Australian Synchrotron Technical Note AS-200906-01

Qt CA Framework -Getting Started

Author: Andrew Rhyder

9 March 2011

Abstract

Details on configuring an application development environment where the Qt CA Framework will be used.

Introduction

The Qt CA Framework can be used in three ways:

Code Free GUI development using Qt's Designer application with the Qt CA Framework plugins to design GUIs, and the ASgui application to present GUIs to users.

Code Rich GUI development using Qt's application development framework with the Qt CA Framework plugins to design GUI applications.

Console application development using Qt's application development framework with the Qt CA Framework data classes to design console applications that can access EPICS data

This document guides setting up the development environment used to build either the run time environment required for Code Free GUI development, or to develop GUI or console applications.

What configuration is required depends on how the framework will be used. The following usages are typical:

Building the run time environment used for code free GUI development

Writing applications that link to QCa widgets

Laying out forms in creator

Laying out forms in designer

Writing applications that dynamically load UI files

Configuration required for each of the above usages is included in this document.

Prerequisites

Qt 4.7At the time of writing, the Qt CA Framework is being developed using Qt 4.7. Depending on requirements the following Qt components are required:

Application development – Qt SDK 2010-05

or

Application runtime – Qt Libraries 4.7.1

both components are available from http://qt.nokia.com/downloads Note, the Qt Libraries are available via repositories for most major Linux distributions.

Qwt - Qt Widgets for Technical ApplicationsThe Qt CA Framework uses Qwt for plotting. The appropriate version of Qwt is included with the Qt CA framework and must be built before the QCa plugins can be built.The Qwt run time libraries must be installed before using the QCa plugin library.

The Qt library dependencies should be managed when installing from a Linux distribution repository. The dependencies may not be accommodated when installing Qt using downloaded binaries from Nokia. One common dependency problem has been older versions of freetype and fontconfig on RedHat EL 5. Installing freetype-2.4.3 and fontconfig-2.8.0 on RedHat EL 5 resolves this dependency issue.

Setting up your environment

The following environment variable is used by the Qt project files in the QCa framework:

QCAFRAMEWORK	Used by QCa application and components to locate QCa framework

The following script is typical of a script to set up these variables, extend the path to allow command line access to qtcreator and designer, and set up an appropriate EPICS environment.

```
#============================
# Qt
#============================
# Current Qt installation used for QCa development
export QCAQTDIR=~/qtsdk-2010.05
```

```
# Used by QCa applications and components to locate QCa framwork
export QCAFRAMEWORK=~/epicsqt/ca_framework

# Extend path to pick up designer
export PATH=$PATH:$QCAQTDIR/qt/bin

# Extend path to pick up qtcreator
export PATH=$PATH:$QCAQTDIR/bin

#============================
# EPICS
#============================
export EPICS_BASE=/epics/base
export EPICS_HOST_ARCH=linux-x86_64
export EPICS_TOP=${EPICS_BASE}
export EPICS_EXTENSIONS=${EPICS_BASE}/extensions
export EPICS_CA_ADDR_LIST=127.0.0.1
```

Building the run time environment used for code free GUI development

Build the following Qt CA Framework projects:

epicsqt/ca_framework/qwt-5.2.1/qwt.pro

epicsqt/ca_framework/plugins/QCaDesignerPlugin.pro

epicsqt/applications/ASguiApp/ASgui.pro

Deploy the following components:

Qt's Designer

Qt's run time libraries

Qt CA Framework libQCaPlugin.so plugin library

Qt CA Framework ASgui application

Writing applications that link to QCa widgets

When writing applications that link to QCa widgets, the following line should be included in the Qt project file:

LIBS += -L$$(QCAFRAMEWORK)/plugins -lQCaPlugin

(Note, the environment variable QCAFRAMEWORK points to the QCa development framework)

For the above to work the Qca plugin library must be ready for use by the linker. This can be done as follows:

The QCa plugin library (or a link to it) should exist as follows:

/usr/lib/libQCaPlugin.so

The following command should be run to cache the plugin library

ldconfig

Laying out forms in creator

If the QCa plugins are to be used when laying out forms in creator, the QCa plugin library (or a link to it) should exist as follows:

$QCAQTDIR/bin/designer/libQCaPlugin.so

(Note, the environment variable QCAQTDIR should be set to point to the version of Qt used for QCa development)

Laying out forms in designer

If the QCa plugins are to be used when laying out forms in designer, the QCa plugin library (or a link to it) should exist as follows:

$QCAQTDIR/qt/plugins/designer/libQCaPlugin.so

(Note, the environment variable QCAQTDIR should be set to point to the version of Qt used for QCa development)

Writing applications that dynamically load UI files

When writing applications that dynamically load UI files (using QUiLoader), where the UI files include QCa plugins, the QCa plugin library (or a link to it) should exist as follows:

<your-executable-directory>/designer/libQCaPlugin.so

Alternatively, your application can specify where QUiLoader should search for the plugin using QUiLoader::addPluginPath()