	9.99975mm	<	3456	>	UNDO

Add motor

Figure 5.3: Motor controller

Chapter 6

Qt Designer

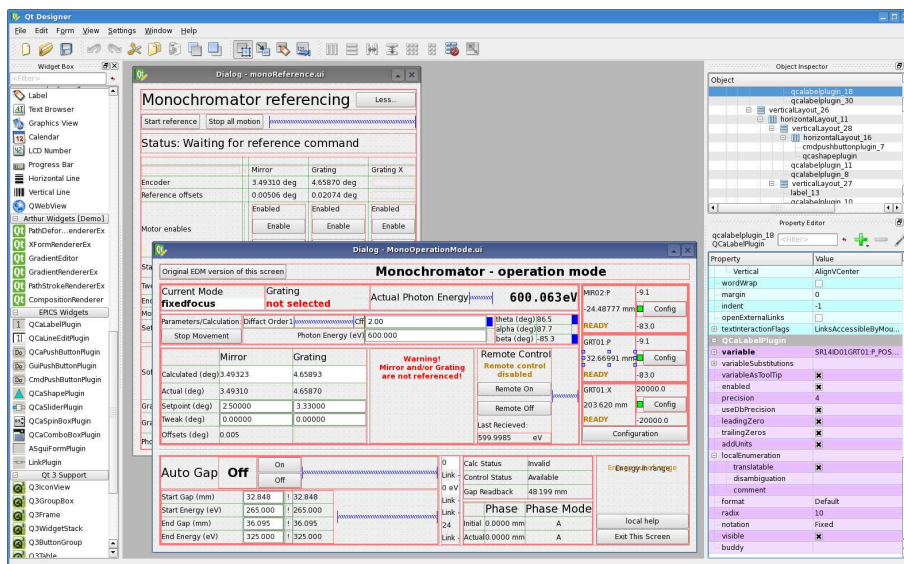


Figure 6.1: Editing multiple GUIs

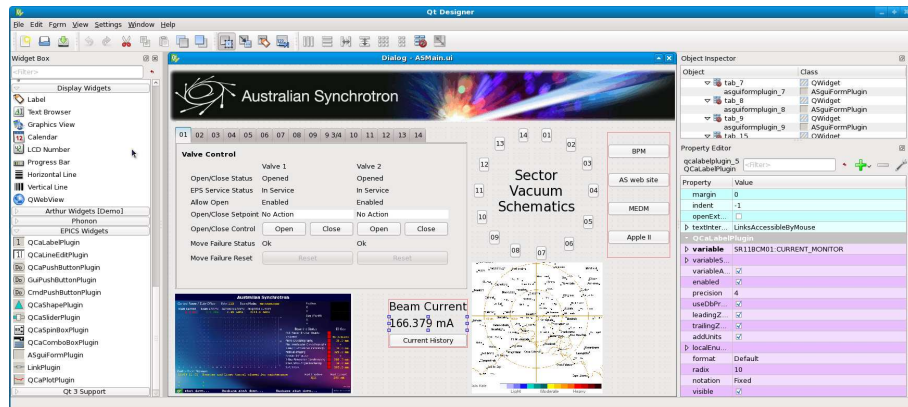


Figure 6.2: Editing a GUI

Chapter 7

Qt Creator

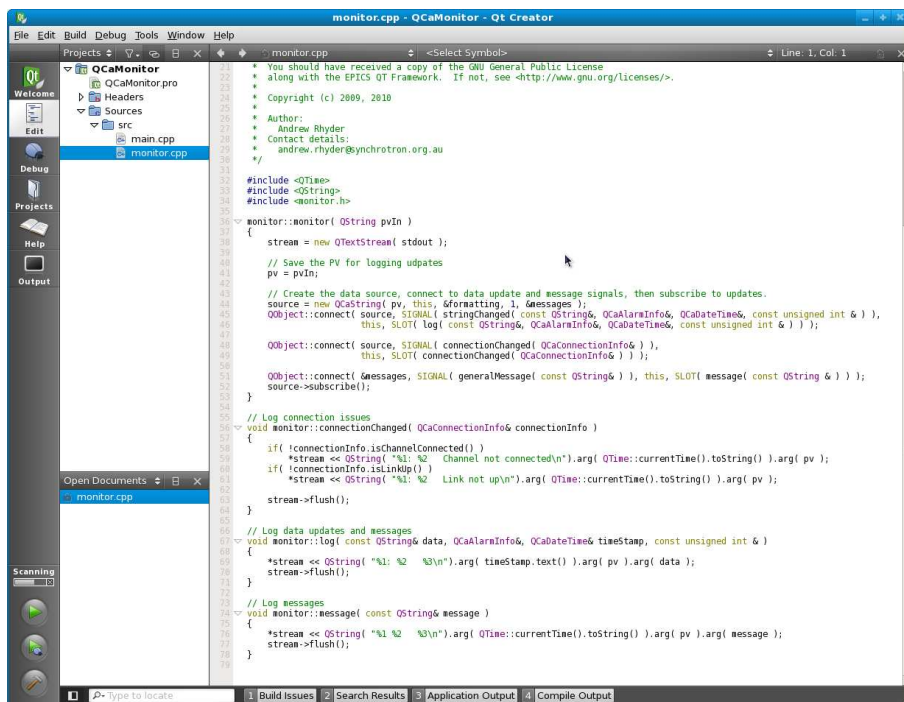


Figure 7.1: Application using epicsqt data source classes

Chapter 8

Namespace Index

8.1 Namespace List

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

[qcaobject](#) 31

Chapter 9

Class Index

9.1 Class Hierarchy

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

ASguiForm	33
ASguiFormPlugin	35
ASguiFormPluginManager	36
CmdPushButtonPluginManager	38
ContainerProfile	39
QCaWidget	133
QCaAnalogProgressBar	56
QCaBitStatus	58
QCaComboBox	61
QCaComboBoxPlugin	63
QCaLabel	75
QCaLabelPlugin	77
QCaLineEdit	81
QCaLineEditPlugin	84
QCaPeriodic	89
QCaPeriodicPlugin	92
QCaPlot	95
QCaPlotPlugin	98
QCaPushButton	102
CmdPushButtonPlugin	37
GuiPushButtonPlugin	42
QCaPushButtonPlugin	104
QCaShape	108
QCaShapePlugin	111
QCaSlider	118
QCaSliderPlugin	119
QCaSpinBox	122
QCaSpinBoxPlugin	124

QSubstitutedLabel	141
QSubstitutedLabelPlugin	142
QCaPeriodic::elementInfoStruct	41
GuiPushButtonPluginManager	43
Link	44
LinkPlugin	46
LinkPluginManager	48
localEnumerationItem	49
managePixmaps	49
QCaLabel	75
QCaPushButton	102
PeriodicDialog	50
PeriodicElementSetupForm	50
PeriodicSetupDialog	50
QAnalogProgressBar	50
QCaAnalogProgressBar	56
QAnalogProgressBarPluginManager	52
QBitStatus	52
QCaBitStatus	58
QBitStatusPluginManager	54
QCaAlarmInfo	55
QCaAnalogProgressBarPluginManager	58
QCaBitStatusPluginManager	60
QCaComboBoxPluginManager	64
QCaConnectionInfo	65
QCaDateTime	66
QCaDesignerPlugin	66
QCaEventFilter	67
QCaEventItem	68
QCaEventUpdate	69
QCaFloatingFormatting	71
QCaInstalledFiltersListItem	72
QCaIntegerFormatting	74
QCaLabelPluginManager	80
QCaLineEditPluginManager	86
QCaObject	87
qcaobject::QCaObject	88
QCaFloating	70
QCaInteger	73
QCaString	128
QCaPeriodicComponentData	91
QCaPeriodicPluginManager	94
QCaPeriodicPluginTaskMenu	94
QCaPeriodicPluginTaskMenuFactory	95
QCaPlotPluginManager	101
QCaPushButtonPluginManager	107
QCaShapePluginManager	117
QCaSliderPluginManager	121

QCaSpinBoxPluginManager	125
QCaStateMachine	126
QCaStringFormatting	129
QCaToolTip	131
QCaWidget	133
QCaVariableNamePropertyManager	132
QCaWidgets	136
QEpicsPV	136
QSubstitutedLabelPluginManager	143
StateMachineTemplate	144
qcastatemachine::QCaStateMachine	127
qcastatemachine::ConnectionQCaStateMachine	39
qcastatemachine::ReadQCaStateMachine	144
qcastatemachine::SubscriptionQCaStateMachine	145
qcastatemachine::WriteQCaStateMachine	150
trace	145
userInfoStruct	146
QCaPeriodic::userInfoStructArray	146
UserMessage	146
VariableNameManager	147
QCaWidget	133
WidgetRef	149

Chapter 10

Class Index

10.1 Class List

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

ASguiForm (A base form GUI)	33
ASguiFormPlugin (Wrapper around ASguiForm to make it a Qt Designer Plugin)	35
ASguiFormPluginManager (ASgui form Plugin for Qt Designer)	36
CmdPushButtonPlugin	37
CmdPushButtonPluginManager (Command Push Button Widget Plugin Manager for designer)	38
qcastatemachine::ConnectionQCaStateMachine	39
ContainerProfile (Defines attributes of the containing window (form, dialog, etc) within which QCa widgets are being created)	39
QCaPeriodic::elementInfoStruct	41
GuiPushButtonPlugin (Launch GUI Push Button Widget Plugin)	42
GuiPushButtonPluginManager (Launch GUI Push Button Widget Plugin Manager for designer)	43
Link (Manage updating tool tip with variable name, alarm state and connected state)	44
LinkPlugin (Link Widget Plugin for designer)	46
LinkPluginManager (Link Widget Plugin Manager for designer)	48
localEnumerationItem	49
managePixmap	49
PeriodicDialog	50
PeriodicElementSetupForm	50
PeriodicSetupDialog	50
QAnalogProgressBar	50
QAnalogProgressBarPluginManager	52
QBitStatus (Bit Status Widget)	52
QBitStatusPluginManager (CA ProgressBar Widget Plugin Manager for designer)	54
QCaAlarmInfo	55

QCaAnalogProgressBar	56
QCaAnalogProgressBarPluginManager	58
QCaBitStatus (CA Bit Status Widget)	58
QCaBitStatusPluginManager (CA ProgressBar Widget Plugin Manager for designer)	60
QCaComboBox (CA Combobox Widget)	61
QCaComboBoxPlugin (CA Combobox Widget Plugin)	63
QCaComboBoxPluginManager (CA Combobox Widget Plugin Manager for designer)	64
QCaConnectionInfo	65
QCaDateTime	66
QCaDesignerPlugin (Collection of CA plugins for QT Designer)	66
QCaEventFilter (Manage CA alarm and severity information)	67
QCaEventItem (Simple container for a list QCaEventUpdate instances)	68
QCaEventUpdate (Packages CA callback data as a Qt event)	69
QCaFloating (Floating specific wrapper for QCaObject)	70
QCaFloatingFormatting (Provides textual formatting for QCaFloating data)	71
QCaInstalledFiltersListItem	72
QCaInteger (Integer specific wrapper for QCaObject)	73
QCaIntegerFormatting (Provides textual formatting for QCaInteger data)	74
QCaLabel (Manage updating tool tip with variable name, alarm state and connected state)	75
QCaLabelPlugin (CA Label Widget Plugin for designer)	77
QCaLabelPluginManager (CA Label Widget Plugin Manager for designer)	80
QCaLineEdit (CA Line Edit Widget)	81
QCaLineEditPlugin (CA Line Edit Widget Plugin)	84
QCaLineEditPluginManager (CA Line Edit Widget Plugin Manager for designer)	86
QCaObject (Provides channel access to QT)	87
qcaobject::QCaObject	88
QCaPeriodic	89
QCaPeriodicComponentData	91
QCaPeriodicPlugin	92
QCaPeriodicPluginManager	94
QCaPeriodicPluginTaskMenu	94
QCaPeriodicPluginTaskMenuFactory	95
QCaPlot (Manage updating tool tip with variable name, alarm state and connected state)	95
QCaPlotPlugin (CA Plot Widget Plugin for designer)	98
QCaPlotPluginManager (CA Plot Widget Plugin Manager for designer)	101
QCaPushButton	102
QCaPushButtonPlugin	104
QCaPushButtonPluginManager (CA Push Button Widget Plugin Manager for designer)	107
QCaShape (CA Shape Widget)	108
QCaShapePlugin (CA Shape Widget Plugin)	111
QCaShapePluginManager (CA Shape Widget Plugin Manager for designer)	117
QCaSlider (CA Slider Widget)	118
QCaSliderPlugin (CA Slider Widget Plugin)	119
QCaSliderPluginManager (CA Slider Widget Plugin Manager for designer)	121

QCaSpinBox (CA Spinbox Widget)	122
QCaSpinBoxPlugin (CA Spinbox Widget Plugin)	124
QCaSpinBoxPluginManager (CA Spinbox Widget Plugin Manager for de- signer)	125
QCaStateMachine (Statemachine architecture)	126
qcastatemachine::QCaStateMachine	127
QCaString (String wrapper for QCaObject variant data)	128
QCaStringFormatting (Formats the string for QCaString data)	129
QCaToolTip (Manage updating tool tip with variable name, alarm state and connected state)	131
QCaVariableNamePropertyManager (Handles variable name sets)	132
QCaWidget (Template for Qt-CA aware widgets)	133
QCaWidgets	136
QEpicsPV	136
QSubstitutedLabel	141
QSubstitutedLabelPlugin	142
QSubstitutedLabelPluginManager	143
qcastatemachine::ReadQCaStateMachine	144
StateMachineTemplate	144
qcastatemachine::SubscriptionQCaStateMachine	145
trace	145
userInfoStruct	146
QCaPeriodic::userInfoStructArray	146
UserMessage (User message manager)	146
VariableNameManager (Variable name management)	147
WidgetRef	149
qcastatemachine::WriteQCaStateMachine	150

Chapter 11

Namespace Documentation

11.1 qcaobject Namespace Reference

Classes

- class [QCaObject](#)

11.1.1 Detailed Description

Version

Revision:

#4

Date

DateTime:

2010/06/23 07:49:40

Author

Last checked in by:

Author:

rhydera

Copyright (c) 2009 Australian Synchrotron

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public Licence as published by the Free Software Foundation; either version 2.1 of the Licence, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public Licence for more details.

You should have received a copy of the GNU Lesser General Public Licence along with this library; if not, write to the Free Software Foundation, Inc., 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA.

Contact details: andrew.rhyder@synchrotron.org.au 800 Blackburn Road, Clayton, Victoria 3168, Australia.

Chapter 12

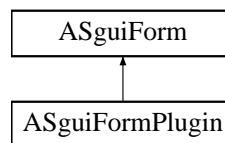
Class Documentation

12.1 ASguiForm Class Reference

A base form GUI.

```
#include <ASguiForm.h>
```

Inheritance diagram for ASguiForm:



Public Types

- enum **creationOptions** { **CREATION_OPTION_OPEN**, **CREATION_OPTION_NEW_TAB**, **CREATION_OPTION_NEW_WINDOW** }

Public Slots

- void **requestEnabled** (const bool &state)
- void **readUiFile** ()
- void **onGeneralMessage** (QString message)

Public Member Functions

- **ASguiForm** (QWidget *parent=0)
- **ASguiForm** (const QString &uifileNameIn, QWidget *parent=0)
- void **commonInit** (const bool alertIfUINoFoundIn)

- `QString getASGuiTitle ()`
- `QString getGuiFileName ()`
- `void setVariableNameAndSubstitutions (QString variableNameIn, QString variableNameSubstitutionsIn, unsigned int variableIndex)`
- `void setUiFileName (QString uiFile)`
- `QString getUiFileName ()`
- `void setHandleGuiLaunchRequests (bool handleGuiLaunchRequests)`
- `bool getHandleGuiLaunchRequests ()`

Protected Member Functions

- `void setVariableNameSubstitutions (QString variableNameSubstitutionsIn)`

Protected Attributes

- `QString uiFileName`
- `bool handleGuiLaunchRequests`

12.1.1 Detailed Description

A base form GUI.

Version

Revision:

#10

Date

DateTime:

2010/09/06 13:16:04

Author

andrew.rhyder

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

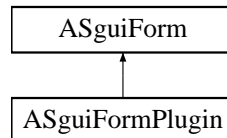
- `epicsqt/ca_framework/widgets/include/ASguiForm.h`

12.2 ASguiFormPlugin Class Reference

Wrapper around [ASguiForm](#) to make it a Qt Designer Plugin.

```
#include <ASguiFormPlugin.h>
```

Inheritance diagram for ASguiFormPlugin:



Public Member Functions

- [ASguiFormPlugin](#) (QWidget *parent=0)
Constructors.
- void **setVariableNameSubstitutionsProperty** (QString variableNameSubstitutions)
- QString **getVariableNameSubstitutionsProperty** ()

Properties

- QString [uiFile](#)
- QString **variableSubstitutions**
- bool **handleGuiLaunchRequests**

12.2.1 Detailed Description

Wrapper around [ASguiForm](#) to make it a Qt Designer Plugin. Wrapper around [QCaLabel](#) to allow [QCaLabel](#) to be used within Designer.

Version

Revision:

#5

Date

DateTime:

2010/09/06 11:58:56

Author

andrew.rhyder

12.2.2 Property Documentation**12.2.2.1 QString ASguiFormPlugin::uiFile** [read, write]

Note, a property macro in the form 'Q_PROPERTY(QString uiFileName READ ...' doesn't work. A property name ending with 'Name' results in some sort of string a variable being displayed, but will only accept alphanumeric and won't generate call-backs on change.

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

- epicsqt/ca_framework/plugins/include/ASguiFormPlugin.h

12.3 ASguiFormPluginManager Class Reference

ASgui form Plugin for Qt Designer.

```
#include <ASguiFormPluginManager.h>
```

Public Member Functions

- **ASguiFormPluginManager** (QObject *parent=0)
- bool **isContainer** () const
- bool **isInitialized** () const
- QIcon **icon** () const
- QString **group** () const
- QString **includeFile** () const
- QString **name** () const
- QString **toolTip** () const
- QString **whatsThis** () const
- QWidget * **createWidget** (QWidget *parent)
- void **initialize** (QDesignerFormEditorInterface *core)

12.3.1 Detailed Description

ASgui form Plugin for Qt Designer.

Version**Revision:**

#4

Date**DateTime:**

2010/06/23 07:49:40

Author

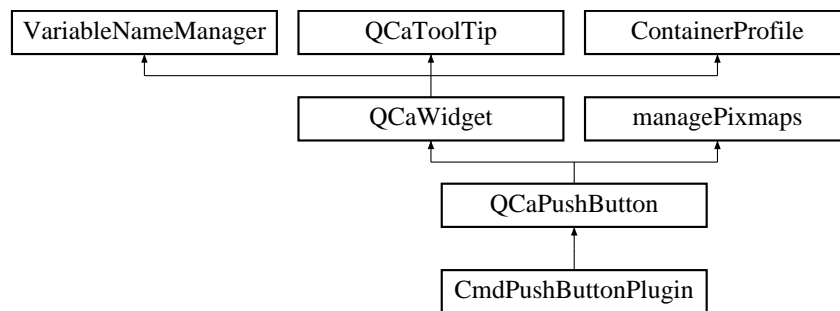
<andrew.rhyder>

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

- `epicsqt/ca_framework/plugins/include/ASguiFormPluginManager.h`

12.4 CmdPushButtonPlugin Class Reference

Inheritance diagram for CmdPushButtonPlugin:

**Public Member Functions**

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

Constructors.

Properties

- QString [program](#)
Text to write on a button click.
- QStringList [arguments](#)
Program to run.
- QString [variableSubstitutions](#)

12.4.1 Property Documentation

12.4.1.1 QString CmdPushButtonPlugin::variableSubstitutions [read, write]

Variable substitutions Example: SECTOR=01 will result in any occurrence of \$SECTOR in variable name being replaced with 01. Note, unlike most objects with a variable name substitutions, changes to this property do not take immediate effect, so the normal code which waits for the user to finish typing, then propagates the substitutions is not required. the substitutions are simply used next time the button is pushed.

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

- epicsqt/ca_framework/plugins/include/CmdPushButtonPlugin.h

12.5 CmdPushButtonPluginManager Class Reference

Command Push Button Widget Plugin Manager for designer.

```
#include <CmdPushButtonPluginManager.h>
```

Public Member Functions

- **CmdPushButtonPluginManager** (QObject *parent=0)
- bool **isContainer** () const
- bool **isInitialized** () const
- QIcon **icon** () const
- QString **group** () const
- QString **includeFile** () const
- QString **name** () const
- QString **toolTip** () const
- QString **whatsThis** () const
- QWidget * **createWidget** (QWidget *parent)
- void **initialize** (QDesignerFormEditorInterface *core)

12.5.1 Detailed Description

Command Push Button Widget Plugin Manager for designer. ???

