logo

QEFileBrowser Widget

Andrew Starritt

26th June 2025

Copyright (c) 2025 Australian Synchrotron

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.3 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts.  
A copy of the license is included in the section entitled "GNU Free Documentation License" within the QE\_QEGuiAndUserInterfaceDesign document.

Contents

[Introduction 3](#__RefHeading___Toc335_2427085548)

[Description 3](#__RefHeading___Toc341_2427085548)

[Properties 4](#__RefHeading___Toc339_2427085548)

# Introduction

This document describes in detail the QEFileBrowser widget provided by the EPICS Qt, aka the QE Framework.

This document was created as a separate widget specification document. The main reason for this is ease of maintenance and avoiding editing large and unwieldly word documents.

The QE Framework is distributed under the GNU Lesser General Public License version 3, distributed with the framework in the file LICENSE. It may also be obtained from here: <http://www.gnu.org/licenses/lgpl-3.0-standalone.html>

The QEFileBrowser widget provides EPICS-awareness via a single control Process Variable (PV), associated with a stringin/stringout record or a wavform record with FTVL set to CHAR.

## Description

The QEFileBrowser widget class inherits directly QEAbstractWidget which provides many of the standard QE properties (e.g. variableAsToolTip, userLevelEnabled) and indirectly from the QFrame class which which allows the QEFileBrowser widget to have a boarder.

The QEFileBrowser widget allows the user to browse existing files from a certain directory.

When connected to a QEFileImage it can be used to select the file being viewed. In this scenario, the QEFileBrowser ‘selected’ signal is connected to the QEFileImage ‘setImageFileName’ slot. Note, this is only one method a QEFileImage widget can use to source its image.

Within Qt Designer, it has the following graphical representation (surrounded by a red rectangle):

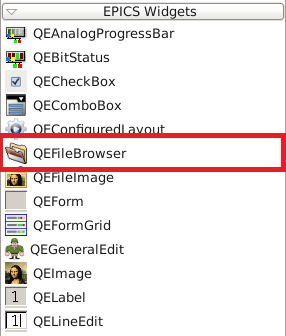


Figure 1 QEFileBrowser within Qt Designer

## Properties

The QEFileBrowser has the following properties (that can be controlled by the user). Only the properties provided by QEFileBrowser itself are described here:

* **variable**Defines the process variable name used b the widget.
* **variableSubsitutions**Defines the default (and lowest priority) variable name substitutions values used by the widget.
* **directoryPath**

Default directory where to browse files when QEFileBrowser is launched for the first time.

* **showDirectoryPath**

Show/hide directory path line edit where the user can specify the directory to browse files.

* **showDirectoryBrowser**

Show/hide button to open the dialog window to browse for directories and files.

* **showRefresh**

Show/hide button to refresh the table containing the list of files being browsed.

* **showTable**

show/hide table containing the list of files being browsed.

* **showColumnTime**

show/hide column containing the time of creation of files.

* **showColumnSize**

Show/hide column containing the size (in bytes) of files.

* **showColumnFilename**

Show/hide column containing the name of files.

* **showFileExtension**

show/hidethe extension of files.

* **fileDialogDirectoriesOnly**

Enable/disable the browsing of directories-only when opening the dialog window.

* **fileCreationAllowed**When set (and fileDialogDirectories is de-selected), the widget will use the file save dialog as opposed to the fileOpen dialog. This allows the selection/creation of new files, and also warns the user if the file already exists.
* **fileFilter**

Specify which files to browse. To specify more than one filter, please separate them with a ";". Example: \*.py;\*.ui (this will only display files with an extension .py or .ui).

* **optionsLayout**Change the order of the widgets. Valid orders are: TOP, BOTTOM, LEFT and RIGHT.
* **margin**

Set the margin between internal widgets and the QEFileBrowser border.

The following figure illustrates the QEFileBrowserwidget in production:

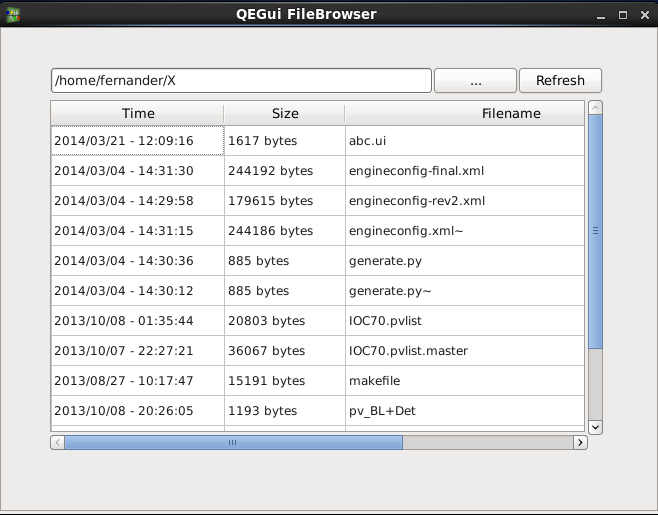


Figure 2 QEFileBrowser displaying existing files in directory "/home/fernander/X"