

Summary

This TN provides information about the integration of the Cisco port of the webdriver protocol [1], being qtwebdriver [2], into the qtbrowser as delivered by Metrological.

Web Driver

Description

QtWebDriver is a WebDriver implementation for Qt. It can be used to perform automated Selenium testing of applications based on QtWebkit.

Building

When building qtbrowser with webdriver support, the following sections are to be taken into account in addition to the settings already required.

Pre-requisites

Required tools:

- Python 2.7.2+
- Generate-Your-Project, or gyp, framework [3]

Environmental settings

- QT_INC_PATH – path to qt5 header files from the staging build directory (e.g. \$(STAGING_DIR)/usr/include/qt5)
- QT_BIN_PATH – path to mips cross-compiler tools (e.g. python, gyp, g++)

When generating the qtbrowser Makefile from the qtbrowser.pro file, the following options are to be set (i.e. *qmake \$(OPTIONS) qtbrowser.pro*):

```
OPTIONS = \
```

```
INCLUDEPATH+=$(STAGING_DIR)/usr/include/qt5/QtWebDriver \
```

```
LIBS+="-lchromium_base -lWebDriver_core -lWebDriver_extension_qt_base -lWebDriver_extension_qt_web" \
```

```
BROWSER_CONFIG+=qtwebdriver
```

Procedure

- qtwebdriver package is depending on the qtwebkit package; please make sure that qtwebkit package is build first;
- Make sure that the *gyp* tool is installed properly, see [3];
- Copy wd.gypi file to the qtwebdriver folder;
- Execute the build script from the qtwebdriver folder: *./build ./out .mips Release \$(STAGING_DIR)*;
- After a successful compilation of qtwebdriver package:

- export the inc-folder containing the qtwebdriver header-files to the staging directory such that the qtbrowser package can find them during compilation (e.g. `cp -dpfru $(@D)/inc $ (STAGING_DIR)/usr/include/qt5/QtWebDriver`);
- export the generated libraries to the staging directory, such that the qtbrowser package can find them during linking (e.g. `cp -dpfu $(@D)/out/bin/mips/Release/* $(STAGING_DIR)/usr/lib`);
- Finally, install the so-libraries of qtwebdriver to the target directory, such that they can be incorporated into the final image (e.g. `$(INSTALL) -D -m 0755 $(@D)/out/bin/mips/Release/*.so $ (TARGET_DIR)/usr/lib`).

References

- [1] <https://w3c.github.io/webdriver/webdriver-spec.html>
- [2] <https://github.com/cisco-open-source/qtwebdriver>
- [3] <https://gyp.gsrc.io/index.md>