Version

Revision:

#4

Date

DateTime:

2010/06/23 07:49:40

Author

Last checked in by:

Author:

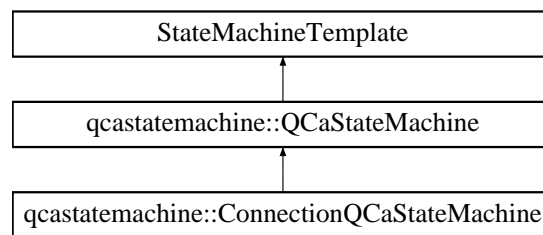
rhydera

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

- epicsqt/ca_framework/plugins/include/CmdPushButtonPluginManager.h

12.6 qcastatemachine::ConnectionQCaStateMachine Class Reference

Inheritance diagram for qcastatemachine::ConnectionQCaStateMachine:

**Public Member Functions**

- **ConnectionQCaStateMachine** (void *parent)
- bool **process** (int requestedState)

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

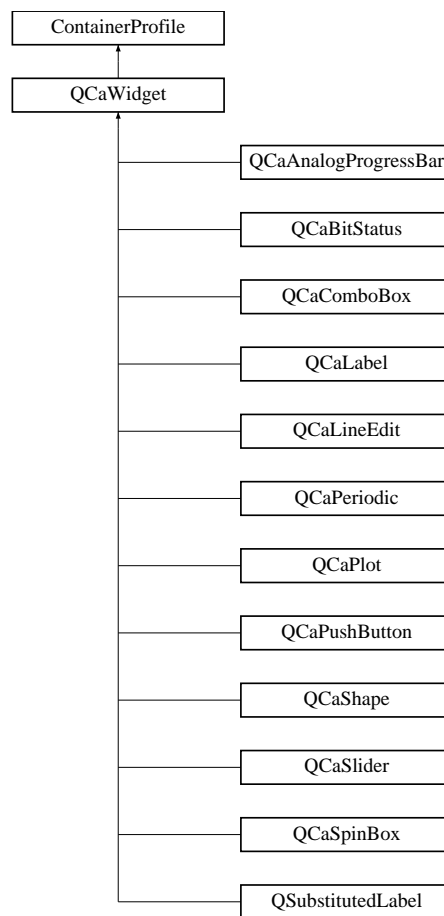
- epicsqt/ca_framework/data/include/QCaStateMachine.h

12.7 ContainerProfile Class Reference

Defines attributes of the containing window (form, dialog, etc) within which QCa widgets are being created.

```
#include <ContainerProfile.h>
```

Inheritance diagram for ContainerProfile:



Public Member Functions

- void **init** ()
- void **setupProfile** (QObject *statusMessageConsumerIn, QObject *errorMessageConsumerIn, QObject *warningMessageConsumerIn, QObject *guiLaunchConsumerIn, QString pathIn, QString macroSubstitutionsIn, bool interactiveIn)
- QObject * **replaceGuiLaunchConsumer** (QObject *newGuiLaunchConsumerIn)

- void **addMacroSubstitutions** (QString macroSubstitutionsIn)
- void **removeMacroSubstitutions** ()
- QObject * **getStatusMessageConsumer** ()
- QObject * **getWarningMessageConsumer** ()
- QObject * **getErrorMessageConsumer** ()
- QObject * **getGuiLaunchConsumer** ()
- QString **getPath** ()
- QString **getMacroSubstitutions** ()
- bool **isProfileDefined** ()
- bool **isInteractive** ()

- void **addContainedWidget** (QCaWidget *containedWidget)
- QCaWidget * **getNextContainedWidget** ()
- void **releaseProfile** ()

12.7.1 Detailed Description

Defines attributes of the containing window (form, dialog, etc) within which QCa widgets are being created.

Version

Revision:

#4

Date

DateTime:

2010/06/23 07:49:40

Author

andrew.rhyder

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

- epicsqt/ca_framework/plugins/include/ContainerProfile.h

12.8 QCaPeriodic::elementInfoStruct Struct Reference

Public Attributes

- unsigned int **number**
- double **atomicWeight**
- QString **name**
- QString **symbol**
- double **meltingPoint**
- double **boilingPoint**
- double **density**
- unsigned int **group**
- double **ionizationEnergy**
- unsigned int **tableRow**
- unsigned int **tableCol**

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

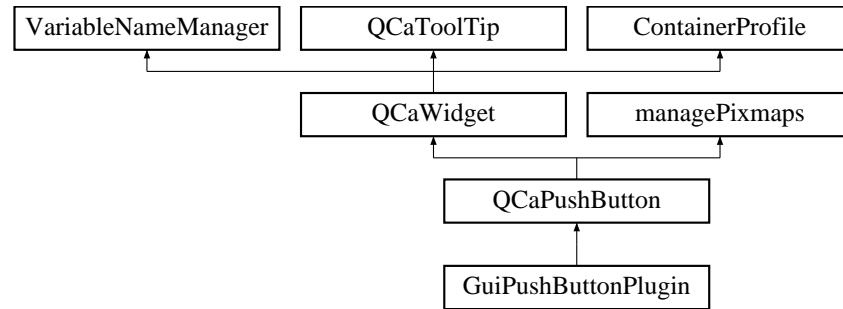
- epicsqt/ca_framework/widgets/include/QCaPeriodic.h

12.9 GuiPushButtonPlugin Class Reference

Launch GUI Push Button Widget Plugin.

```
#include <GuiPushButtonPlugin.h>
```

Inheritance diagram for GuiPushButtonPlugin:



Public Types

- enum **CreationOptionNames** { **Open** = ASguiForm::CREATION_OPTION_OPEN, **NewTab** = ASguiForm::CREATION_OPTION_NEW_TAB, **NewWindow** = ASguiForm::CREATION_OPTION_NEW_WINDOW }

Public Member Functions

- [GuiPushButtonPlugin](#) (QWidget *parent=0)
Constructors.
- void **setCreationOptionProperty** (CreationOptionNames creationOptionIn)
- CreationOptionNames **getCreationOptionProperty** ()

Properties

- QString **guiFile**
- CreationOptionNames [creationOption](#)
GUI file name to launch.
- QString **variableSubstitutions**

12.9.1 Detailed Description

Launch GUI Push Button Widget Plugin.

Version**Revision:**

#6

Date**DateTime:**

2010/09/06 11:58:56

Author

Last checked in by:

Author:

rhydera

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

- epicsqt/ca_framework/plugins/include/GuiPushButtonPlugin.h

12.10 GuiPushButtonPluginManager Class Reference

Launch GUI Push Button Widget Plugin Manager for designer.

```
#include <GuiPushButtonPluginManager.h>
```

Public Member Functions

- **GuiPushButtonPluginManager** (QObject *parent=0)
- bool **isContainer** () const
- bool **isInitialized** () const
- QIcon **icon** () const
- QString **group** () const
- QString **includeFile** () const
- QString **name** () const
- QString **toolTip** () const
- QString **whatsThis** () const
- QWidget * **createWidget** (QWidget *parent)
- void **initialize** (QDesignerFormEditorInterface *core)

12.10.1 Detailed Description

Launch GUI Push Button Widget Plugin Manager for designer. ???

Version

Revision:

#4

Date

DateTime:

2010/06/23 07:49:40

Author

Last checked in by:

Author:

rhydera

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

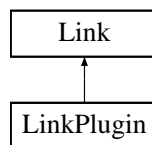
- epicsqt/ca_framework/plugins/include/GuiPushButtonPluginManager.h

12.11 Link Class Reference

Manage updating tool tip with variable name, alarm state and connected state.

```
#include <Link.h>
```

Inheritance diagram for Link:



Public Types

- enum **conditions** {
 CONDITION_EQ, CONDITION_NE, CONDITION_GT, CONDITION_-
 GE,
 CONDITION_LT, CONDITION_LE }

Public Slots

- void **in** (const bool &in)
- void **in** (const qlonglong &in)
- void **in** (const double &in)
- void **in** (const QString &in)
- void **autoFillBackground** (const bool &enable)

Signals

- void **out** (const bool &out)
- void **out** (const qlonglong &out)
- void **out** (const double &out)
- void **out** (const QString &out)

Public Member Functions

- **Link** (QWidget *parent=0)
- void **setCondition** (conditions conditionIn)
- conditions **getCondition** ()
- void **setComparisonValue** (QString comparisonValue)
- QString **getComparisonValue** ()
- void **setSignalTrue** (bool signalTrue)
- bool **getSignalTrue** ()
- void **setSignalFalse** (bool signalFalse)
- bool **getSignalFalse** ()
- void **setOutTrueValue** (QString outTrueValue)
- QString **getOutTrueValue** ()
- void **setOutFalseValue** (QString outFalseValue)
- QString **getOutFalseValue** ()
- void **setRunVisible** (bool visibleIn)
- bool **getRunVisible** ()

Protected Attributes

- conditions **condition**
- QVariant **comparisonValue**
- bool **signalTrue**
- bool **signalFalse**
- QVariant **outTrueValue**
- QVariant **outFalseValue**
- bool **visible**

12.11.1 Detailed Description

Manage updating tool tip with variable name, alarm state and connected state.

Version

Revision:

#8

Date

DateTime:

2010/09/06 13:16:04

Author

andrew.rhyder

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

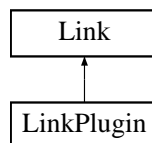
- epicsqt/ca_framework/widgets/include/Link.h

12.12 LinkPlugin Class Reference

[Link](#) Widget Plugin for designer.

```
#include <LinkPlugin.h>
```

Inheritance diagram for LinkPlugin:



Public Types

- enum **ConditionNames** {
 Equal = Link::CONDITION_EQ, **NotEqual** = Link::CONDITION_NE, **Greater**
 Than = Link::CONDITION_GT, **GreaterThanOrEqualTo** = Link::CONDITION_GE,
 LessThan = Link::CONDITION_LT, **LessThanOrEqualTo** = Link::CONDITION_-
 LE }

Public Member Functions

- [LinkPlugin](#) (QWidget *parent=0)
Constructors.
- void **setConditionProperty** (ConditionNames condition)
- ConditionNames **getConditionProperty** ()

Properties

- ConditionNames [condition](#)
Qt Designer Properties - condition.
- QString **comparisonValue**
- bool **signalTrue**
- bool **signalFalse**
- QString **outTrueValue**
- QString **outFalseValue**

12.12.1 Detailed Description

[Link](#) Widget Plugin for designer.

Version

Revision:

#6

Date

DateTime:

2010/09/06 11:58:56

Author

andrew.rhyder

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

- epicsqt/ca_framework/plugins/include/LinkPlugin.h

12.13 LinkPluginManager Class Reference

[Link](#) Widget Plugin Manager for designer.

```
#include <LinkPluginManager.h>
```

Public Member Functions

- **LinkPluginManager** (QObject *parent=0)
- bool **isContainer** () const
- bool **isInitialized** () const
- QIcon **icon** () const
- QString **group** () const
- QString **includeFile** () const
- QString **name** () const
- QString **toolTip** () const
- QString **whatsThis** () const
- QWidget * **createWidget** (QWidget *parent)
- void **initialize** (QDesignerFormEditorInterface *core)

12.13.1 Detailed Description

[Link](#) Widget Plugin Manager for designer. ???

Version

Revision:

#3

Date

DateTime:

2010/06/23 07:49:40

Author

andrew.rhyder

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

- epicsqt/ca_framework/plugins/include/LinkPluginManager.h

12.14 localEnumerationItem Class Reference

Public Types

- enum **operations** {
LESS, LESS_EQUAL, EQUAL, NOT_EQUAL,
GREATER_EQUAL, GREATER, ALWAYS, UNKNOWN }

Public Attributes

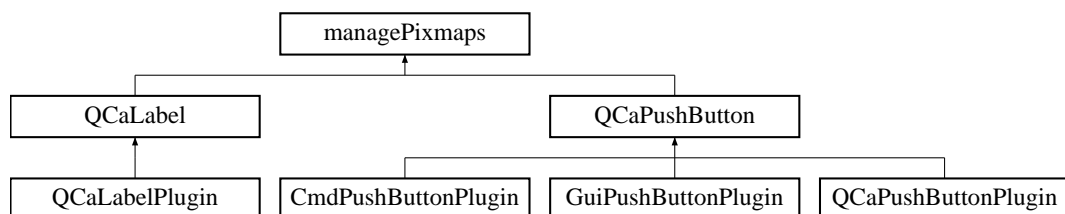
- int **value**
- operations **op**
- QString **text**

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

- epicsqt/ca_framework/data/include/QCaStringFormatting.h

12.15 managePixmaps Class Reference

Inheritance diagram for managePixmaps:



Public Member Functions

- void **setDataPixmap** (QPixmap &Pixmap, unsigned int index)
- QPixmap **getDataPixmap** (unsigned int index)
- QPixmap **getDataPixmap** (QString value)

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

- epicsqt/ca_framework/widgets/include/managePixmaps.h

12.16 PeriodicDialog Class Reference

Public Member Functions

- **PeriodicDialog** (QWidget *parent=0)
- QString **getElement** ()
- void **setElement** (QString elementIn, QList< bool > &enabledList, QList< QString > &elementList)

Protected Member Functions

- void **changeEvent** (QEvent *e)

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

- epicsqt/ca_framework/widgets/include/PeriodicDialog.h

12.17 PeriodicElementSetupForm Class Reference

Public Member Functions

- **PeriodicElementSetupForm** (QWidget *parent=0)

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

- epicsqt/ca_framework/plugins/include/PeriodicElementSetupForm.h

12.18 PeriodicSetupDialog Class Reference

Public Member Functions

- **PeriodicSetupDialog** (QWidget *parent=0)

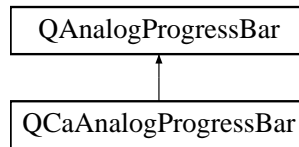
The documentation for this class was generated from the following file:

- epicsqt/ca_framework/plugins/include/PeriodicSetupDialog.h

12.19 QAnalogProgressBar Class Reference

```
#include <QAnalogProgressBar.h>
```

Inheritance diagram for QAnalogProgressBar:



Public Slots

- void **setAnalogRange** (const double analogMinimumIn, const double analogMaximumIn)
- void **setAnalogMinimum** (const double analogMinimumIn)
- void **setAnalogMaximum** (const double analogMaximumIn)
- void **setAnalogValue** (const double analogValueIn)

Public Member Functions

- [QAnalogProgressBar](#) (QWidget *parent=0)
Constructor.
- double **getAnalogMinimum** ()
- double **getAnalogMaximum** ()
- double **getAnalogValue** ()

Protected Member Functions

- virtual QString **getSprintfFormat** ()

Properties

- double **analogMinimum**
- double **analogMaximum**
- double **analogValue**

12.19.1 Detailed Description

This class inherits directly from QProgressBar and provides an analog view of the progress bar widget including analogous analog functions and properties:

QProgressBar [QAnalogProgressBar](#)

setRange() setAnalogRange() setMinimum() setAnalogMinimum() getMinimum() getAnalogMinimum() setMaximum() setAnalogMaximum() getMaximum() getAnalogMaximum() setValue() setAnalogValue() getValue() getAnalogValue()

Minimum analogMinimum Maximum analogMaximum Value analogValue

The class sets the range of of the underlying `QProgressBar` to 0 .. 10000. It is considered unlikely, in practice, that an `AnalogProgressBar` will stretch more than 10000 pixels given that current screen technology is limited to 2000 pixels.

This class inherited from rather than containing a `QProgressBar`, and as such cannot hide and does not override the integer related base class functions and properties. Therefore they should not be used.

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

- `epicsqt/ca_framework/widgets/include/QAnalogProgressBar.h`

12.20 QAnalogProgressBarPluginManager Class Reference

```
#include <QAnalogProgressBarPluginManager.h>
```

Public Member Functions

- **QAnalogProgressBarPluginManager** (`QObject *parent=0`)
- `bool isContainer () const`
- `bool isInitialized () const`
- `QIcon icon () const`
- `QString group () const`
- `QString includeFile () const`
- `QString name () const`
- `QString toolTip () const`
- `QString whatsThis () const`
- `QWidget * createWidget (QWidget *parent)`
- `void initialize (QDesignerFormEditorInterface *core)`

12.20.1 Detailed Description

???

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

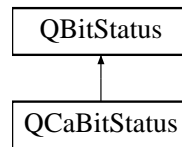
- `epicsqt/ca_framework/plugins/include/QAnalogProgressBarPluginManager.h`

12.21 QBitStatus Class Reference

Bit Status Widget.

```
#include <QBitStatus.h>
```

Inheritance diagram for `QBitStatus`:



Public Types

- enum **Orientations** { **LSB_On_Right**, **LSB_On_Bottom**, **LSB_On_Left**, **LSB_On_Top** }

Public Slots

- void **setValue** (const long value)

Public Member Functions

- **QBitStatus** (QWidget *parent=0)
- virtual QSize **sizeHint** () const
- void **setBorderColour** (const QColor value)
- QColor **getBorderColour** ()
- void **setOnColour** (const QColor value)
- QColor **getOnColour** ()
- void **setOffColour** (const QColor value)
- QColor **getOffColour** ()
- void **setInvalidColour** (const QColor value)
- QColor **getInvalidColour** ()
- void **setDrawBorder** (const bool value)
- bool **getDrawBorder** ()
- void **setNumberOfBits** (const int value)
- int **getNumberOfBits** ()
- void **setShift** (const int value)
- int **getShift** ()
- void **setIsValid** (const bool value)
- bool **getIsValid** ()
- void **setOrientation** (const enum Orientations value)
- enum Orientations **getOrientation** ()
- long **getValue** ()

Properties

- int **value**
- int **numberOfBits**
- int **shift**

- QColor **boarderColour**
- QColor **invalidColour**
- QColor **onColour**
- QColor **offColour**
- bool **drawBorder**
- bool **isValid**
- Orientations **Orientation**

12.21.1 Detailed Description

Bit Status Widget.

Version

Revision:

#3

Date

DateTime:

2011/06/12 15:07:03

Author

andrew.starritt

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

- epicsqt/ca_framework/widgets/include/QBitStatus.h

12.22 QBitStatusPluginManager Class Reference

CA ProgressBar Widget Plugin Manager for designer.

```
#include <QBitStatusPluginManager.h>
```

Public Member Functions

- **QBitStatusPluginManager** (QObject *parent=0)
- bool **isContainer** () const
- bool **isInitialized** () const
- QIcon **icon** () const
- QString **group** () const

- QString **includeFile** () const
- QString **name** () const
- QString **toolTip** () const
- QString **whatsThis** () const
- QWidget * **createWidget** (QWidget *parent)
- void **initialize** (QDesignerFormEditorInterface *core)

12.22.1 Detailed Description

CA ProgressBar Widget Plugin Manager for designer. ???

Version

Revision:

#1

Date

DateTime:

2011/06/11 16:35:50

Author

andrew.starritt

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

- epicsqt/ca_framework/plugins/include/QBitStatusPluginManager.h

12.23 QCaAlarmInfo Class Reference

Public Member Functions

- **QCaAlarmInfo** (unsigned short statusIn, unsigned short severityIn)
- QString **statusName** ()
- QString **severityName** ()
- bool **isInAlarm** ()
- bool **isMinor** ()
- bool **isMajor** ()
- bool **isInvalid** ()
- QString **style** ()
- QCAALARMINFO_SEVERITY **getSeverity** ()

Static Public Member Functions

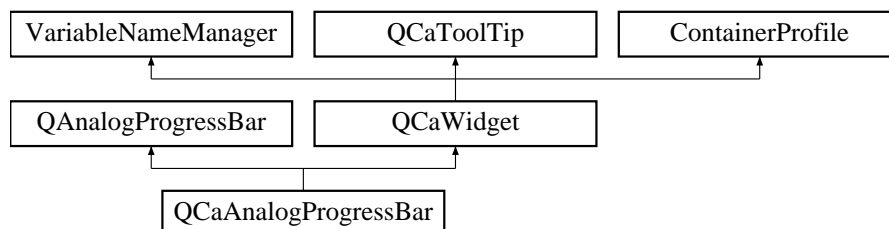
- static QCAALARMINFO_SEVERITY **getInvalidSeverity** ()

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

- epicsqt/ca_framework/data/include/QCaAlarmInfo.h

12.24 QCaAnalogProgressBar Class Reference

Inheritance diagram for QCaAnalogProgressBar:



Public Slots

- void **requestEnabled** (const bool &state)

Signals

- void **dbValueChanged** (const double &out)

Public Member Functions

- **QCaAnalogProgressBar** (QWidget *parent=0)
- **QCaAnalogProgressBar** (const QString &variableName, QWidget *parent=0)
- bool **isEnabled** () const
- void **setEnabled** (bool state)
- void **setVariableNameProperty** (QString variableName)
- QString **getVariableNameProperty** ()
- void **setVariableNameSubstitutionsProperty** (QString variableNameSubstitutions)
- QString **getVariableNameSubstitutionsProperty** ()
- void **setVariableNameAndSubstitutions** (QString variableNameIn, QString variableNameSubstitutionsIn, unsigned int variableIndex)
- void **setVariableAsToolTip** (bool variableAsToolTip)
- bool **getVariableAsToolTip** ()
- void **setUseDbPrecision** (bool useDbPrecisionIn)

- bool **getDbPrecision** ()
- void **setDbDisplayLimits** (bool useDbDisplayLimitsIn)
- bool **getDbDisplayLimits** ()
- void **setRunVisible** (bool visibleIn)
- bool **getRunVisible** ()

Protected Member Functions

- virtual QString **getSprintfFormat** ()
- void **establishConnection** (unsigned int variableIndex)
Initialise macro substitutions. May be used on creation.

Protected Attributes

- [QCaFloatingFormatting](#) **floatingFormatting**
- bool **localEnabled**

Properties

- QString **variable**
- QString **variableSubstitutions**
- bool **variableAsToolTip**
Flag if data updates should be requested.
- bool **enabled**
- bool **useDbDisplayLimits**
Display properties.
- bool **visible**

12.24.1 Property Documentation

12.24.1.1 QString QCaAnalogProgressBar::variable [read, write]

Note, a property macro in the form 'Q_PROPERTY(QString variableName READ ...)' doesn't work. A property name ending with 'Name' results in some sort of string a variable being displayed, but will only accept alphanumeric and won't generate callbacks on change.

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

- `epicsqt/ca_framework/widgets/include/QCaAnalogProgressBar.h`

12.25 QCaAnalogProgressBarPluginManager Class Reference

```
#include <QCaAnalogProgressBarPluginManager.h>
```

Public Member Functions

- **QCaAnalogProgressBarPluginManager** (QObject *parent=0)
- bool **isContainer** () const
- bool **isInitialized** () const
- QIcon **icon** () const
- QString **group** () const
- QString **includeFile** () const
- QString **name** () const
- QString **toolTip** () const
- QString **whatsThis** () const
- QWidget * **createWidget** (QWidget *parent)
- void **initialize** (QDesignerFormEditorInterface *core)

12.25.1 Detailed Description

???

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

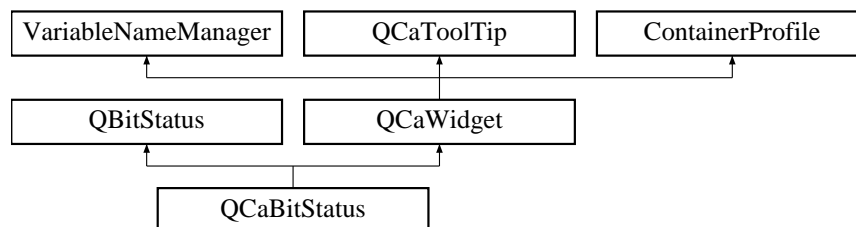
- epicsqt/ca_framework/plugins/include/QCaAnalogProgressBarPluginManager.h

12.26 QCaBitStatus Class Reference

CA Bit Status Widget.

```
#include <QCaBitStatus.h>
```

Inheritance diagram for QCaBitStatus:



Public Slots

- void **requestEnabled** (const bool &state)

Signals

- void **dbValueChanged** (const long &out)

Public Member Functions

- [QCaBitStatus](#) (QWidget *parent=0)
#endif
- **QCaBitStatus** (const QString &variableName, QWidget *parent=0)
- bool **isEnabled** () const
- void **setEnabled** (bool state)
- void **setVariableNameProperty** (QString variableName)
- QString **getVariableNameProperty** ()
- void **setVariableNameSubstitutionsProperty** (QString variableNameSubstitutions)
- QString **getVariableNameSubstitutionsProperty** ()
- void **setVariableNameAndSubstitutions** (QString variableNameIn, QString variableNameSubstitutionsIn, unsigned int variableIndex)
- void **setVariableAsToolTip** (bool variableAsToolTip)
- bool **getVariableAsToolTip** ()
- void **setRunVisible** (bool visibleIn)
- bool **getRunVisible** ()

Protected Member Functions

- void [establishConnection](#) (unsigned int variableIndex)
Initialise macro substitutions. May be used on creation.

Protected Attributes

- [QCaIntegerFormatting](#) **integerFormatting**
- bool **localEnabled**

Properties

- QString [variable](#)
#ifdef PLUGIN_APP
- QString **variableSubstitutions**
- bool [variableAsToolTip](#)
Flag if data updates should be requested.
- bool **enabled**
- bool **visible**

12.26.1 Detailed Description

CA Bit Status Widget.

Version

Revision:

#1

Date

DateTime:

2011/06/11 21:36:00

Author

andrew.starritt

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

- epicsqt/ca_framework/widgets/include/QCaBitStatus.h

12.27 QCaBitStatusPluginManager Class Reference

CA ProgressBar Widget Plugin Manager for designer.

```
#include <QCaBitStatusPluginManager.h>
```

Public Member Functions

- **QCaBitStatusPluginManager** (QObject *parent=0)
- bool **isContainer** () const
- bool **isInitialized** () const
- QIcon **icon** () const
- QString **group** () const
- QString **includeFile** () const
- QString **name** () const
- QString **toolTip** () const
- QString **whatsThis** () const
- QWidget * **createWidget** (QWidget *parent)
- void **initialize** (QDesignerFormEditorInterface *core)

12.27.1 Detailed Description

CA ProgressBar Widget Plugin Manager for designer. ???

Version

Revision:

#1

Date

DateTime:

2011/06/11 21:36:00

Author

andrew.starritt

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

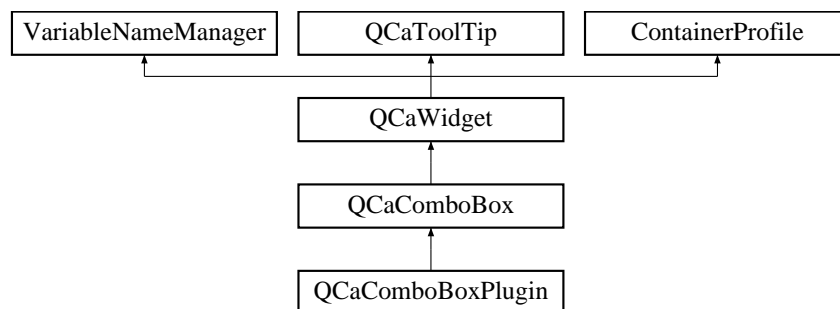
- epicsqt/ca_framework/plugins/include/QCaBitStatusPluginManager.h

12.28 QCaComboBox Class Reference

CA Combobox Widget.

```
#include <QCaComboBox.h>
```

Inheritance diagram for QCaComboBox:



Public Slots

- void **requestEnabled** (const bool &state)

Signals

- void **dbValueChanged** (const qulonglong &out)

Public Member Functions

- **QCaComboBox** (QWidget *parent=0)
- **QCaComboBox** (const QString &variableName, QWidget *parent=0)
- bool **isEnabled** () const
- void **setEnabled** (bool state)
- void **setVariableNameAndSubstitutions** (QString variableNameIn, QString variableNameSubstitutionsIn, unsigned int variableIndex)
- void **setSubscribe** (bool [subscribe](#))
- bool **getSubscribe** ()
- void **setVariableAsToolTip** (bool [variableAsToolTip](#))
- bool **getVariableAsToolTip** ()
- void **setUseDbEnumerations** (bool useDbEnumerations)
- bool **getUseDbEnumerations** ()

Protected Member Functions

- void [establishConnection](#) (unsigned int variableIndex)
Initialise macro substitutions. May be used on creation.

Protected Attributes

- [QCaIntegerFormatting](#) **integerFormatting**
- bool **useDbEnumerations**
- bool **localEnabled**

12.28.1 Detailed Description

CA Combobox Widget.

Version

Revision:

#14

Date

DateTime:

2010/09/06 13:16:04

Author

andrew.rhyder

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

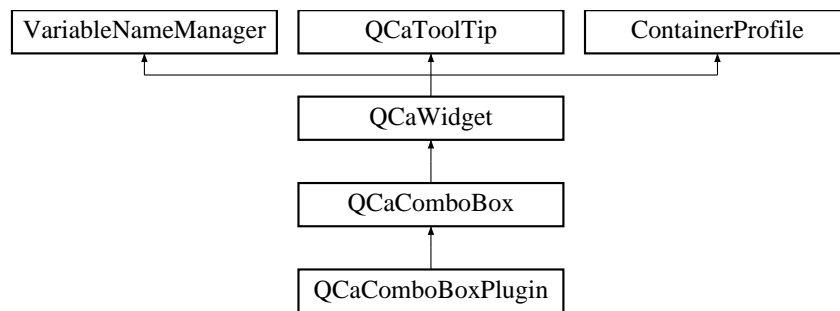
- epicsqt/ca_framework/widgets/include/QCaComboBox.h

12.29 QCaComboBoxPlugin Class Reference

CA Combobox Widget Plugin.

```
#include <QCaComboBoxPlugin.h>
```

Inheritance diagram for QCaComboBoxPlugin:

**Public Member Functions**

- **QCaComboBoxPlugin** (QWidget *parent=0)
- **QCaComboBoxPlugin** (QString variableName, QWidget *parent=0)
- void **setVariableNameProperty** (QString variableName)
- QString **getVariableNameProperty** ()
- void **setVariableNameSubstitutionsProperty** (QString variableNameSubstitutions)
- QString **getVariableNameSubstitutionsProperty** ()

Properties

- QString **variable**
- QString **variableSubstitutions**
- bool **subscribe**

Set the number of variables that will stream data updates to the widget. Default of 1 if not called.

- bool [variableAsToolTip](#)
Flag if data updates should be requested.
- bool **enabled**
- bool **useDbEnumerations**

12.29.1 Detailed Description

CA Combobox Widget Plugin.

Version

Revision:

#7

Date

DateTime:

2010/09/06 11:58:56

Author

andrew.rhyder

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

- `epicsqt/ca_framework/plugins/include/QCaComboBoxPlugin.h`

12.30 QCaComboBoxPluginManager Class Reference

CA Combobox Widget Plugin Manager for designer.

```
#include <QCaComboBoxPluginManager.h>
```

Public Member Functions

- **QCaComboBoxPluginManager** (QObject *parent=0)
- bool **isContainer** () const
- bool **isInitialized** () const
- QIcon **icon** () const
- QString **group** () const
- QString **includeFile** () const

- QString **name** () const
- QString **toolTip** () const
- QString **whatsThis** () const
- QWidget * **createWidget** (QWidget *parent)
- void **initialize** (QDesignerFormEditorInterface *core)

12.30.1 Detailed Description

CA Combobox Widget Plugin Manager for designer.

Version

Revision:

#4

Date

DateTime:

2010/06/23 07:49:40

Author

andrew.rhyder

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

- epicsqt/ca_framework/plugins/include/QCaComboBoxPluginManager.h

12.31 QCaConnectionInfo Class Reference

Public Member Functions

- **QCaConnectionInfo** (unsigned short channelStateIn, unsigned short linkStateIn)
- bool **isChannelConnected** ()
- bool **isLinkUp** ()

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

- epicsqt/ca_framework/data/include/QCaConnectionInfo.h

12.32 QCaDateTime Class Reference

Public Member Functions

- **QCaDateTime** (QDateTime dt)
- void **operator=** (const [QCaDateTime](#) &other)
- **QCaDateTime** (unsigned long seconds, unsigned long nanoseconds)
- QString **text** ()
- double **floating** (QDateTime base)

Public Attributes

- unsigned long **nSec**

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

- epicsqt/ca_framework/data/include/QCaDateTime.h

12.33 QCaDesignerPlugin Class Reference

Collection of CA plugins for QT Designer.

```
#include <QCaDesignerPlugin.h>
```

12.33.1 Detailed Description

Collection of CA plugins for QT Designer.

Version

Revision:

#3

Date

DateTime:

2010/06/23 07:49:40

Author

andrew.rhyder

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

- epicsqt/ca_framework/plugins/include/QCaDesignerPlugin.h

12.34 QCaEventFilter Class Reference

Manage CA alarm and severity information.

```
#include <QCaEventFilter.h>
```

Public Member Functions

- void [addFilter](#) (QObject *objectIn)
Add a filter to an object.
- void [deleteFilter](#) (QObject *objectIn)
Remove an event filter from an object.
- bool [eventFilter](#) (QObject *watched, QEvent *e)
Event filter added to a QObject.

12.34.1 Detailed Description

Manage CA alarm and severity information. CA event filter for QT.

CA Date Time manager.

Manage CA connection information.

This class allows instances of [QCaObject](#) to receive events posted to them by foreign CA threads. Each event recipient creates an instance of this class. This class, however, manages only a single event filter for all instances of itself.

Version

Revision:

#5

Date

DateTime:

2010/06/23 07:49:40

Author

andrew.rhyder

Version

Revision:

#3

Date**DateTime:**

2010/06/23 07:49:40

Author

andrew.rhyder

Version**Revision:**

#4

Date**DateTime:**

2010/08/30 16:37:08

Author

andrew.rhyder

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

- `epicsqt/ca_framework/data/include/QCaEventFilter.h`

12.35 QCaEventItem Class Reference

Simple container for a list [QCaEventUpdate](#) instances.

```
#include <QCaEventUpdate.h>
```

Public Member Functions

- [QCaEventItem](#) ([QCaEventUpdate](#) *newEvent)

Public Attributes

- [QCaEventUpdate](#) * event

12.35.1 Detailed Description

Simple container for a list [QCaEventUpdate](#) instances. This class is simply a reference to an update event (an instance of class [QCaEventUpdate](#)) This class exists so that a list of references to outstanding events can be maintained.

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

- `epicsqt/ca_framework/data/include/QCaEventUpdate.h`

12.36 QCaEventUpdate Class Reference

Packages CA callback data as a Qt event.

```
#include <QCaEventUpdate.h>
```

Public Member Functions

- **QCaEventUpdate** ([qcaobject::QCaObject](#) *emitterObjectIn, long newReason, void *newDataPtr)

Public Attributes

- bool **acceptThisEvent**
- [qcaobject::QCaObject](#) * **emitterObject**
If true, the originator of this event has been deleted. Ignore the event.
- long **reason**
The object to process the data and emit a signal.
- void * **dataPtr**
Callback reason. Actually of type `caobject::callback_reasons`.

Static Public Attributes

- static `QEvent::Type` **EVENT_UPDATE_TYPE**

12.36.1 Detailed Description

Packages CA callback data as a Qt event.

Version

Revision:

#4

Date**DateTime:**

2010/06/23 07:49:40

Author

andrew.rhyder

This class packages data and status generated by a CA callback as a Qt event. When data or status updates must be passed from a foreign CA thread to a Qt thread an instance of this class is created, populated with the CA information and posted through the event mechanism to a Qt thread.

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

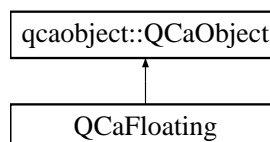
- `epicsqt/ca_framework/data/include/QCaEventUpdate.h`

12.37 QCaFloating Class Reference

Floating specific wrapper for [QCaObject](#).

```
#include <QCaFloating.h>
```

Inheritance diagram for QCaFloating:

**Public Slots**

- void **writeFloating** (const double &data)

Signals

- void **floatingChanged** (const double &value, [QCaAlarmInfo](#) &alarmInfo, [QCaDateTime](#) &timeStamp, const unsigned int &variableIndex)
- void **floatingArrayChanged** (const QVector< double > &values, [QCaAlarmInfo](#) &alarmInfo, [QCaDateTime](#) &timeStamp, const unsigned int &variableIndex)

Public Member Functions

- **QCaFloating** (QString recordName, QObject *eventObject, [QCaFloatingFormatting](#) *floatingFormattingIn, unsigned int variableIndexIn)
- **QCaFloating** (QString recordName, QObject *eventObject, [QCaFloatingFormatting](#) *floatingFormattingIn, unsigned int variableIndexIn, [UserMessage](#) *userMessageIn)

12.37.1 Detailed Description

Floating specific wrapper for [QCaObject](#).

Version

Revision:

#5

Date

DateTime:

2010/09/06 13:16:04

Author

andrew.rhyder

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

- `epicsqt/ca_framework/data/include/QCaFloating.h`

12.38 QCaFloatingFormatting Class Reference

Provides textual formatting for [QCaFloating](#) data.

```
#include <QCaFloatingFormatting.h>
```

Public Types

- enum **formats** {
 FORMAT_e = 'e', **FORMAT_E** = 'E', **FORMAT_f** = 'f', **FORMAT_g** = 'g',
 FORMAT_G = 'G' }

Public Member Functions

- double **formatFloating** (const QVariant &value)
- QVector< double > **formatFloatingArray** (const QVariant &value)
- QVariant **formatValue** (const double &floatingValue, generic::generic_types valueType)
- void **setPrecision** (unsigned int precision)
- void **setFormat** (formats format)
- unsigned int **getPrecision** ()
- int **getFormat** ()

12.38.1 Detailed Description

Provides textual formatting for [QCaFloating](#) data.

Version

Revision:

#4

Date

DateTime:

2010/08/30 16:37:08

Author

andrew.rhyder

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

- epicsqt/ca_framework/data/include/QCaFloatingFormatting.h

12.39 QCaInstalledFiltersListItem Class Reference

Public Member Functions

- **QCaInstalledFiltersListItem** (QObject *eventObjectIn)

Public Attributes

- `QObject *` [eventObject](#)
QObject that an event filter has been added to.
- `long` [referenceCount](#)
Count of how many QCaObjects expect the event filter to be on the QObject.

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

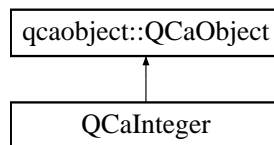
- `epicsqt/ca_framework/data/include/QCaEventFilter.h`

12.40 QCaInteger Class Reference

Integer specific wrapper for [QCaObject](#).

```
#include <QCaInteger.h>
```

Inheritance diagram for QCaInteger:



Public Slots

- `void` **writeInteger** (`const long &data`)

Signals

- `void` **integerChanged** (`const long &value`, [QCaAlarmInfo](#) &alarmInfo, [QCaDateTime](#) &timeStamp, `const unsigned int &variableIndex`)

Public Member Functions

- **QCaInteger** (`QString recordName`, `QObject *eventObject`, [QCaIntegerFormatting](#) *integerFormattingIn, `unsigned int variableIndexIn`)
- **QCaInteger** (`QString recordName`, `QObject *eventObject`, [QCaIntegerFormatting](#) *integerFormattingIn, `unsigned int variableIndexIn`, [UserMessage](#) *userMessageIn)

12.40.1 Detailed Description

Integer specific wrapper for [QCaObject](#).

Version

Revision:

#6

Date

DateTime:

2010/09/06 13:16:04

Author

andrew.rhyder

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

- `epicsqt/ca_framework/data/include/QCaInteger.h`

12.41 QCaIntegerFormatting Class Reference

Provides textual formatting for [QCaInteger](#) data.

```
#include <QCaIntegerFormatting.h>
```

Public Member Functions

- long **formatInteger** (const QVariant &value)
- QVariant **formatValue** (const long &integerValue, generic::generic_types value-Type)
- void **setRadix** (unsigned int radix)
- unsigned int **getPrecision** ()
- unsigned int **getRadix** ()

12.41.1 Detailed Description

Provides textual formatting for [QCaInteger](#) data.

Version

Revision:

#4

Date**DateTime:**

2010/06/23 07:49:40

Author

andrew.rhyder

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

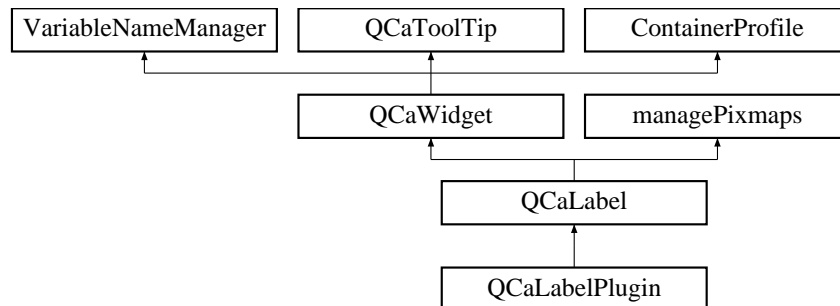
- epicsqt/ca_framework/data/include/QCaIntegerFormatting.h

12.42 QCaLabel Class Reference

Manage updating tool tip with variable name, alarm state and connected state.

```
#include <QCaLabel.h>
```

Inheritance diagram for QCaLabel:

**Public Types**

- enum **updateOptions** { **UPDATE_TEXT**, **UPDATE_PIXMAP** }

Public Slots

- void **requestEnabled** (const bool &state)

Signals

- void **dbValueChanged** (const QString &out)
- void **requestResend** ()

Public Member Functions

- **QCaLabel** (QWidget *parent=0)
- **QCaLabel** (const QString &variableName, QWidget *parent=0)
- bool **isEnabled** () const
- void **setEnabled** (bool state)
- void **setUpdateOption** (updateOptions updateOptionIn)
- updateOptions **getUpdateOption** ()
- void **setVariableNameAndSubstitutions** (QString variableNameIn, QString variableNameSubstitutionsIn, unsigned int variableIndex)
- void **setVariableAsToolTip** (bool [variableAsToolTip](#))
- bool **getVariableAsToolTip** ()
- void **setPrecision** (unsigned int precision)
- unsigned int **getPrecision** ()
- void **setUseDbPrecision** (bool useDbPrecision)
- bool **getUseDbPrecision** ()
- void **setLeadingZero** (bool leadingZero)
- bool **getLeadingZero** ()
- void **setTrailingZeros** (bool trailingZeros)
- bool **getTrailingZeros** ()
- void **setAddUnits** (bool addUnits)
- bool **getAddUnits** ()
- void **setLocalEnumeration** (QString localEnumeration)
- QString **getLocalEnumeration** ()
- void **setFormat** (QCaStringFormatting::formats format)
- QCaStringFormatting::formats **getFormat** ()
- void **setRadix** (unsigned int radix)
- unsigned int **getRadix** ()
- void **setNotation** (QCaStringFormatting::notations notation)
- QCaStringFormatting::notations **getNotation** ()
- void **setArrayAction** (QCaStringFormatting::arrayActions arrayAction)
- QCaStringFormatting::arrayActions **getArrayAction** ()
- void **setArrayIndex** (unsigned int arrayIndex)
- unsigned int **getArrayIndex** ()
- void **setRunVisible** (bool visibleIn)
- bool **getRunVisible** ()

Protected Member Functions

- void [establishConnection](#) (unsigned int variableIndex)
Initialise macro substitutions. May be used on creation.

Protected Attributes

- [QCaStringFormatting](#) **stringFormatting**
- bool **caEnabled**
- bool **caVisible**
- updateOptions **updateOption**

12.42.1 Detailed Description

Manage updating tool tip with variable name, alarm state and connected state.

Version**Revision:**

#13

Date**DateTime:**

2010/09/06 13:16:04

Author

andrew.rhyder

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

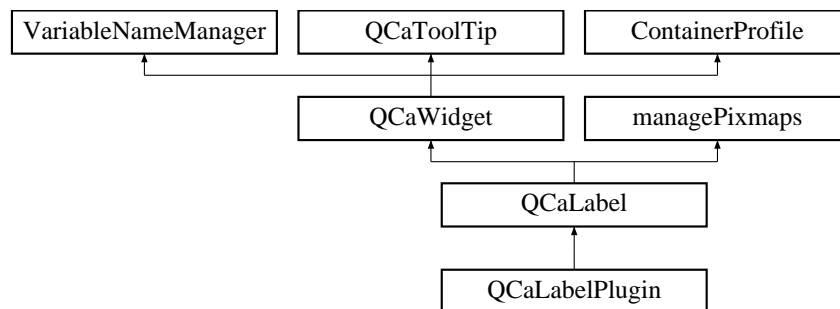
- `epicsqt/ca_framework/widgets/include/QCaLabel.h`

12.43 QCaLabelPlugin Class Reference

CA Label Widget Plugin for designer.

```
#include <QCaLabelPlugin.h>
```

Inheritance diagram for QCaLabelPlugin:



Public Types

- enum **UpdateOptions** { **Text** = QCaLabel::UPDATE_TEXT, **Picture** = QCaLabel::UPDATE_PIXMAP }
- enum **Formats** {
Default = QCaStringFormatting::FORMAT_DEFAULT, **Floating** = QCaStringFormatting::FORMAT_FLOATING, **Integer** = QCaStringFormatting::FORMAT_INTEGER, **UnsignedInteger** = QCaStringFormatting::FORMAT_UNSIGNEDINTEGER,
Time = QCaStringFormatting::FORMAT_TIME, **LocalEnumeration** = QCaStringFormatting::FORMAT_LOCAL_ENUMERATE }
- enum **Notations** { **Fixed** = QCaStringFormatting::NOTATION_FIXED, **Scientific** = QCaStringFormatting::NOTATION_SCIENTIFIC, **Automatic** = QCaStringFormatting::NOTATION_AUTOMATIC }
- enum **ArrayActions** { **Append** = QCaStringFormatting::APPEND, **Ascii** = QCaStringFormatting::ASCII, **Index** = QCaStringFormatting::INDEX }

Public Member Functions

- [QCaLabelPlugin](#) (QWidget *parent=0)
Constructors.
- **QCaLabelPlugin** (QString variableName, QWidget *parent=0)
- void **setVariableNameProperty** (QString variableName)
- QString **getVariableNameProperty** ()
- void **setVariableNameSubstitutionsProperty** (QString variableNameSubstitutions)
- QString **getVariableNameSubstitutionsProperty** ()
- void **setUpdateOptionProperty** (UpdateOptions updateOption)
- UpdateOptions **getUpdateOptionProperty** ()
- void **setPixmap0Property** (QPixmap pixmap)
- void **setPixmap1Property** (QPixmap pixmap)
- void **setPixmap2Property** (QPixmap pixmap)
- void **setPixmap3Property** (QPixmap pixmap)
- void **setPixmap4Property** (QPixmap pixmap)
- void **setPixmap5Property** (QPixmap pixmap)
- void **setPixmap6Property** (QPixmap pixmap)
- void **setPixmap7Property** (QPixmap pixmap)
- QPixmap **getPixmap0Property** ()
- QPixmap **getPixmap1Property** ()
- QPixmap **getPixmap2Property** ()
- QPixmap **getPixmap3Property** ()
- QPixmap **getPixmap4Property** ()
- QPixmap **getPixmap5Property** ()
- QPixmap **getPixmap6Property** ()
- QPixmap **getPixmap7Property** ()
- void **setFormatProperty** (Formats format)

- Formats **getFormatProperty** ()
- void **setNotationProperty** (Notations notation)
- Notations **getNotationProperty** ()
- void **setArrayActionProperty** (ArrayActions arrayAction)
- ArrayActions **getArrayActionProperty** ()

Properties

- QString **variable**
- QString **variableSubstitutions**
- bool **variableAsToolTip**
Flag if data updates should be requested.
- bool **enabled**
- UpdateOptions **updateOption**
Update options (text, pixmap, or both)
- QPixmap **pixmap0**
Pixmaps.
- QPixmap **pixmap1**
- QPixmap **pixmap2**
- QPixmap **pixmap3**
- QPixmap **pixmap4**
- QPixmap **pixmap5**
- QPixmap **pixmap6**
- QPixmap **pixmap7**
- unsigned **int**
String formatting properties.
- bool **useDbPrecision**
- bool **leadingZero**
- bool **trailingZeros**
- bool **addUnits**
- QString **localEnumeration**
- Formats **format**
- Notations **notation**
- ArrayActions **arrayAction**
- bool **visible**

12.43.1 Detailed Description

CA Label Widget Plugin for designer.

Version

Revision:

#11

Date

DateTime:

2010/09/06 11:58:56

Author

andrew.rhyder

12.43.2 Property Documentation

12.43.2.1 QString QCaLabelPlugin::variable [read, write]

Note, a property macro in the form 'Q_PROPERTY(QString variableName READ ...' doesn't work. A property name ending with 'Name' results in some sort of string a variable being displayed, but will only accept alphanumeric and won't generate call-backs on change.

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

- epicsqt/ca_framework/plugins/include/QCaLabelPlugin.h

12.44 QCaLabelPluginManager Class Reference

CA Label Widget Plugin Manager for designer.

```
#include <QCaLabelPluginManager.h>
```

Public Member Functions

- **QCaLabelPluginManager** (QObject *parent=0)
- bool **isContainer** () const
- bool **isInitialized** () const
- QIcon **icon** () const
- QString **group** () const

- QString **includeFile** () const
- QString **name** () const
- QString **toolTip** () const
- QString **whatsThis** () const
- QWidget * **createWidget** (QWidget *parent)
- void **initialize** (QDesignerFormEditorInterface *core)

12.44.1 Detailed Description

CA Label Widget Plugin Manager for designer. ???

Version

Revision:

#4

Date

DateTime:

2010/06/23 07:49:40

Author

andrew.rhyder

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

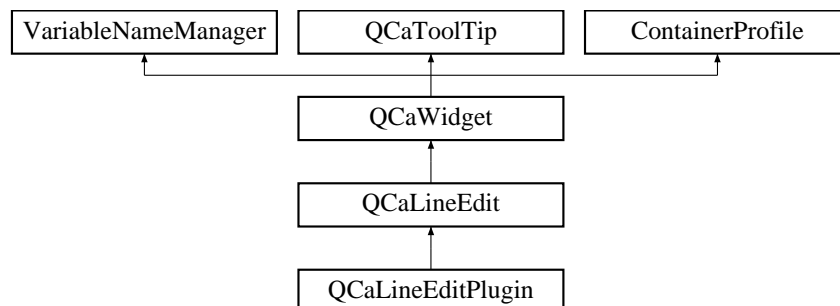
- epicsqt/ca_framework/plugins/include/QCaLabelPluginManager.h

12.45 QCaLineEdit Class Reference

CA Line Edit Widget.

```
#include <QCaLineEdit.h>
```

Inheritance diagram for QCaLineEdit:



Public Slots

- void [requestEnabled](#) (const bool &state)
Act on the user signaling text editing is complete (pressing return)

Signals

- void **dbValueChanged** (const QString &out)

Public Member Functions

- **QCaLineEdit** (QWidget *parent=0)
- **QCaLineEdit** (const QString &variableName, QWidget *parent=0)
- bool **isEnabled** () const
- void **setEnabled** (bool state)
- void **setVariableNameAndSubstitutions** (QString variableNameIn, QString variableNameSubstitutionsIn, unsigned int variableIndex)
- void **setWriteOnLoseFocus** (bool [writeOnLoseFocus](#))
- bool **getWriteOnLoseFocus** ()
- void **setWriteOnEnter** (bool [writeOnEnter](#))
- bool **getWriteOnEnter** ()
- void **setSubscribe** (bool [subscribe](#))
- bool **getSubscribe** ()
- void **setVariableAsToolTip** (bool [variableAsToolTip](#))
- bool **getVariableAsToolTip** ()
- void **setConfirmWrite** (bool [confirmWrite](#))
- bool **getConfirmWrite** ()
- void **setPrecision** (unsigned int precision)
- unsigned int **getPrecision** ()
- void **setUseDbPrecision** (bool useDbPrecision)
- bool **getUseDbPrecision** ()
- void **setLeadingZero** (bool leadingZero)
- bool **getLeadingZero** ()
- void **setTrailingZeros** (bool trailingZeros)
- bool **getTrailingZeros** ()
- void **setAddUnits** (bool addUnits)
- bool **getAddUnits** ()
- void **setFormat** (QCaStringFormatting::formats format)
- QCaStringFormatting::formats **getFormat** ()
- void **setRadix** (unsigned int radix)
- unsigned int **getRadix** ()
- void **setNotation** (QCaStringFormatting::notations notation)
- QCaStringFormatting::notations **getNotation** ()

Protected Member Functions

- void [establishConnection](#) (unsigned int variableIndex)

Request confirmation before writing a value.

Protected Attributes

- [QCaStringFormatting](#) **stringFormatting**

- bool [writeOnLoseFocus](#)

String formatting options.

- bool [writeOnEnter](#)

Write changed value to database when widget object loses focus (user moves from widget)

- bool [localEnabled](#)

Write changed value to database when enter is pressed with focus on the widget.

- bool [confirmWrite](#)

Override the default widget `setEnabled` to allow alarm states to override current enabled state.

12.45.1 Detailed Description

CA Line Edit Widget.

Version

Revision:

#13

Date

DateTime:

2010/09/06 13:16:04

Author

andrew.rhyder

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

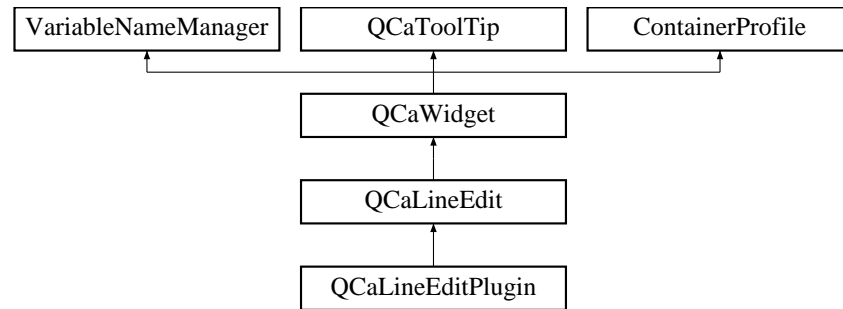
- `epicsqt/ca_framework/widgets/include/QCaLineEdit.h`

12.46 QCaLineEditPlugin Class Reference

CA Line Edit Widget Plugin.

```
#include <QCaLineEditPlugin.h>
```

Inheritance diagram for QCaLineEditPlugin:



Public Types

- enum **Formats** {
Default = QCaStringFormatting::FORMAT_DEFAULT, **Floating** = QCaStringFormatting::FORMAT_FLOATING, **Integer** = QCaStringFormatting::FORMAT_INTEGER, **UnsignedInteger** = QCaStringFormatting::FORMAT_UNSIGNEDINTEGER,
Time = QCaStringFormatting::FORMAT_TIME }
- enum **Notations** { **Fixed** = QCaStringFormatting::NOTATION_FIXED, **Scientific** = QCaStringFormatting::NOTATION_SCIENTIFIC, **Automatic** = QCaStringFormatting::NOTATION_AUTOMATIC }

Public Member Functions

- [QCaLineEditPlugin](#) (QWidget *parent=0)
Constructors.
- **QCaLineEditPlugin** (QString variableName, QWidget *parent=0)
- void **setVariableNameProperty** (QString variableName)
- QString **getVariableNameProperty** ()
- void **setVariableNameSubstitutionsProperty** (QString variableNameSubstitutions)
- QString **getVariableNameSubstitutionsProperty** ()
- void **setFormatProperty** (Formats format)
- Formats **getFormatProperty** ()
- void **setNotationProperty** (Notations notation)
- Notations **getNotationProperty** ()

Properties

- QString **variable**
- QString **variableSubstitutions**
- bool **writeOnLoseFocus**
String formatting options.
- bool **writeOnEnter**
Write changed value to database when widget object loses focus (user moves from widget)
- bool **subscribe**
Set the number of variables that will stream data updates to the widget. Default of 1 if not called.
- bool **variableAsToolTip**
Flag if data updates should be requested.
- bool **enabled**
- bool **confirmWrite**
Override the default widget setEnabled to allow alarm states to override current enabled state.
- unsigned **int**
String formatting properties.
- bool **useDbPrecision**
- bool **leadingZero**
- bool **trailingZeros**
- bool **addUnits**
- Formats **format**
- Notations **notation**

12.46.1 Detailed Description

CA Line Edit Widget Plugin.

Version

Revision:

#9

Date

DateTime:

2010/09/06 11:58:56

Author

andrew.rhyder

12.46.2 Property Documentation**12.46.2.1 QString QCaLineEditPlugin::variable** [read, write]

Note, a property macro in the form 'Q_PROPERTY(QString variableName READ ...)' doesn't work. A property name ending with 'Name' results in some sort of string a variable being displayed, but will only accept alphanumeric and won't generate call-backs on change.

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

- epicsqt/ca_framework/plugins/include/QCaLineEditPlugin.h

12.47 QCaLineEditPluginManager Class Reference

CA Line Edit Widget Plugin Manager for designer.

```
#include <QCaLineEditPluginManager.h>
```

Public Member Functions

- **QCaLineEditPluginManager** (QObject *parent=0)
- bool **isContainer** () const
- bool **isInitialized** () const
- QIcon **icon** () const
- QString **group** () const
- QString **includeFile** () const
- QString **name** () const
- QString **toolTip** () const
- QString **whatsThis** () const
- QWidget * **createWidget** (QWidget *parent)
- void **initialize** (QDesignerFormEditorInterface *core)

12.47.1 Detailed Description

CA Line Edit Widget Plugin Manager for designer.

Version

Revision:

#4

Date**DateTime:**

2010/06/23 07:49:40

Author

andrew.rhyder

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

- epicsqt/ca_framework/plugins/include/QCaLineEditPluginManager.h

12.48 QCaObject Class Reference

Provides channel access to QT.

```
#include <QCaObject.h>
```

12.48.1 Detailed Description

Provides channel access to QT.

Version**Revision:**

#10

Date**DateTime:**

2010/09/06 13:16:04

Author

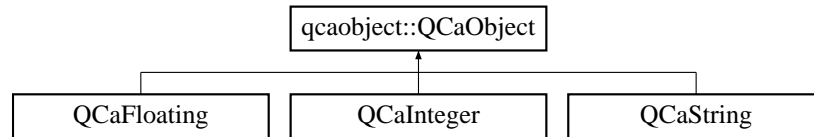
anthony.owen

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

- epicsqt/ca_framework/data/include/QCaObject.h

12.49 qcaobject::QCaObject Class Reference

Inheritance diagram for qcaobject::QCaObject:



Public Slots

- bool **writeData** (const QVariant &value)
- void **resendLastData** ()

Signals

- void **dataChanged** (const QVariant &value, [QCaAlarmInfo](#) &alarmInfo, [QCaDateTime](#) &timeStamp)
- void **connectionChanged** ([QCaConnectionInfo](#) &connectionInfo)

Public Member Functions

- **QCaObject** (const QString &recordName, QObject *eventObject)
- **QCaObject** (const QString &recordName, QObject *eventObject, [UserMessage](#) *userMessageIn)
- bool **subscribe** ()
- bool **singleShotRead** ()
- bool **dataTypeKnown** ()
- bool **createChannel** ()

State machine access functions.

- void **deleteChannel** ()
- bool **createSubscription** ()
- bool **getChannel** ()
- bool **putChannel** ()
- bool **isChannelConnected** ()
- void **startConnectionTimer** ()
- void **stopConnectionTimer** ()
- void **setUserMessage** ([UserMessage](#) *userMessageIn)
- void **enableWriteCallbacks** (bool enable)
- bool **isWriteCallbacksEnabled** ()
- QString **getEgu** ()
- QStringList **getEnumerations** ()
- unsigned int **getPrecision** ()

- double **getDisplayLimitUpper** ()
- double **getDisplayLimitLower** ()
- double **getAlarmLimitUpper** ()
- double **getAlarmLimitLower** ()
- double **getWarningLimitUpper** ()
- double **getWarningLimitLower** ()
- double **getControlLimitUpper** ()
- double **getControlLimitLower** ()

Static Public Member Functions

- static void **processEventStatic** ([QCaEventUpdate](#) *dataUpdateEvent)

Protected Member Functions

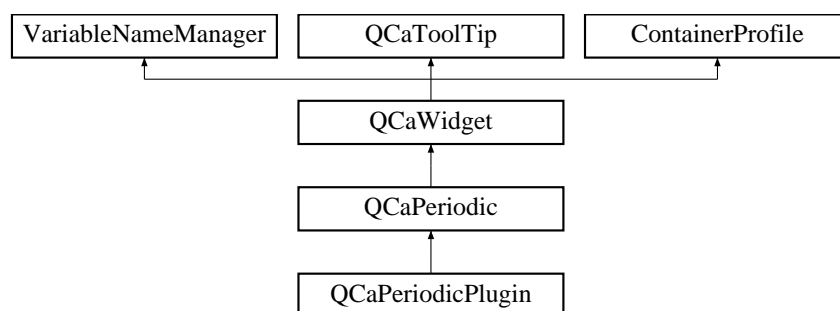
- generic::generic_types **getDataType** ()

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

- [epicsqt/ca_framework/data/include/QCaObject.h](#)

12.50 QCaPeriodic Class Reference

Inheritance diagram for QCaPeriodic:



Classes

- struct [elementInfoStruct](#)
- struct [userInfoStructArray](#)

Public Types

- enum **variableTypes** {
VARIABLE_TYPE_NUMBER, **VARIABLE_TYPE_ATOMIC_WEIGHT**, **VARIABLE_TYPE_MELTING_POINT**, **VARIABLE_TYPE_BOILING_POINT**,
VARIABLE_TYPE_DENSITY, **VARIABLE_TYPE_GROUP**, **VARIABLE_TYPE_IONIZATION_ENERGY**, **VARIABLE_TYPE_USER_VALUE_1**,
VARIABLE_TYPE_USER_VALUE_2 }
- enum **presentationOptions** { **PRESENTATION_BUTTON_AND_LABEL**, **PRESENTATION_BUTTON_ONLY**, **PRESENTATION_LABEL_ONLY** }

Public Slots

- void **requestEnabled** (const bool &state)

Signals

- void **dbValueChanged** (const double &out)
- void **dbElementChanged** (const QString &out)
- void **requestResend** ()

Public Member Functions

- **QCaPeriodic** (QWidget *parent=0)
- **QCaPeriodic** (const QString &variableName, QWidget *parent=0)
- bool **isEnabled** () const
- void **setEnabled** (const bool &state)
- void **setVariableNameAndSubstitutions** (QString variableNameIn, QString variableNameSubstitutionsIn, unsigned int variableIndex)
- void **setSubscribe** (bool [subscribe](#))
- bool **getSubscribe** ()
- void **setVariableAsToolTip** (bool [variableAsToolTip](#))
- bool **getVariableAsToolTip** ()
- void **setPresentationOption** (presentationOptions presentationOptionIn)
- presentationOptions **getPresentationOption** ()
- void **setVariableType1** (variableTypes variableType1In)
- variableTypes **getVariableType1** ()
- void **setVariableType2** (variableTypes variableType2In)
- variableTypes **getVariableType2** ()
- void **setVariableTolerance1** (double variableTolerance1In)
- double **getVariableTolerance1** ()
- void **setVariableTolerance2** (double variableTolerance2In)
- double **getVariableTolerance2** ()
- void **setUserInfo** (QString userInfo)
- QString **getUserInfo** ()

Public Attributes

- [userInfoStruct](#) **userInfo** [NUM_ELEMENTS]

Static Public Attributes

- static [elementInfoStruct](#) **elementInfo** [NUM_ELEMENTS]

Protected Member Functions

- void [establishConnection](#) (unsigned int variableIndex)
Initialise macro substitutions. May be used on creation.

Protected Attributes

- [QCaFloatingFormatting](#) **floatingFormatting**
- bool **localEnabled**
- variableTypes **variableType1**
- variableTypes **variableType2**
- double **variableTolerance1**
- double **variableTolerance2**

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

- epicsqt/ca_framework/widgets/include/QCaPeriodic.h

12.51 QCaPeriodicComponentData Class Reference

Public Attributes

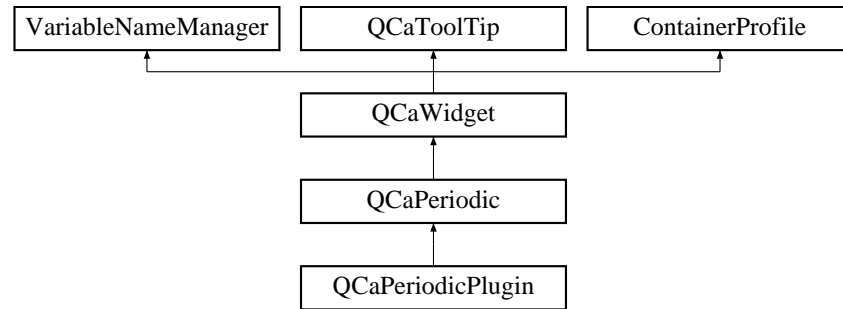
- unsigned int **variableIndex1**
- double **lastData1**
- bool **haveLastData1**
- unsigned int **variableIndex2**
- double **lastData2**
- bool **haveLastData2**

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

- epicsqt/ca_framework/widgets/include/QCaPeriodic.h

12.52 QCaPeriodicPlugin Class Reference

Inheritance diagram for QCaPeriodicPlugin:



Public Types

- enum **PresentationOptions** { **buttonAndLabel** = QCaPeriodic::PRESENTATION_BUTTON_AND_LABEL, **buttonOnly** = QCaPeriodic::PRESENTATION_BUTTON_ONLY, **labelOnly** = QCaPeriodic::PRESENTATION_LABEL_ONLY }
- enum **VariableTypes** {
Number = QCaPeriodic::VARIABLE_TYPE_NUMBER, **atomicWeight** = QCaPeriodic::VARIABLE_TYPE_ATOMIC_WEIGHT, **meltingPoint** = QCaPeriodic::VARIABLE_TYPE_MELTING_POINT, **boilingPoint** = QCaPeriodic::VARIABLE_TYPE_BOILING_POINT,
density = QCaPeriodic::VARIABLE_TYPE_DENSITY, **group** = QCaPeriodic::VARIABLE_TYPE_GROUP, **ionizationEnergy** = QCaPeriodic::VARIABLE_TYPE_IONIZATION_ENERGY, **userValue1** = QCaPeriodic::VARIABLE_TYPE_USER_VALUE_1,
userValue2 = QCaPeriodic::VARIABLE_TYPE_USER_VALUE_2 }

Public Member Functions

- [QCaPeriodicPlugin](#) (QWidget *parent=0)
Constructors.
- **QCaPeriodicPlugin** (QString variableName, QWidget *parent=0)
- void **setVariableName1Property** (QString variable1Name)
- QString **getVariableName1Property** ()
- void **setVariableNameSubstitutions1Property** (QString variableNameSubstitutions1)
- QString **getVariableNameSubstitutions1Property** ()
- void **setVariableName2Property** (QString variable2Name)
- QString **getVariableName2Property** ()
- void **setVariableNameSubstitutions2Property** (QString variableNameSubstitutions2)

- QString **getVariableNameSubstitutions2Property** ()
- void **setVariableName3Property** (QString variable3Name)
- QString **getVariableName3Property** ()
- void **setVariableNameSubstitutions3Property** (QString variableNameSubstitutions3)
- QString **getVariableNameSubstitutions3Property** ()
- void **setVariableName4Property** (QString variable4Name)
- QString **getVariableName4Property** ()
- void **setVariableNameSubstitutions4Property** (QString variableNameSubstitutions4)
- QString **getVariableNameSubstitutions4Property** ()
- void **setPresentationOptionProperty** (PresentationOptions presentationOption)
- PresentationOptions **getPresentationOptionProperty** ()
- void **setVariableType1Property** (VariableTypes variableType)
- void **setVariableType2Property** (VariableTypes variableType)
- VariableTypes **getVariableType1Property** ()
- VariableTypes **getVariableType2Property** ()

Properties

- QString **writeButtonVariable1**
- QString **writeButtonVariableSubstitutions1**
- QString **writeButtonVariable2**
- QString **writeButtonVariableSubstitutions2**
- QString **readbackLabelVariable1**
- QString **readbackLabelVariableSubstitutions1**
- QString **readbackLabelVariable2**
- QString **readbackLabelVariableSubstitutions2**
- bool **subscribe**
Set the number of variables that will stream data updates to the widget. Default of 1 if not called.
- bool **variableAsToolTip**
Flag if data updates should be requested.
- bool **enabled**
- PresentationOptions **presentationOption**
- VariableTypes **variableType1**
- VariableTypes **variableType2**
- double **variableTolerance1**
- double **variableTolerance2**
- QString **userInfo**

12.52.1 Property Documentation

12.52.1.1 QString QCaPeriodicPlugin::writeButtonVariable1 [read, write]

Note, a property macro in the form 'Q_PROPERTY(QString variableName READ ...)' doesn't work. A property name ending with 'Name' results in some sort of string a variable being displayed, but will only accept alphanumeric and won't generate callbacks on change.

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

- epicsqt/ca_framework/plugins/include/QCaPeriodicPlugin.h

12.53 QCaPeriodicPluginManager Class Reference

```
#include <QCaPeriodicPluginManager.h>
```

Public Member Functions

- **QCaPeriodicPluginManager** (QObject *parent=0)
- bool **isContainer** () const
- bool **isInitialized** () const
- QIcon **icon** () const
- QString **group** () const
- QString **includeFile** () const
- QString **name** () const
- QString **toolTip** () const
- QString **whatsThis** () const
- QWidget * **createWidget** (QWidget *parent)
- void **initialize** (QDesignerFormEditorInterface *core)

12.53.1 Detailed Description

???

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

- epicsqt/ca_framework/plugins/include/QCaPeriodicPluginManager.h

12.54 QCaPeriodicPluginTaskMenu Class Reference

Public Member Functions

- **QCaPeriodicPluginTaskMenu** ([QCaPeriodicPlugin](#) *periodicPlugin, QObject *parent)

- QAction * **preferredEditAction** () const
- QList< QAction * > **taskActions** () const

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

- epicsqt/ca_framework/plugins/include/QCaPeriodicPluginTaskMenu.h

12.55 QCaPeriodicPluginTaskMenuFactory Class Reference

Public Member Functions

- **QCaPeriodicPluginTaskMenuFactory** (QExtensionManager *parent=0)

Protected Member Functions

- QObject * **createExtension** (QObject *object, const QString &iid, QObject *parent)
const

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

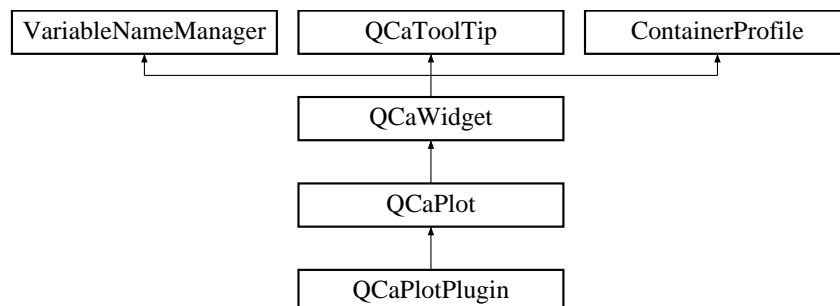
- epicsqt/ca_framework/plugins/include/QCaPeriodicPluginTaskMenu.h

12.56 QCaPlot Class Reference

Manage updating tool tip with variable name, alarm state and connected state.

```
#include <QCaPlot.h>
```

Inheritance diagram for QCaPlot:



Public Slots

- void **requestEnabled** (const bool &state)

Signals

- void **dbValueChanged** (const double &out)
- void **dbValueChanged** (const QVector< double > &out)

Public Member Functions

- **QCaPlot** (QWidget *parent=0)
- **QCaPlot** (const QString &variableName, QWidget *parent=0)
- bool **isEnabled** () const
- void **setEnabled** (bool state)
- void **setVariableNameAndSubstitutions** (QString variableNameIn, QString variableNameSubstitutionsIn, unsigned int variableIndex)
- void **setSubscribe** (bool [subscribe](#))
- bool **getSubscribe** ()
- void **setVariableAsToolTip** (bool [variableAsToolTip](#))
- bool **getVariableAsToolTip** ()
- void **setYMin** (double yMin)
- double **getYMin** ()
- void **setYMax** (double yMax)
- double **getYMax** ()
- void **setAutoScale** (bool autoScale)
- bool **getAutoScale** ()
- void **setAxisEnableX** (bool axisEnableXIn)
- bool **getAxisEnableX** ()
- void **setAxisEnableY** (bool axisEnableYIn)
- bool **getAxisEnableY** ()
- QString **getTitle** ()
- void **setBackgroundColor** (QColor backgroundColor)
- QColor **getBackgroundColor** ()
- void **setTraceStyle** (QwtPlotCurve::CurveStyle traceStyle, const unsigned int variableIndex)
- QwtPlotCurve::CurveStyle **getTraceStyle** (const unsigned int variableIndex)
- void **setTraceColor** (QColor traceColor, const unsigned int variableIndex)
- void **setTraceColor1** (QColor traceColor)
- void **setTraceColor2** (QColor traceColor)
- void **setTraceColor3** (QColor traceColor)
- void **setTraceColor4** (QColor traceColor)
- QColor **getTraceColor** (const unsigned int variableIndex)
- QColor **getTraceColor1** ()
- QColor **getTraceColor2** ()
- QColor **getTraceColor3** ()
- QColor **getTraceColor4** ()
- void **setTraceLegend1** (QString traceLegend)
- void **setTraceLegend2** (QString traceLegend)
- void **setTraceLegend3** (QString traceLegend)

- void **setTraceLegend4** (QString traceLegend)
- QString **getTraceLegend1** ()
- QString **getTraceLegend2** ()
- QString **getTraceLegend3** ()
- QString **getTraceLegend4** ()
- void **setXUnit** (QString xUnit)
- QString **getXUnit** ()
- void **setYUnit** (QString yUnit)
- QString **getYUnit** ()
- void **setGridEnableMajorX** (bool gridEnableMajorXIn)
- void **setGridEnableMajorY** (bool gridEnableMajorYIn)
- void **setGridEnableMinorX** (bool gridEnableMinorXIn)
- void **setGridEnableMinorY** (bool gridEnableMinorYIn)
- bool **getGridEnableMajorX** ()
- bool **getGridEnableMajorY** ()
- bool **getGridEnableMinorX** ()
- bool **getGridEnableMinorY** ()
- void **setGridMajorColor** (QColor gridMajorColorIn)
- void **setGridMinorColor** (QColor gridMinorColorIn)
- QColor **getGridMajorColor** ()
- QColor **getGridMinorColor** ()
- void **setXStart** (double xStart)
- double **getXStart** ()
- void **setXIncrement** (double xIncrement)
- double **getXIncrement** ()
- void **setTimeSpan** (unsigned int timeSpan)
- unsigned int **getTimeSpan** ()
- void **setTickRate** (unsigned int tickRate)
- unsigned int **getTickRate** ()

Protected Member Functions

- void [establishConnection](#) (unsigned int variableIndex)
Initialise macro substitutions. May be used on creation.
- void **setRunVisible** (bool visibleIn)

Protected Attributes

- [QCaFloatingFormatting](#) **floatingFormatting**
- bool **localEnabled**
- bool **visible**

12.56.1 Detailed Description

Manage updating tool tip with variable name, alarm state and connected state.

Version

Revision:

#4

Date

DateTime:

2010/09/06 13:16:04

Author

glenn.jackson

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

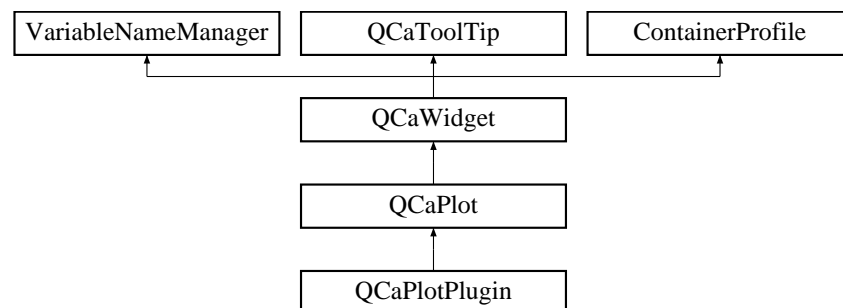
- epicsqt/ca_framework/widgets/include/QCaPlot.h

12.57 QCaPlotPlugin Class Reference

CA Plot Widget Plugin for designer.

```
#include <QCaPlotPlugin.h>
```

Inheritance diagram for QCaPlotPlugin:



Public Types

- enum **TraceStyles** { **Lines** = QwtPlotCurve::Lines, **Sticks** = QwtPlotCurve::Sticks, **Steps** = QwtPlotCurve::Steps, **Dots** = QwtPlotCurve::Dots }

Public Member Functions

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

Constructors.

- **QCaPlotPlugin** (QString variableName, QWidget *parent=0)
- void **setVariableName1Property** (QString variable1Name)
- QString **getVariableName1Property** ()
- void **setVariableName2Property** (QString variable2Name)
- QString **getVariableName2Property** ()
- void **setVariableName3Property** (QString variable3Name)
- QString **getVariableName3Property** ()
- void **setVariableName4Property** (QString variable4Name)
- QString **getVariableName4Property** ()
- void **setVariableNameSubstitutionsProperty** (QString variableNameSubstitutions)
- QString **getVariableNameSubstitutionsProperty** ()
- void **setTraceStyle1** (TraceStyles traceStyle)
- void **setTraceStyle2** (TraceStyles traceStyle)
- void **setTraceStyle3** (TraceStyles traceStyle)
- void **setTraceStyle4** (TraceStyles traceStyle)
- TraceStyles **getTraceStyle1** ()
- TraceStyles **getTraceStyle2** ()
- TraceStyles **getTraceStyle3** ()
- TraceStyles **getTraceStyle4** ()

Properties

- QString **variable1**
- QString **variable2**
- QString **variable3**
- QString **variable4**
- QString **variableSubstitutions**
- bool [variableAsToolTip](#)

Flag if data updates should be requested.

- QColor **traceColor1**
- QColor **traceColor2**
- QColor **traceColor3**
- QColor **traceColor4**
- TraceStyles **traceStyle1**
- TraceStyles **traceStyle2**
- TraceStyles **traceStyle3**
- TraceStyles **traceStyle4**
- QString **traceLegend1**
- QString **traceLegend2**

- QString **traceLegend3**
- QString **traceLegend4**
- double **yMin**
- double **yMax**
- bool **autoScale**
- bool **axisEnableX**
- bool **axisEnableY**
- bool **gridEnableMajorX**
- bool **gridEnableMajorY**
- bool **gridEnableMinorX**
- bool **gridEnableMinorY**
- QColor **gridMajorColor**
- QColor **gridMinorColor**
- QString **title**
- QColor **backgroundColor**
- QString **xUnit**
- QString **yUnit**
- double **xStart**
- double **xIncrement**
- unsigned **int**

12.57.1 Detailed Description

CA Plot Widget Plugin for designer.

Version

Revision:

#3

Date

DateTime:

2010/09/06 11:58:56

Author

glenn.jackson

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

- epicsqt/ca_framework/plugins/include/QCaPlotPlugin.h

12.58 QCaPlotPluginManager Class Reference

CA Plot Widget Plugin Manager for designer.

```
#include <QCaPlotPluginManager.h>
```

Public Member Functions

- **QCaPlotPluginManager** (QObject *parent=0)
- bool **isContainer** () const
- bool **isInitialized** () const
- QIcon **icon** () const
- QString **group** () const
- QString **includeFile** () const
- QString **name** () const
- QString **toolTip** () const
- QString **whatsThis** () const
- QWidget * **createWidget** (QWidget *parent)
- void **initialize** (QDesignerFormEditorInterface *core)

12.58.1 Detailed Description

CA Plot Widget Plugin Manager for designer. ???

Version

Revision:

#1

Date

DateTime:

2010/08/30 16:37:08

Author

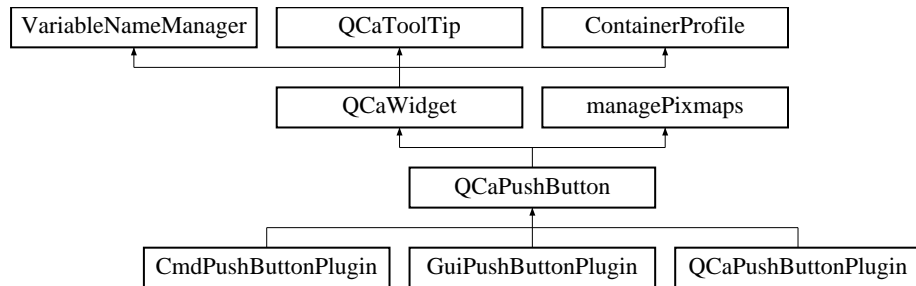
glenn.jackson

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

- epicsqt/ca_framework/plugins/include/QCaPlotPluginManager.h

12.59 QCaPushButton Class Reference

Inheritance diagram for QCaPushButton:



Public Types

- enum **updateOptions** { **UPDATE_TEXT**, **UPDATE_ICON**, **UPDATE_TEXT_AND_ICON** }

Public Slots

- void **requestEnabled** (const bool &state)
- void **onGeneralMessage** (QString message)

Signals

- void **dbValueChanged** (const QString &out)
- void **newGui** (QString [guiName](#), QString substitutions, ASguiForm::creationOptions [creationOption](#))

Public Member Functions

- **QCaPushButton** (QWidget *parent=0)
- **QCaPushButton** (const QString &variableName, QWidget *parent=0)
- bool **isEnabled** () const
- void **setEnabled** (const bool &state)
- void **setVariableNameAndSubstitutions** (QString variableNameIn, QString variableNameSubstitutionsIn, unsigned int variableIndex)
- void **setSubscribe** (bool [subscribe](#))
- bool **getSubscribe** ()
- void **setVariableAsToolTip** (bool [variableAsToolTip](#))
- bool **getVariableAsToolTip** ()
- void **setUpdateOption** (updateOptions updateOptionIn)
- updateOptions **getUpdateOption** ()

- void **setPrecision** (unsigned int precision)
- unsigned int **getPrecision** ()
- void **setUseDbPrecision** (bool useDbPrecision)
- bool **getUseDbPrecision** ()
- void **setLeadingZero** (bool leadingZero)
- bool **getLeadingZero** ()
- void **setTrailingZeros** (bool trailingZeros)
- bool **getTrailingZeros** ()
- void **setAddUnits** (bool addUnits)
- bool **getAddUnits** ()
- void **setFormat** (QCaStringFormatting::formats format)
- QCaStringFormatting::formats **getFormat** ()
- void **setRadix** (unsigned int radix)
- unsigned int **getRadix** ()
- void **setNotation** (QCaStringFormatting::notations notation)
- QCaStringFormatting::notations **getNotation** ()
- void **setWriteOnPress** (bool writeOnPress)
- bool **getWriteOnPress** ()
- void **setWriteOnRelease** (bool writeOnRelease)
- bool **getWriteOnRelease** ()
- void **setWriteOnClick** (bool writeOnClick)
- bool **getWriteOnClick** ()
- void **setPressText** (QString [pressText](#))
- QString **getPressText** ()
- void **setReleaseText** (QString releaseTextIn)
- QString **getReleaseText** ()
- void **setClickText** (QString clickTextIn)
- QString **getClickText** ()
- void **setProgram** (QString [program](#))
- QString **getProgram** ()
- void **setArguments** (QStringList [arguments](#))
- QStringList **getArguments** ()
- void **setGuiName** (QString [guiName](#))
- QString **getGuiName** ()
- void **setCreationOption** (ASguiForm::creationOptions [creationOption](#))
- ASguiForm::creationOptions **getCreationOption** ()

Protected Member Functions

- void [establishConnection](#) (unsigned int variableIndex)
Initialise macro substitutions. May be used on creation.

Protected Attributes

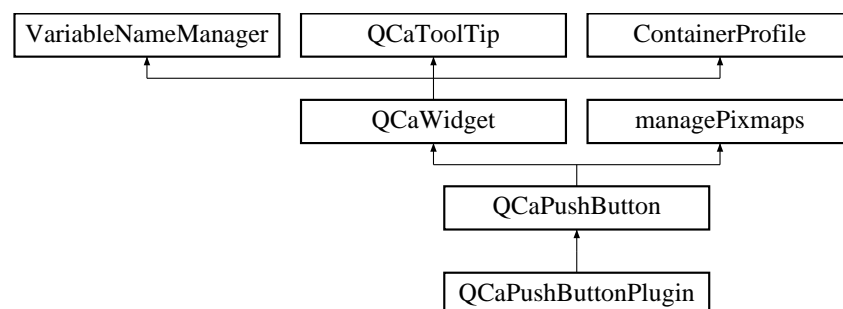
- [QCaStringFormatting](#) **stringFormatting**
- bool **writeOnPress**
- bool **writeOnRelease**
- bool **writeOnClick**
- QString **releaseText**
- QString **pressText**
Text to write on a button release.
- QString **clickText**
Text to write on a button press.
- QString **program**
Text to write on a button click.
- QStringList **arguments**
Program to run.
- QString **guiName**
Program arguments.
- ASguiForm::creationOptions **creationOption**
GUI file name to launch.
- bool **localEnabled**
- updateOptions **updateOption**

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

- `epicsqt/ca_framework/widgets/include/QCaPushButton.h`

12.60 QCaPushButtonPlugin Class Reference

Inheritance diagram for QCaPushButtonPlugin:



Public Types

- enum **UpdateOptions** { **Text** = QCaPushButton::UPDATE_TEXT, **Icon** = QCaPushButton::UPDATE_ICON, **TextAndIcon** = QCaPushButton::UPDATE_TEXT_AND_ICON }
- enum **Formats** {
Default = QCaStringFormatting::FORMAT_DEFAULT, **Floating** = QCaStringFormatting::FORMAT_FLOATING, **Integer** = QCaStringFormatting::FORMAT_INTEGER, **UnsignedInteger** = QCaStringFormatting::FORMAT_UNSIGNEDINTEGER,
Time = QCaStringFormatting::FORMAT_TIME }
- enum **Notations** { **Fixed** = QCaStringFormatting::NOTATION_FIXED, **Scientific** = QCaStringFormatting::NOTATION_SCIENTIFIC, **Automatic** = QCaStringFormatting::NOTATION_AUTOMATIC }
- enum **CreationOptionNames** { **Open** = ASguiForm::CREATION_OPTION_OPEN, **NewTab** = ASguiForm::CREATION_OPTION_NEW_TAB, **NewWindow** = ASguiForm::CREATION_OPTION_NEW_WINDOW }

Public Member Functions

- [QCaPushButtonPlugin](#) (QWidget *parent=0)
Constructors.
- **QCaPushButtonPlugin** (QString variableName, QWidget *parent=0)
- void **setVariableNameProperty** (QString variableName)
- QString **getVariableNameProperty** ()
- void **setVariableNameSubstitutionsProperty** (QString variableNameSubstitutions)
- QString **getVariableNameSubstitutionsProperty** ()
- void **setUpdateOptionProperty** (UpdateOptions updateOption)
- UpdateOptions **getUpdateOptionProperty** ()
- void **setPixmap0Property** (QPixmap pixmap)
- void **setPixmap1Property** (QPixmap pixmap)
- void **setPixmap2Property** (QPixmap pixmap)
- void **setPixmap3Property** (QPixmap pixmap)
- void **setPixmap4Property** (QPixmap pixmap)
- void **setPixmap5Property** (QPixmap pixmap)
- void **setPixmap6Property** (QPixmap pixmap)
- void **setPixmap7Property** (QPixmap pixmap)
- QPixmap **getPixmap0Property** ()
- QPixmap **getPixmap1Property** ()
- QPixmap **getPixmap2Property** ()
- QPixmap **getPixmap3Property** ()
- QPixmap **getPixmap4Property** ()
- QPixmap **getPixmap5Property** ()
- QPixmap **getPixmap6Property** ()
- QPixmap **getPixmap7Property** ()
- void **setFormatProperty** (Formats format)

- Formats **getFormatProperty** ()
- void **setNotationProperty** (Notations notation)
- Notations **getNotationProperty** ()
- void **setCreationOptionProperty** (CreationOptionNames creationOptionIn)
- CreationOptionNames **getCreationOptionProperty** ()

Properties

- QString **variable**
- QString **variableSubstitutions**
- bool **subscribe**

Set the number of variables that will stream data updates to the widget. Default of 1 if not called.

- bool **variableAsToolTip**

Flag if data updates should be requested.

- bool **enabled**
- UpdateOptions **updateOption**

Update options (text, pixmap, or both)

- QPixmap **pixmap0**

Pixmaps.

- QPixmap **pixmap1**
- QPixmap **pixmap2**
- QPixmap **pixmap3**
- QPixmap **pixmap4**
- QPixmap **pixmap5**
- QPixmap **pixmap6**
- QPixmap **pixmap7**
- unsigned **int**

String formatting properties.

- bool **useDbPrecision**
- bool **leadingZero**
- bool **trailingZeros**
- bool **addUnits**
- Formats **format**
- Notations **notation**
- bool **writeOnPress**
- bool **writeOnRelease**
- bool **writeOnClick**
- QString **pressText**

Text to write on a button release.

- QString **releaseText**
- QString [clickText](#)
Text to write on a button press.
- QString [program](#)
Text to write on a button click.
- QStringList [arguments](#)
Program to run.
- QString **guiFile**
- CreationOptionNames [creationOption](#)
GUI file name to launch.

12.60.1 Property Documentation

12.60.1.1 QString QCaPushButtonPlugin::variable [read, write]

Note, a property macro in the form 'Q_PROPERTY(QString variableName READ ...)' doesn't work. A property name ending with 'Name' results in some sort of string a variable being displayed, but will only accept alphanumeric and won't generate call-backs on change.

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

- `epicsqt/ca_framework/plugins/include/QCaPushButtonPlugin.h`

12.61 QCaPushButtonPluginManager Class Reference

CA Push Button Widget Plugin Manager for designer.

```
#include <QCaPushButtonPluginManager.h>
```

Public Member Functions

- **QCaPushButtonPluginManager** (QObject *parent=0)
- bool **isContainer** () const
- bool **isInitialized** () const
- QIcon **icon** () const
- QString **group** () const
- QString **includeFile** () const
- QString **name** () const
- QString **toolTip** () const
- QString **whatsThis** () const
- QWidget * **createWidget** (QWidget *parent)
- void **initialize** (QDesignerFormEditorInterface *core)

12.61.1 Detailed Description

CA Push Button Widget Plugin Manager for designer. ???

Version

Revision:

#4

Date

DateTime:

2010/06/23 07:49:40

Author

andrew.rhyder

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

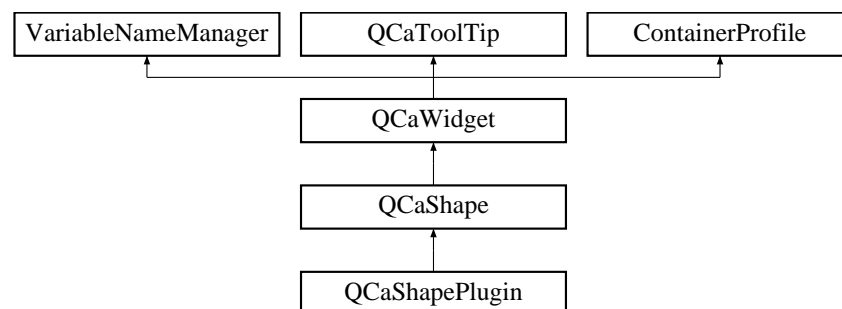
- `epicsqt/ca_framework/plugins/include/QCaPushButtonPluginManager.h`

12.62 QCaShape Class Reference

CA Shape Widget.

```
#include <QCaShape.h>
```

Inheritance diagram for QCaShape:



Public Types

- enum **Shape** {

- Line, Points, Polyline, Polygon,**
- Rect, RoundedRect, Ellipse, Arc,**
- Chord, Pie, Path, Text,**
- Pixmap }**
- enum **Animations** {
- Width, Height, X, Y,**
- Transperency, Rotation, ColourHue, ColourSaturation,**
- ColourValue, ColourIndex, Penwidth }**

Public Slots

- void **requestEnabled** (const bool &state)

Signals

- void **dbValueChanged1** (const qlonglong &out)
- void **dbValueChanged2** (const qlonglong &out)
- void **dbValueChanged3** (const qlonglong &out)
- void **dbValueChanged4** (const qlonglong &out)
- void **dbValueChanged5** (const qlonglong &out)
- void **dbValueChanged6** (const qlonglong &out)

Public Member Functions

- **QCaShape** (QWidget *parent=0)
- **QCaShape** (const QString &variableName, QWidget *parent=0)
- bool **isEnabled** () const
- void **setEnabled** (bool state)
- void **setVariableNameAndSubstitutions** (QString variableNameIn, QString variableNameSubstitutionsIn, unsigned int variableIndex)
- void **setAnimation** (Animations animation, const int index)
- Animations **getAnimation** (const int index)
- void **setScale** (const double scale, const int index)
- double **getScale** (const int index)
- void **setOffset** (const double offset, const int index)
- double **getOffset** (const int index)
- void **setBorder** (const bool border)
- bool **getBorder** ()
- void **setFill** (const bool fill)
- bool **getFill** ()
- void **setSubscribe** (const bool [subscribe](#))
- bool **getSubscribe** ()
- void **setVariableAsToolTip** (const bool [variableAsToolTip](#))
- bool **getVariableAsToolTip** ()

- void **setShape** (Shape shape)
- Shape **getShape** ()
- void **setNumPoints** (const unsigned int numPoints)
- unsigned int **getNumPoints** ()
- void **setOriginTranslation** (const QPoint originTranslation)
- QPoint **getOriginTranslation** ()
- void **setPoint** (const QPoint point, const int index)
- QPoint **getPoint** (const int index)
- void **setColor** (const QColor color, const int index)
- QColor **getColor** (const int index)
- void **setDrawBorder** (const bool drawBorder)
- bool **getDrawBorder** ()
- void **setLineWidth** (const unsigned int lineWidth)
- unsigned int **getLineWidth** ()
- void **setStartAngle** (const double startAngle)
- double **getStartAngle** ()
- void **setRotation** (const double rotation)
- double **getRotation** ()
- void **setArcLength** (const double arcLength)
- double **getArcLength** ()
- void **setText** (const QString text)
- QString **getText** ()

Protected Member Functions

- void **colorChange** (unsigned int index)
 - void **establishConnection** (unsigned int variableIndex)
- Index into colorsProperty last used when setting brush color.*

Protected Attributes

- [QCaIntegerFormatting](#) **integerFormatting**
- double **offsets** [OFFSETS_SIZE]
- double **scales** [SCALES_SIZE]
- Shape **shape**
- QPoint **originTranslation**
- QPoint **points** [POINTS_SIZE]
- unsigned int **numPoints**
- QColor **colors** [COLORS_SIZE]
- Animations **animations** [6]
- bool **fill**
- bool **border**
- double **startAngle**
- double **arcLength**

- QString **text**
- double **rotation**
- unsigned int **lineWidth**
- bool **drawBorder**
- bool **localEnabled**
- unsigned int **currentColor**

Act on a color property change. (will update shape if using the color)

12.62.1 Detailed Description

CA Shape Widget.

Version

Revision:

#12

Date

DateTime:

2010/09/06 13:16:04

Author

andrew.rhyder

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

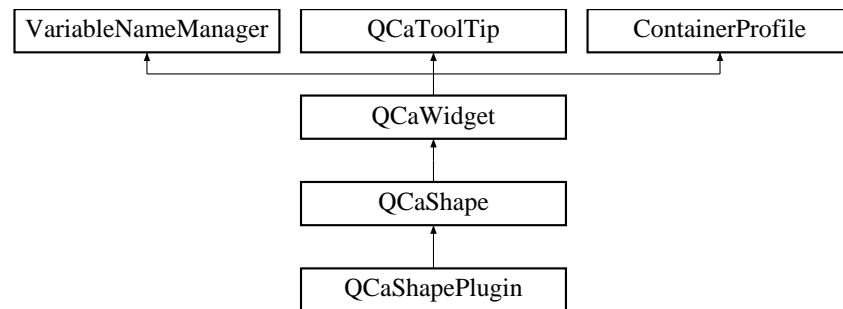
- epicsqt/ca_framework/widgets/include/QCaShape.h

12.63 QCaShapePlugin Class Reference

CA Shape Widget Plugin.

```
#include <QCaShapePlugin.h>
```

Inheritance diagram for QCaShapePlugin:



Public Types

- enum **animationOptions** {
Width = QCaShape::Width, **Height** = QCaShape::Height, **X** = QCaShape::X, **Y** = QCaShape::Y,
Transperency = QCaShape::Transperency, **Rotation** = QCaShape::Rotation, **ColourHue** = QCaShape::ColourHue, **ColourSaturation** = QCaShape::ColourSaturation,
ColourValue = QCaShape::ColourValue, **ColourIndex** = QCaShape::ColourIndex,
Penwidth = QCaShape::Penwidth }
- enum **shapeOptions** {
Line = QCaShape::Line, **Points** = QCaShape::Points, **Polyline** = QCaShape::Polyline,
Polygon = QCaShape::Polygon,
Rect = QCaShape::Rect, **RoundedRect** = QCaShape::RoundedRect, **Ellipse** = QCaShape::Ellipse, **Arc** = QCaShape::Arc,
Chord = QCaShape::Chord, **Pie** = QCaShape::Pie, **Path** = QCaShape::Path,
Text = QCaShape::Text,
Pixmap = QCaShape::Pixmap }

Public Member Functions

- [QCaShapePlugin](#) (QWidget *parent=0)
Constructors.
- QCaShapePlugin** (QString variableName, QWidget *parent=0)
- void **setVariableName1Property** (QString variable1Name)
- QString **getVariableName1Property** ()
- void **setVariableName2Property** (QString variable2Name)
- QString **getVariableName2Property** ()
- void **setVariableName3Property** (QString variable3Name)
- QString **getVariableName3Property** ()
- void **setVariableName4Property** (QString variable4Name)
- QString **getVariableName4Property** ()
- void **setVariableName5Property** (QString variable5Name)

- **QString** **getVariableName5Property** ()
- **void** **setVariableName6Property** (QString variable6Name)
- **QString** **getVariableName6Property** ()
- **void** **setAnimation1Property** (animationOptions animation)
- **animationOptions** **getAnimation1Property** ()
- **void** **setAnimation2Property** (animationOptions animation)
- **animationOptions** **getAnimation2Property** ()
- **void** **setAnimation3Property** (animationOptions animation)
- **animationOptions** **getAnimation3Property** ()
- **void** **setAnimation4Property** (animationOptions animation)
- **animationOptions** **getAnimation4Property** ()
- **void** **setAnimation5Property** (animationOptions animation)
- **animationOptions** **getAnimation5Property** ()
- **void** **setAnimation6Property** (animationOptions animation)
- **animationOptions** **getAnimation6Property** ()
- **void** **setScale1Property** (double scale)
- **double** **getScale1Property** ()
- **void** **setScale2Property** (double scale)
- **double** **getScale2Property** ()
- **void** **setScale3Property** (double scale)
- **double** **getScale3Property** ()
- **void** **setScale4Property** (double scale)
- **double** **getScale4Property** ()
- **void** **setScale5Property** (double scale)
- **double** **getScale5Property** ()
- **void** **setScale6Property** (double scale)
- **double** **getScale6Property** ()
- **void** **setOffset1Property** (double offset)
- **double** **getOffset1Property** ()
- **void** **setOffset2Property** (double offset)
- **double** **getOffset2Property** ()
- **void** **setOffset3Property** (double offset)
- **double** **getOffset3Property** ()
- **void** **setOffset4Property** (double offset)
- **double** **getOffset4Property** ()
- **void** **setOffset5Property** (double offset)
- **double** **getOffset5Property** ()
- **void** **setOffset6Property** (double offset)
- **double** **getOffset6Property** ()
- **void** **setVariableNameSubstitutionsProperty** (QString variableNameSubstitutions)
- **QString** **getVariableNameSubstitutionsProperty** ()
- **void** **setShapeProperty** (shapeOptions shape)
- **shapeOptions** **getShapeProperty** ()
- **void** **setPoint1Property** (QPoint point)
- **QPoint** **getPoint1Property** ()

- void **setPoint2Property** (QPoint point)
- QPoint **getPoint2Property** ()
- void **setPoint3Property** (QPoint point)
- QPoint **getPoint3Property** ()
- void **setPoint4Property** (QPoint point)
- QPoint **getPoint4Property** ()
- void **setPoint5Property** (QPoint point)
- QPoint **getPoint5Property** ()
- void **setPoint6Property** (QPoint point)
- QPoint **getPoint6Property** ()
- void **setPoint7Property** (QPoint point)
- QPoint **getPoint7Property** ()
- void **setPoint8Property** (QPoint point)
- QPoint **getPoint8Property** ()
- void **setPoint9Property** (QPoint point)
- QPoint **getPoint9Property** ()
- void **setPoint10Property** (QPoint point)
- QPoint **getPoint10Property** ()
- void **setColor1Property** (QColor color)
- QColor **getColor1Property** ()
- void **setColor2Property** (QColor color)
- QColor **getColor2Property** ()
- void **setColor3Property** (QColor color)
- QColor **getColor3Property** ()
- void **setColor4Property** (QColor color)
- QColor **getColor4Property** ()
- void **setColor5Property** (QColor color)
- QColor **getColor5Property** ()
- void **setColor6Property** (QColor color)
- QColor **getColor6Property** ()
- void **setColor7Property** (QColor color)
- QColor **getColor7Property** ()
- void **setColor8Property** (QColor color)
- QColor **getColor8Property** ()
- void **setColor9Property** (QColor color)
- QColor **getColor9Property** ()
- void **setColor10Property** (QColor color)
- QColor **getColor10Property** ()

Properties

- QString [variable1](#)
- QString **variable2**
- QString **variable3**
- QString **variable4**
- QString **variable5**

- QString **variable6**
- animationOptions **animation1**
- animationOptions **animation2**
- animationOptions **animation3**
- animationOptions **animation4**
- animationOptions **animation5**
- animationOptions **animation6**
- double **scale1**
- double **scale2**
- double **scale3**
- double **scale4**
- double **scale5**
- double **scale6**
- double **offset1**
- double **offset2**
- double **offset3**
- double **offset4**
- double **offset5**
- double **offset6**
- bool **border**
- bool **fill**
- QString **variableSubstitutions**
- bool [subscribe](#)

Set the number of variables that will stream data updates to the widget. Default of 1 if not called.

- bool [variableAsToolTip](#)

Flag if data updates should be requested.

- bool **enabled**
- shapeOptions **shape**
- unsigned **int**
- QPoint **originTranslation**
- QPoint **point1**
- QPoint **point2**
- QPoint **point3**
- QPoint **point4**
- QPoint **point5**
- QPoint **point6**
- QPoint **point7**
- QPoint **point8**
- QPoint **point9**
- QPoint **point10**
- QColor **color1**
- QColor **color2**
- QColor **color3**

- QColor **color4**
- QColor **color5**
- QColor **color6**
- QColor **color7**
- QColor **color8**
- QColor **color9**
- QColor **color10**
- bool **drawBorder**
- double **startAngle**
- double **rotation**
- double **arcLength**
- QString **text**

12.63.1 Detailed Description

CA Shape Widget Plugin.

Version

Revision:

#7

Date

DateTime:

2010/09/06 11:58:56

Author

andrew.rhyder

12.63.2 Property Documentation

12.63.2.1 QString QCaShapePlugin::variable1 [read, write]

Note, a property macro in the form 'Q_PROPERTY(QString variableName READ ...)' doesn't work. A property name ending with 'Name' results in some sort of string a variable being displayed, but will only accept alphanumeric and won't generate callbacks on change.

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

- epicsqt/ca_framework/plugins/include/QCaShapePlugin.h

12.64 QCaShapePluginManager Class Reference

CA Shape Widget Plugin Manager for designer.

```
#include <QCaShapePluginManager.h>
```

Public Member Functions

- **QCaShapePluginManager** (QObject *parent=0)
- bool **isContainer** () const
- bool **isInitialized** () const
- QIcon **icon** () const
- QString **group** () const
- QString **includeFile** () const
- QString **name** () const
- QString **toolTip** () const
- QString **whatsThis** () const
- QWidget * **createWidget** (QWidget *parent)
- void **initialize** (QDesignerFormEditorInterface *core)

12.64.1 Detailed Description

CA Shape Widget Plugin Manager for designer.

Version

Revision:

#4

Date

DateTime:

2010/06/23 07:49:40

Author

andrew.rhyder

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

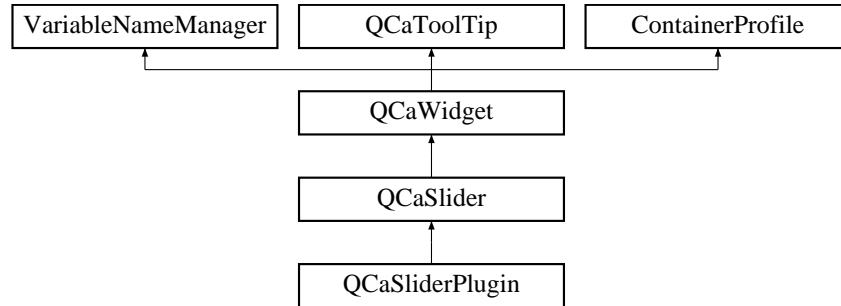
- epicsqt/ca_framework/plugins/include/QCaShapePluginManager.h

12.65 QCaSlider Class Reference

CA Slider Widget.

```
#include <QCaSlider.h>
```

Inheritance diagram for QCaSlider:



Public Slots

- void **requestEnabled** (const bool &state)

Signals

- void **dbValueChanged** (const qlonglong &out)

Public Member Functions

- **QCaSlider** (QWidget *parent=0)
- **QCaSlider** (const QString &variableName, QWidget *parent=0)
- bool **isEnabled** () const
- void **setEnabled** (bool state)
- void **setVariableNameAndSubstitutions** (QString variableNameIn, QString variableNameSubstitutionsIn, unsigned int variableIndex)
- void **setWriteOnChange** (bool [writeOnChange](#))
- bool **getWriteOnChange** ()
- void **setSubscribe** (bool [subscribe](#))
- bool **getSubscribe** ()
- void **setVariableAsToolTip** (bool [variableAsToolTip](#))
- bool **getVariableAsToolTip** ()

Protected Member Functions

- void [establishConnection](#) (unsigned int variableIndex)
Initialise macro substitutions. May be used on creation.

Protected Attributes

- [QCaIntegerFormatting](#) **integerFormatting**
- bool [writeOnChange](#)

Integer formatting options.

- bool [localEnabled](#)

Write changed value to database when ever the position changes.

12.65.1 Detailed Description

CA Slider Widget.

Version

Revision:

#12

Date

DateTime:

2010/09/06 13:16:04

Author

andrew.rhyder

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

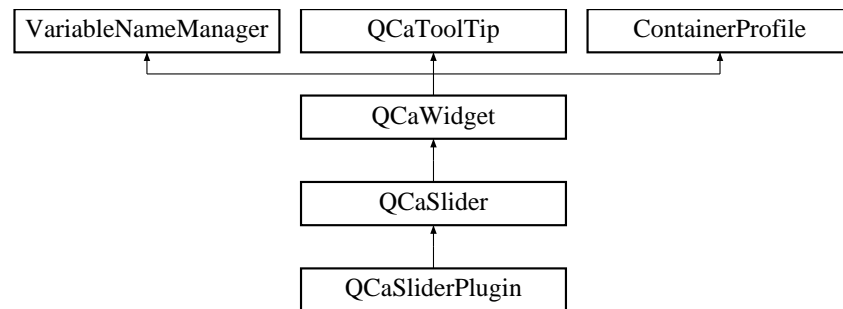
- `epicsqt/ca_framework/widgets/include/QCaSlider.h`

12.66 QCaSliderPlugin Class Reference

CA Slider Widget Plugin.

```
#include <QCaSliderPlugin.h>
```

Inheritance diagram for QCaSliderPlugin:



Public Member Functions

- [QCaSliderPlugin](#) (QWidget *parent=0)
Constructors.
- [QCaSliderPlugin](#) (QString variableName, QWidget *parent=0)
- void [setVariableNameProperty](#) (QString variableName)
- QString [getVariableNameProperty](#) ()
- void [setVariableNameSubstitutionsProperty](#) (QString variableNameSubstitutions)
- QString [getVariableNameSubstitutionsProperty](#) ()

Properties

- QString [variable](#)
- QString [variableSubstitutions](#)
- bool [enabled](#)
- bool [writeOnChange](#)
Integer formatting options.
- bool [subscribe](#)
Set the number of variables that will stream data updates to the widget. Default of 1 if not called.
- bool [variableAsToolTip](#)
Flag if data updates should be requested.

12.66.1 Detailed Description

CA Slider Widget Plugin.

Version

Revision:

#6

Date**DateTime:**

2010/09/06 11:58:56

Author

andrew.rhyder

12.66.2 Property Documentation**12.66.2.1 QString QCaSliderPlugin::variable** [read, write]

Note, a property macro in the form 'Q_PROPERTY(QString variableName READ ...)' doesn't work. A property name ending with 'Name' results in some sort of string a variable being displayed, but will only accept alphanumeric and won't generate call-backs on change.

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

- epicsqt/ca_framework/plugins/include/QCaSliderPlugin.h

12.67 QCaSliderPluginManager Class Reference

CA Slider Widget Plugin Manager for designer.

```
#include <QCaSliderPluginManager.h>
```

Public Member Functions

- **QCaSliderPluginManager** (QObject *parent=0)
- bool **isContainer** () const
- bool **isInitialized** () const
- QIcon **icon** () const
- QString **group** () const
- QString **includeFile** () const
- QString **name** () const
- QString **toolTip** () const
- QString **whatsThis** () const
- QWidget * **createWidget** (QWidget *parent)
- void **initialize** (QDesignerFormEditorInterface *core)

12.67.1 Detailed Description

CA Slider Widget Plugin Manager for designer.

Version

Revision:

#4

Date

DateTime:

2010/06/23 07:49:40

Author

andrew.rhyder

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

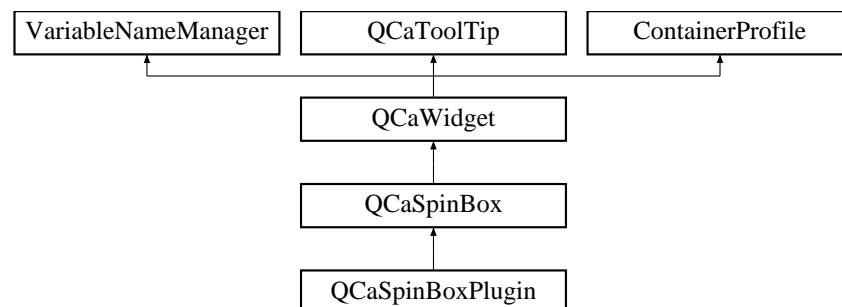
- `epicsqt/ca_framework/plugins/include/QCaSliderPluginManager.h`

12.68 QCaSpinBox Class Reference

CA Spinbox Widget.

```
#include <QCaSpinBox.h>
```

Inheritance diagram for QCaSpinBox:



Public Slots

- `void requestEnabled (const bool &state)`

Signals

- void **dbValueChanged** (const qulonglong &out)

Public Member Functions

- **QCaSpinBox** (QWidget *parent=0)
- **QCaSpinBox** (const QString &variableName, QWidget *parent=0)
- bool **isEnabled** () const
- void **setEnabled** (bool state)
- void **setVariableNameAndSubstitutions** (QString variableNameIn, QString variableNameSubstitutionsIn, unsigned int variableIndex)
- void **setSubscribe** (bool [subscribe](#))
- bool **getSubscribe** ()
- void **setVariableAsToolTip** (bool [variableAsToolTip](#))
- bool **getVariableAsToolTip** ()

Protected Member Functions

- void [establishConnection](#) (unsigned int variableIndex)
Initialise macro substitutions. May be used on creation.

Protected Attributes

- [QCaIntegerFormatting](#) **integerFormatting**
- bool **localEnabled**

12.68.1 Detailed Description

CA Spinbox Widget.

Version

Revision:

#12

Date

DateTime:

2010/09/06 13:16:04

Author

andrew.rhyder

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

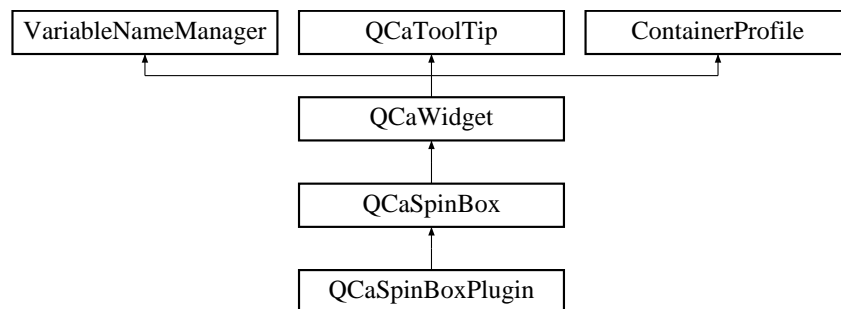
- epicsqt/ca_framework/widgets/include/QCaSpinBox.h

12.69 QCaSpinBoxPlugin Class Reference

CA Spinbox Widget Plugin.

```
#include <QCaSpinBoxPlugin.h>
```

Inheritance diagram for QCaSpinBoxPlugin:

**Public Member Functions**

- **QCaSpinBoxPlugin** (QWidget *parent=0)
- **QCaSpinBoxPlugin** (QString variableName, QWidget *parent=0)
- void **setVariableNameProperty** (QString variableName)
- QString **getVariableNameProperty** ()
- void **setVariableNameSubstitutionsProperty** (QString variableNameSubstitutions)
- QString **getVariableNameSubstitutionsProperty** ()

Properties

- QString **variable**
- QString **variableSubstitutions**
- bool **subscribe**

Set the number of variables that will stream data updates to the widget. Default of 1 if not called.

- bool **variableAsToolTip**

Flag if data updates should be requested.

- bool **enabled**

12.69.1 Detailed Description

CA Spinbox Widget Plugin.

Version

Revision:

#6

Date

DateTime:

2010/09/06 11:58:56

Author

andrew.rhyder

12.69.2 Property Documentation

12.69.2.1 QString QCaSpinBoxPlugin::variable [read, write]

Note, a property macro in the form 'Q_PROPERTY(QString variableName READ ...)' doesn't work. A property name ending with 'Name' results in some sort of string a variable being displayed, but will only accept alphanumeric and won't generate call-backs on change.

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

- epicsqt/ca_framework/plugins/include/QCaSpinBoxPlugin.h

12.70 QCaSpinBoxPluginManager Class Reference

CA Spinbox Widget Plugin Manager for designer.

```
#include <QCaSpinBoxPluginManager.h>
```

Public Member Functions

- **QCaSpinBoxPluginManager** (QObject *parent=0)

- bool **isContainer** () const
- bool **isInitialized** () const
- QIcon **icon** () const
- QString **group** () const
- QString **includeFile** () const
- QString **name** () const
- QString **toolTip** () const
- QString **whatsThis** () const
- QWidget * **createWidget** (QWidget *parent)
- void **initialize** (QDesignerFormEditorInterface *core)

12.70.1 Detailed Description

CA Spinbox Widget Plugin Manager for designer.

Version

Revision:

#4

Date

DateTime:

2010/06/23 07:49:40

Author

andrew.rhyder

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

- epicsqt/ca_framework/plugins/include/QCaSpinBoxPluginManager.h

12.71 QCaStateMachine Class Reference

Statemachine architecture.

```
#include <QCaStateMachine.h>
```

12.71.1 Detailed Description

Statemachine architecture.

Version

Revision:

#7

Date

DateTime:

2010/06/23 07:49:40

Author

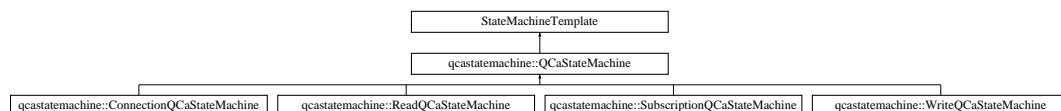
anthony.owen

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

- epicsqt/ca_framework/data/include/QCaStateMachine.h

12.72 qcastatemachine::QCaStateMachine Class Reference

Inheritance diagram for qcastatemachine::QCaStateMachine:



Public Member Functions

- **QCaStateMachine** (void *parent)
- virtual bool **process** (int requestedState)=0

Public Attributes

- QMutex **lock**
- bool **pending**
- bool **active**
- bool **expired**

- void * **myWorker**

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

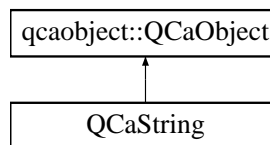
- epicsqt/ca_framework/data/include/QCaStateMachine.h

12.73 QCaString Class Reference

String wrapper for [QCaObject](#) variant data.

```
#include <QCaString.h>
```

Inheritance diagram for QCaString:



Public Slots

- void **writeString** (const QString &data)

Signals

- void **stringChanged** (const QString &value, [QCaAlarmInfo](#) &alarmInfo, [QCaDateTime](#) &timeStamp, const unsigned int &variableIndex)

Public Member Functions

- **QCaString** (QString recordName, QObject *eventObject, [QCaStringFormatting](#) *stringFormattingIn, unsigned int variableIndexIn)
- **QCaString** (QString recordName, QObject *eventObject, [QCaStringFormatting](#) *stringFormattingIn, unsigned int variableIndexIn, [UserMessage](#) *userMessageIn)

12.73.1 Detailed Description

String wrapper for [QCaObject](#) variant data.

Version

Revision:

#6

Date**DateTime:**

2010/09/06 13:16:04

Author

andrew.rhyder

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

- epicsqt/ca_framework/data/include/QCaString.h

12.74 QCaStringFormatting Class Reference

Formats the string for [QCaString](#) data.

```
#include <QCaStringFormatting.h>
```

Public Types

- enum **formats** {
FORMAT_DEFAULT, **FORMAT_FLOATING**, **FORMAT_INTEGER**, **FORMAT_**-
UNSIGNEDINTEGER,
FORMAT_TIME, **FORMAT_LOCAL_ENUMERATE** }
- enum **notations** { **NOTATION_FIXED** = QTextStream::FixedNotation, **NOTATION_**-
SCIENTIFIC = QTextStream::ScientificNotation, **NOTATION_AUTOMATIC**
= QTextStream::SmartNotation }
- enum **arrayActions** { **APPEND**, **ASCII**, **INDEX** }

Public Member Functions

- QString **formatString** (const QVariant &value)
- QVariant **formatValue** (const QString &text)
- void **setDbEgu** (QString egu)
- void **setDbEnumerations** (QStringList enumerations)
Units to be added (or removed) from the formatted string if 'addUnits' flag is set.
- void **setDbPrecision** (unsigned int dbPrecisionIn)
- void **setDbVariableIsStatField** (bool isStatField)
- void **setPrecision** (unsigned int precision)

- void **setUseDbPrecision** (bool useDbPrecision)
- void **setLeadingZero** (bool leadingZero)
- void **setTrailingZeros** (bool trailingZeros)
- void **setFormat** (formats format)
- void **setRadix** (unsigned int radix)
- void **setNotation** (notations notation)
- void **setArrayAction** (arrayActions arrayActionIn)
- void **setArrayIndex** (unsigned int arrayIndexIn)
- void **setAddUnits** (bool addUnits)
- void **setLocalEnumeration** (QString localEnumerationIn)
- unsigned int **getPrecision** ()
- bool **getUseDbPrecision** ()
- bool **getLeadingZero** ()
- bool **getTrailingZeros** ()
- formats **getFormat** ()
- unsigned int **getRadix** ()
- notations **getNotation** ()
- arrayActions **getArrayAction** ()
- unsigned int **getArrayIndex** ()
- bool **getAddUnits** ()
- QString **getLocalEnumeration** ()

12.74.1 Detailed Description

Formats the string for [QCaString](#) data.

Version

Revision:

#9

Date

DateTime:

2010/06/23 07:49:40

Author

andrew.rhyder

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

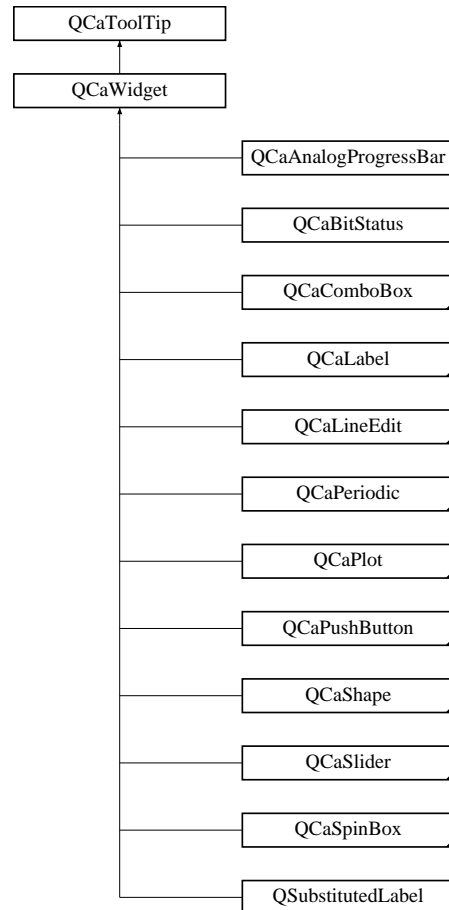
- epicsqt/ca_framework/data/include/QCaStringFormatting.h

12.75 QCaToolTip Class Reference

Manage updating tool tip with variable name, alarm state and connected state.

```
#include <QCaToolTip.h>
```

Inheritance diagram for QCaToolTip:



Public Member Functions

- void **updateToolTipVariable** (const QString &variable)
- void **updateToolTipAlarm** (const QString &alarm)
- void **updateToolTipConnection** (bool connection)
- virtual void **updateToolTip** (const QString &tip)=0

12.75.1 Detailed Description

Manage updating tool tip with variable name, alarm state and connected state.

Version**Revision:**

#5

Date**DateTime:**

2010/06/23 07:49:40

Author

andrew.rhyder

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

- epicsqt/ca_framework/widgets/include/QCaToolTip.h

12.76 QCaVariableNamePropertyManager Class Reference

Handles variable name sets.

```
#include <QCaVariableNamePropertyManager.h>
```

Signals

- void **newVariableNameProperty** (QString variable, QString Substitutions, unsigned int variableIndex)

Public Member Functions

- QString **getVariableNameProperty** ()
- void **setVariableNameProperty** (QString variableNamePropertyIn)
- QString **getSubstitutionsProperty** ()
- void **setSubstitutionsProperty** (QString substitutionsPropertyIn)
- void **setVariableIndex** (unsigned int variableIndexIn)

12.76.1 Detailed Description

Handles variable name sets.

Version

Revision:

#3

Date**DateTime:**

2010/06/23 07:49:40

Author

andrew.rhyder

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

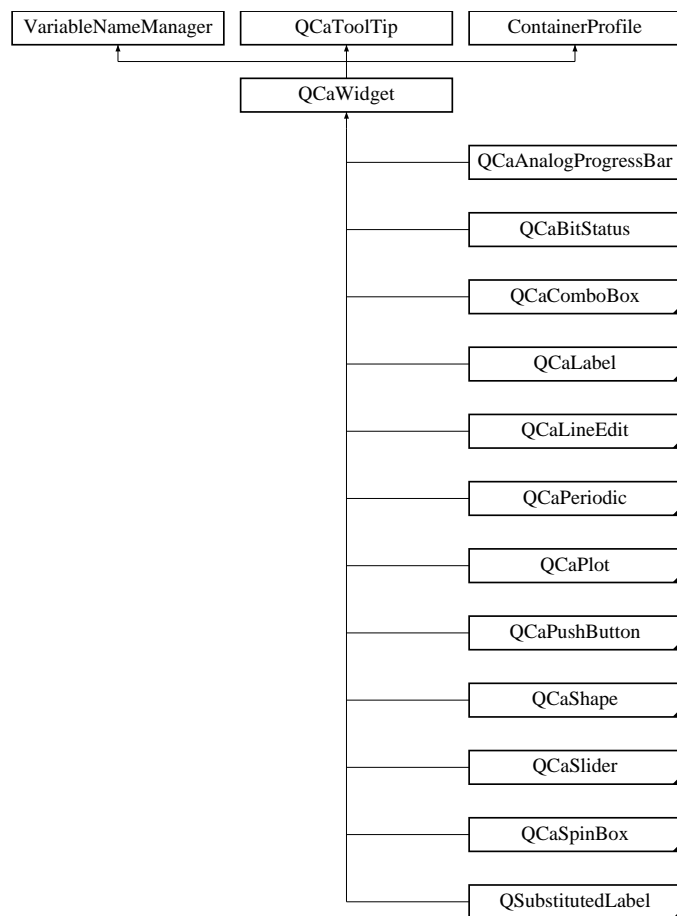
- `epicsqt/ca_framework/data/include/QCaVariableNamePropertyManager.h`

12.77 QCaWidget Class Reference

Template for Qt-CA aware widgets.

```
#include <QCaWidget.h>
```

Inheritance diagram for QCaWidget:



Public Member Functions

- [~QCaWidget \(\)](#)

Constructor.

- void [activate \(\)](#)

Manager of messages to the user.

Public Attributes

- [UserMessage userMessage](#)

Destructor.

Protected Member Functions

- void [setNumVariables](#) (unsigned int numVariablesIn)
Initiate updates.
- [qcaobject::QCaObject](#) * [createConnection](#) (unsigned int variableIndex)
Flag the tool tip should be set to the variable name.
- virtual void [setup](#) ()=0
Create a CA connection. Return a [QCaObject](#) if successfull.
- virtual [qcaobject::QCaObject](#) * [createQcaItem](#) (unsigned int variableIndex)=0
???
- [qcaobject::QCaObject](#) * [getQcaItem](#) (unsigned int variableIndex)
Function to create a appropriate superclass of [QCaObject](#) to stream data updates.

Protected Attributes

- bool [subscribe](#)
Set the number of variables that will stream data updates to the widget. Default of 1 if not called.
- bool [variableAsToolTip](#)
Flag if data updates should be requested.

12.77.1 Detailed Description

Template for Qt-CA aware widgets.

Version

Revision:

#10

Date

DateTime:

2010/09/06 13:16:04

Author

anthony.owen

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

- `epicsqt/ca_framework/data/include/QCaWidget.h`

12.78 QCaWidgets Class Reference

Public Member Functions

- **QCaWidgets** (QObject *parent=0)
- virtual QList< QDesignerCustomWidgetInterface * > **customWidgets** () const

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

- `epicsqt/ca_framework/plugins/include/QCaDesignerPlugin.h`

12.79 QEpicsPV Class Reference

Public Slots

- const QVariant & **set** (QVariant value, int delay=-1)
Sets new value for the field.
- void **setPV** (const QString &_pvName="")
Sets new PV.

Signals

- void **connectionChanged** (bool connected)
- void **connected** ()
Emitted on successful connection.
- void **disconnected** ()
Emitted on disconnection, destruction and in `::setPV()` method.
- void **valueChanged** (const QVariant &value)
- void **valueUpdated** (const QVariant &value)
- void **valueInitd** (const QVariant &value)
Emitted on first value update after the connection was established.

Public Member Functions

- [QEpicsPV](#) (const QString &_pvName, QObject *parent=0)
- [QEpicsPV](#) (QObject *parent=0)
- [~QEpicsPV](#) ()

Destructor.

- const QVariant & [get](#) () const
- void [needUpdated](#) () const

Starts monitoring for the updates.

- const QVariant & [getUpdated](#) (int delay=defaultDelay) const

Returns an updated value of the field.

- bool [isConnected](#) () const
- const QStringList & [getEnum](#) () const
- const QString & [pv](#) () const
- const QVariant & [getReady](#) (int delay=defaultDelay) const

Ensures that the object is ready to operate.

Static Public Member Functions

- static void [setDebugLevel](#) (unsigned level=0)
- static QVariant [get](#) (const QString &_pvName, int delay=defaultDelay)
- static QVariant [set](#) (QString &_pvName, const QVariant &value, int delay=-1)

Static version of the ::getUpdated() method.

Static version of the ::set() method.

12.79.1 Constructor & Destructor Documentation

12.79.1.1 QEpicsPV::QEpicsPV (const QString & _pvName, QObject * parent = 0)

Constructor.

Parameters

<i>_pvName</i>	new PV name.
<i>parent</i>	Object's parent.

12.79.1.2 QEpicsPV::QEpicsPV (QObject * parent = 0)

Constructor.

Parameters

<i>parent</i>	Object's parent.
---------------	------------------

12.79.2 Member Function Documentation**12.79.2.1 void QEpicsPV::connectionChanged (bool *connected*) [signal]**

Emitted whenever the connection status has changed.

Parameters

<i>connected</i>	new connection status.
------------------	------------------------

12.79.2.2 static QVariant QEpicsPV::get (const QString & *_pvName*, int *delay* = defaultDelay) [static]

Static version of the ::getUpdated() method.

Constructs new object of the [QEpicsPV](#) type, gets it ready and returns the value. Useful for the single shot readouts.

Parameters

<i>_pvName</i>	PV name of the field.
<i>delay</i>	Maximum waiting time in milliseconds. If zero, then unlimited waiting.

Returns

Value of the field. If the value was not updated during the specified delay then returns an invalid object.

12.79.2.3 const QVariant& QEpicsPV::get () const

Returns current value of the field.

Returns

Latest value of the PV field or invalid object if the field is not connected or before first update.

12.79.2.4 const QStringList& QEpicsPV::getEnum () const

PV's enumeration.

Returns

The enumeration of the PV field, or the empty list if the PV is not an enumeration.

12.79.2.5 const QVariant& QEpicsPV::getReady (int *delay* = defaultDelay) const

Ensures that the object is ready to operate.

Connection to the EPICS PV and first update takes certain time after the ::setPV() method or the constructor with name are called. This method waits for the PV to get connected and the value to be initialized before returning.

Parameters

<i>delay</i>	Maximum waiting time in milliseconds. If zero, then unlimited waiting. Note that the actual waiting time can be twice as large as the delay: one delay to wait for the connection and one more to wait for the first update.
--------------	--

Returns

Field's updated value. If the connection could not be established or the value was not updated then returns an invalid data.

12.79.2.6 const QVariant& QEpicsPV::getUpdated (int *delay* = defaultDelay) const

Returns an updated value of the field.

Same as ::get(), but ensures that the PV's value was updated since the last call to ::needUpdated() method or establishing the connection.

Parameters

<i>delay</i>	Maximum waiting time in milliseconds. If zero, then unlimited waiting.
--------------	--

Returns

Updated value of the field. If the value was not updated during the specified delay then returns an invalid object.

12.79.2.7 bool QEpicsPV::isConnected () const

Connection status.

Returns

true if connected, false otherwise.

12.79.2.8 void QEpicsPV::needUpdated () const

Starts monitoring for the updates.

Makes sense only in combination with further call to ::getUpdated(). Ensures that the next call to ::getUpdated() method will return only after a new value of the PV is received.

12.79.2.9 `const QString& QEpicsPV::pv () const`

PV name.

Returns

PV name.

12.79.2.10 `const QVariant& QEpicsPV::set (QVariant value, int delay = -1) [slot]`

Sets new value for the field.

Parameters

<i>value</i>	Value to be set.
<i>delay</i>	Negative delay tells that the confirmation of the update is not required (just puts the new value and returns). If non-negative, then sets the maximum time in milliseconds to wait for the update event before return. Zero delay sets unlimited waiting time. Note that the method does not confirm the updated value to be equal to the one set.

Returns

Last updated value or invalid object if could not confirm the update within the specified delay time.

12.79.2.11 `static QVariant QEpicsPV::set (QString & _pvName, const QVariant & value, int delay = -1) [static]`

Static version of the `::set()` method.

Constructs new object of the [QEpicsPV](#) type, gets it ready, sets the new value and returns the result of the `::set()` method. Useful for the single shot write.

Parameters

<i>_pvName</i>	PV name of the field.
<i>value</i>	Value to be set.
<i>delay</i>	Negative delay tells that the confirmation of the update is not required (just puts the new value and returns). If non-negative, then sets the maximum time in milliseconds to wait for the update event before return. Zero delay sets unlimited waiting time. Note that the method does not confirm the updated value to be equal to the one set.

Returns

Updated value or invalid object if could not confirm the update within the specified delay time.

12.79.2.12 void QEpicsPV::setPV (const QString & *_pvName* = " ") [slot]

Sets new PV.

First disconnects any previous PV (if any was connected) and connects to the new one (if non-empty).

Parameters

<i>_pvName</i>	New PV name.
----------------	--------------

12.79.2.13 void QEpicsPV::valueChanged (const QVariant & *value*) [signal]

Emitted whenever the field is updated with a value different from the old one.

Parameters

<i>value</i>	new value.
--------------	------------

12.79.2.14 void QEpicsPV::valueUpdated (const QVariant & *value*) [signal]

Emitted whenever the field is updated (even if the new value is equal to the old one).

Parameters

<i>value</i>	new value.
--------------	------------

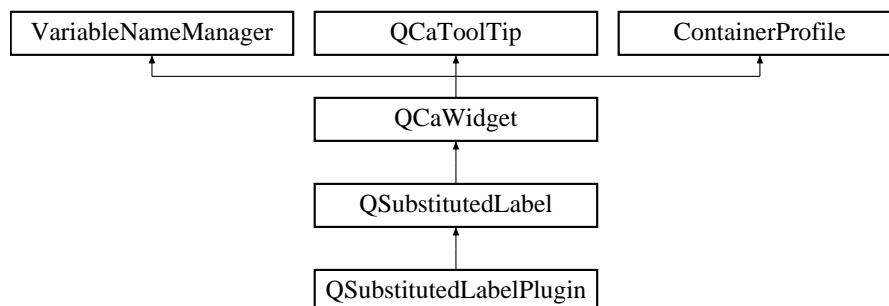
The documentation for this class was generated from the following file:

- epicsqt/ca_framework/data/include/qepicspv.h

12.80 QSubstitutedLabel Class Reference

```
#include <QSubstitutedLabel.h>
```

Inheritance diagram for QSubstitutedLabel:



Public Member Functions

- **QSubstitutedLabel** (QWidget *parent=0)
- void **setVariableNameAndSubstitutions** (QString variableNameIn, QString variableNameSubstitutionsIn, unsigned int variableIndex)
- void **establishConnection** (unsigned int variableIndex)

Initialise macro substitutions. May be used on creation.

12.80.1 Detailed Description

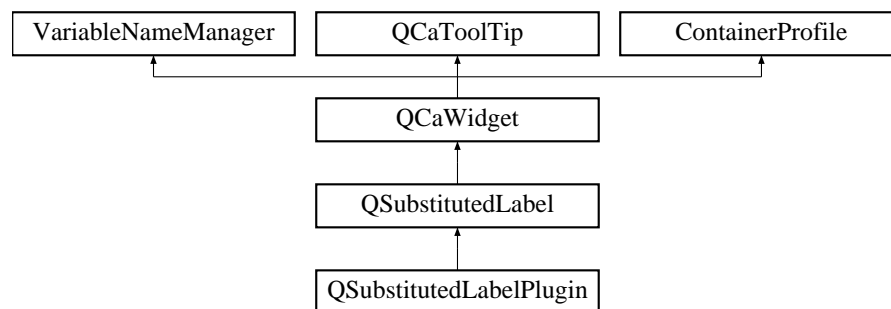
This class is based on the Qt label widget. It is NOT a CA aware widget. It does, however, use the [QCaWidget](#) and its [VariableNameManager](#) base class to manage substituting the label's text in the same way most other widgets manage substitutions in variable names. It is tightly integrated with the base class [QCaWidget](#). Refer to [QCaWidget.cpp](#) for details, but watch out for references to variable names when it is actually just talking about the text that will be displayed in the QLabel widget. This widget is required to implement some virtual functions required by [QCaWidget](#) for managing data connections. These functions are just stubs.

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

- `epicsqt/ca_framework/widgets/include/QSubstitutedLabel.h`

12.81 QSubstitutedLabelPlugin Class Reference

Inheritance diagram for QSubstitutedLabelPlugin:



Public Member Functions

- **QSubstitutedLabelPlugin** (QWidget *parent=0)
Constructors.
- **QSubstitutedLabelPlugin** (QString variableName, QWidget *parent=0)
- void **setVariableNameProperty** (QString variableName)

- QString **getVariableNameProperty** ()
- void **setVariableNameSubstitutionsProperty** (QString variableNameSubstitutions)
- QString **getVariableNameSubstitutionsProperty** ()

Properties

- QString [initialText](#)
- QString [textSubstitutions](#)

12.81.1 Property Documentation

12.81.1.1 QString QSubstitutedLabelPlugin::initialText [read, write]

Text to be substituted. NOTE, this plugin uses the variable-name-and-substitutions mechism used to manage variable names to manage text displayed in the label.

12.81.1.2 QString QSubstitutedLabelPlugin::textSubstitutions [read, write]

Text substitutions. NOTE, this plugin uses the variable-name-and-substitutions mechism used to manage variable names to manage text displayed in the label.

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

- epicsqt/ca_framework/plugins/include/QSubstitutedLabelPlugin.h

12.82 QSubstitutedLabelPluginManager Class Reference

```
#include <QSubstitutedLabelPluginManager.h>
```

Public Member Functions

- **QSubstitutedLabelPluginManager** (QObject *parent=0)
- bool **isContainer** () const
- bool **isInitialized** () const
- QIcon **icon** () const
- QString **group** () const
- QString **includeFile** () const
- QString **name** () const
- QString **toolTip** () const
- QString **whatsThis** () const
- QWidget * **createWidget** (QWidget *parent)
- void **initialize** (QDesignerFormEditorInterface *core)

12.82.1 Detailed Description

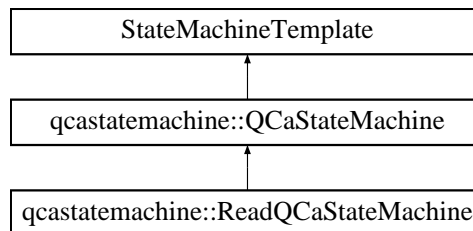
???

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

- epicsqt/ca_framework/plugins/include/QSubstitutedLabelPluginManager.h

12.83 qcastatemachine::ReadQCaStateMachine Class Reference

Inheritance diagram for qcastatemachine::ReadQCaStateMachine:



Public Member Functions

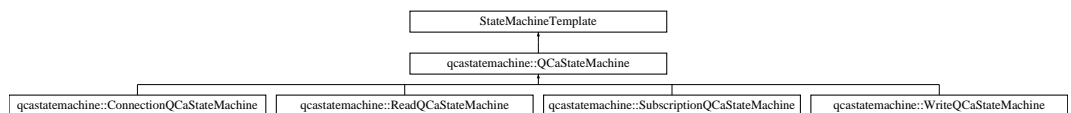
- **ReadQCaStateMachine** (void *parent)
- bool **process** (int requestedState)

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

- epicsqt/ca_framework/data/include/QCaStateMachine.h

12.84 StateMachineTemplate Class Reference

Inheritance diagram for StateMachineTemplate:



Public Member Functions

- virtual bool **process** (int requestedState)=0

Public Attributes

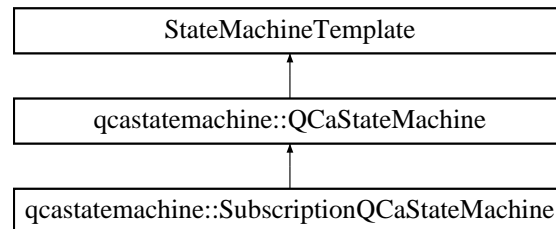
- int **currentState**
- int **requestState**

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

- epicsqt/ca_framework/data/include/QCaStateMachine.h

12.85 qcastatemachine::SubscriptionQCaStateMachine Class Reference

Inheritance diagram for qcastatemachine::SubscriptionQCaStateMachine:



Public Member Functions

- **SubscriptionQCaStateMachine** (void *parent)
- bool **process** (int requestedState)

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

- epicsqt/ca_framework/data/include/QCaStateMachine.h

12.86 trace Class Reference

Public Attributes

- QVector< [QCaDateTime](#) > **timeStamps**
- QVector< double > **xdata**
- QVector< double > **ydata**
- QwtPlotCurve * **curve**
- QColor **color**
- QString **legend**
- bool **waveform**
- QwtPlotCurve::CurveStyle **style**

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

- `epicsqt/ca_framework/widgets/include/QCaPlot.h`

12.87 `userInfoStruct` Class Reference

Public Attributes

- bool **enable**
- double **value1**
- double **value2**
- QString **elementText**

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

- `epicsqt/ca_framework/widgets/include/QCaPeriodic.h`

12.88 `QCaPeriodic::userInfoStructArray` Struct Reference

Public Attributes

- [`userInfoStruct`](#) array [NUM_ELEMENTS]

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

- `epicsqt/ca_framework/widgets/include/QCaPeriodic.h`

12.89 `UserMessage` Class Reference

User message manager.

```
#include <UserMessage.h>
```

Signals

- void **statusMessage** (QString message)
- void **warningMessage** (QString message)
- void **errorMessage** (QString message)
- void **generalMessage** (QString message)

Public Member Functions

- void **setup** (QObject *statusMessageConsumer, QObject *warningMessageConsumer, QObject *errorMessageConsumer)
- void **setup** (QObject *generalMessageConsumer)
- void **sendStatusMessage** (QString message)
- void **sendStatusMessage** (QString message, QString source)
- void **sendWarningMessage** (QString message)
- void **sendWarningMessage** (QString message, QString source)
- void **sendErrorMessage** (QString message)
- void **sendErrorMessage** (QString message, QString source)

12.89.1 Detailed Description

User message manager.

Version

Revision:

#4

Date

DateTime:

2010/06/23 07:49:40

Author

andrew.rhyder

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

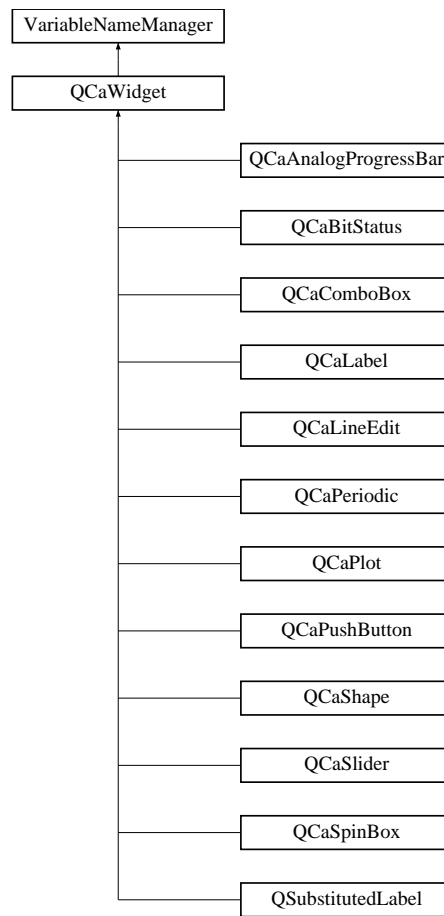
- epicsqt/ca_framework/data/include/UserMessage.h

12.90 VariableNameManager Class Reference

Variable name management.

```
#include <VariableNameManager.h>
```

Inheritance diagram for VariableNameManager:



Public Member Functions

- void **variableNameManagerInitialise** (unsigned int numVariables)
- QString **getOriginalVariableName** (unsigned int variableIndex)
Used to set up the number of variables required (defaults to 1 if this is not called)
- QString **getVariableNameSubstitutions** ()
Return variable name prior to any macro substitutions. (eg, SR\$SECTOR\$V)
- QString **getSubstitutedVariableName** (unsigned int variableIndex)
Return macro substitutions used for variable names (eg, SECTOR=01,V=PRESURE)
- void **initialiseVariableNameSubstitutions** (const QString &substitutions)
Return variable name after macro substitutions.
- virtual void **establishConnection** (unsigned int variableIndex)=0
Initialise macro substitutions. May be used on creation.

- void [setVariableNameSubstitutionsOverride](#) (const QString &substitutions)
Create a CA connection and initiates updates if required.
- void [setVariableName](#) (const QString &variableName, unsigned int variableIndex)
- void [setVariableNameSubstitutions](#) (const QString &substitutions)
Accept a new variable name which may include substitution keys preceeded by \$.

12.90.1 Detailed Description

Variable name management.

Version

Revision:

#7

Date

DateTime:

2010/06/23 07:49:40

Author

andrew.rhyder

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

- `epicsqt/ca_framework/data/include/VariableNameManager.h`

12.91 WidgetRef Class Reference

Public Member Functions

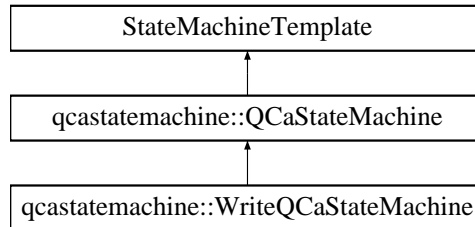
- [WidgetRef](#) ([QCaWidget](#) *refIn)
- [QCaWidget](#) * [getRef](#) ()

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

- `epicsqt/ca_framework/plugins/include/ContainerProfile.h`

12.92 qcastatemachine::WriteQCaStateMachine Class Reference

Inheritance diagram for qcastatemachine::WriteQCaStateMachine:



Public Member Functions

- **WriteQCaStateMachine** (void *parent)
- bool **process** (int requestedState)

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

- epicsqt/ca_framework/data/include/QCaStateMachine.h

Index

- ASguiForm, [33](#)
- ASguiFormPlugin, [35](#)
 - uiFile, [36](#)
- ASguiFormPluginManager, [36](#)
- CmdPushButtonPlugin, [37](#)
 - variableSubstitutions, [38](#)
- CmdPushButtonPluginManager, [38](#)
- connectionChanged
 - QEpicsPV, [138](#)
- ContainerProfile, [39](#)
- get
 - QEpicsPV, [138](#)
- getEnum
 - QEpicsPV, [138](#)
- getReady
 - QEpicsPV, [138](#)
- getUpdated
 - QEpicsPV, [139](#)
- GuiPushButtonPlugin, [42](#)
- GuiPushButtonPluginManager, [43](#)
- initialText
 - QSubstitutedLabelPlugin, [143](#)
- isConnected
 - QEpicsPV, [139](#)
- Link, [44](#)
- LinkPlugin, [46](#)
- LinkPluginManager, [48](#)
- localEnumerationItem, [49](#)
- managePixmaps, [49](#)
- needUpdated
 - QEpicsPV, [139](#)
- PeriodicDialog, [50](#)
- PeriodicElementSetupForm, [50](#)
- PeriodicSetupDialog, [50](#)
- pv
 - QEpicsPV, [139](#)
- QAnalogProgressBar, [50](#)
- QAnalogProgressBarPluginManager, [52](#)
- QBitStatus, [52](#)
- QBitStatusPluginManager, [54](#)
- QCaAlarmInfo, [55](#)
- QCaAnalogProgressBar, [56](#)
 - variable, [57](#)
- QCaAnalogProgressBarPluginManager, [58](#)
- QCaBitStatus, [58](#)
- QCaBitStatusPluginManager, [60](#)
- QCaComboBox, [61](#)
- QCaComboBoxPlugin, [63](#)
- QCaComboBoxPluginManager, [64](#)
- QCaConnectionInfo, [65](#)
- QCaDateTime, [66](#)
- QCaDesignerPlugin, [66](#)
- QCaEventFilter, [67](#)
- QCaEventItem, [68](#)
- QCaEventUpdate, [69](#)
- QCaFloating, [70](#)
- QCaFloatingFormatting, [71](#)
- QCaInstalledFiltersListItem, [72](#)
- QCaInteger, [73](#)
- QCaIntegerFormatting, [74](#)
- QCaLabel, [75](#)
- QCaLabelPlugin, [77](#)
 - variable, [80](#)
- QCaLabelPluginManager, [80](#)
- QCaLineEdit, [81](#)
- QCaLineEditPlugin, [84](#)
 - variable, [86](#)
- QCaLineEditPluginManager, [86](#)
- QCaObject, [87](#)
- qcaobject, [31](#)
- qcaobject::QCaObject, [88](#)
- QCaPeriodic, [89](#)
- QCaPeriodic::elementInfoStruct, [41](#)
- QCaPeriodic::userInfoStructArray, [146](#)
- QCaPeriodicComponentData, [91](#)

- QCaPeriodicPlugin, 92
 - writeButtonVariable1, 94
- QCaPeriodicPluginManager, 94
- QCaPeriodicPluginTaskMenu, 94
- QCaPeriodicPluginTaskMenuFactory, 95
- QCaPlot, 95
- QCaPlotPlugin, 98
- QCaPlotPluginManager, 101
- QCaPushButton, 102
- QCaPushButtonPlugin, 104
 - variable, 107
- QCaPushButtonPluginManager, 107
- QCaShape, 108
- QCaShapePlugin, 111
 - variable1, 116
- QCaShapePluginManager, 117
- QCaSlider, 118
- QCaSliderPlugin, 119
 - variable, 121
- QCaSliderPluginManager, 121
- QCaSpinBox, 122
- QCaSpinBoxPlugin, 124
 - variable, 125
- QCaSpinBoxPluginManager, 125
- QCaStateMachine, 126
- qcastatemachine::ConnectionQCaStateMachine, 39
- qcastatemachine::QCaStateMachine, 127
- qcastatemachine::ReadQCaStateMachine, 144
- qcastatemachine::SubscriptionQCaStateMachine, 145
- qcastatemachine::WriteQCaStateMachine, 150
- QCaString, 128
- QCaStringFormatting, 129
- QCaToolTip, 131
- QCaVariableNamePropertyManager, 132
- QCaWidget, 133
- QCaWidgets, 136
- QEpicsPV, 136
 - connectionChanged, 138
 - get, 138
 - getEnum, 138
 - getReady, 138
 - getUpdated, 139
 - isConnected, 139
 - needUpdated, 139
 - pv, 139
 - QEpicsPV, 137
 - set, 140
 - setPV, 140
 - valueChanged, 141
 - valueUpdated, 141
- QSubstitutedLabel, 141
- QSubstitutedLabelPlugin, 142
 - initialText, 143
 - textSubstitutions, 143
- QSubstitutedLabelPluginManager, 143
- set
 - QEpicsPV, 140
- setPV
 - QEpicsPV, 140
- StateMachineTemplate, 144
- textSubstitutions
 - QSubstitutedLabelPlugin, 143
- trace, 145
- uiFile
 - ASguiFormPlugin, 36
- userInfoStruct, 146
- UserMessage, 146
- valueChanged
 - QEpicsPV, 141
- valueUpdated
 - QEpicsPV, 141
- variable
 - QCaAnalogProgressBar, 57
 - QCaLabelPlugin, 80
 - QCaLineEditPlugin, 86
 - QCaPushButtonPlugin, 107
 - QCaSliderPlugin, 121
 - QCaSpinBoxPlugin, 125
- variable1
 - QCaShapePlugin, 116
- VariableNameManager, 147
- variableSubstitutions
 - CmdPushButtonPlugin, 38
- WidgetRef, 149
- writeButtonVariable1
 - QCaPeriodicPlugin, 94