QSpin
Quality & SPI Network

Keywords:	QTaste, test, guidelines, quick start, demo

REVISION RECORD

Rev.	Date	Description	Pages
1.0	21/10/2009	First version for Open Source release	All
1.1	29/03/2012	Some minor change for release 1.1	All

TABLE OF CONTENTS

1. INSTALLATION OF THE QTASTE FRAMEWORK

The QTaste framework is mainly developed in java programming language and python. So by definition, it can be installed on any platform running java VM 1.6. However, it has been only validated on Windows (XP Pro / Vista) and Linux platform (Ubuntu 9.04, Fedora 11).

1.1. QTaste system requirements

- Java Virtual Machine (JDK) 1.6 (http://java.sun.com/javase/downloads/index.jsp)
- (optional) subversion command-line client accessible from PATH (“*svn*”), for test script versioning
- At least 100 MB of disk space
- At least 256 MB of system memory (Running with less memory may cause disk swapping which has a severe effect on performance. Very large programs may require more RAM for adequate performance.)

1.2. Installation of the QTaste

The QTaste framework is composed of:

- Test Engine kernel
- Simulators base classes
- Other tools
- Components Test API and Component Implementations
- Test Suites containing test scripts and test data
- Test Campaigns
- Testbeds configurations

The installer available on source forge (*https://sourceforge.net/projects/qtaste/files/*) contains the QTaste Kernel (sources and binaries), the demonstration (sources and binaries) and the documentation.

2. QTASTE DEMO QUICK START

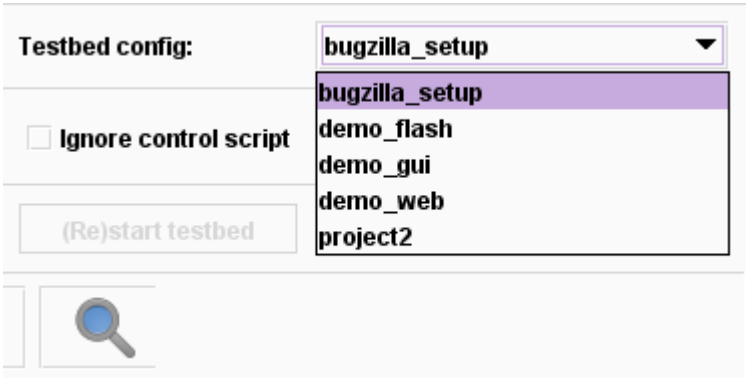
2.1. About the demos

The goal of demos are to demonstrate that QTaste can be used in different kind of environments like Web, Windows GUI, Databases ,embedded systems or others. For that reason, each demo required the installation of the test environment. It comes with some pre-requisites of each demo described in the README.txt file located in the Test suite directory of each demo (demo/TestSuites/*)

2.2. Starting the calculator demo

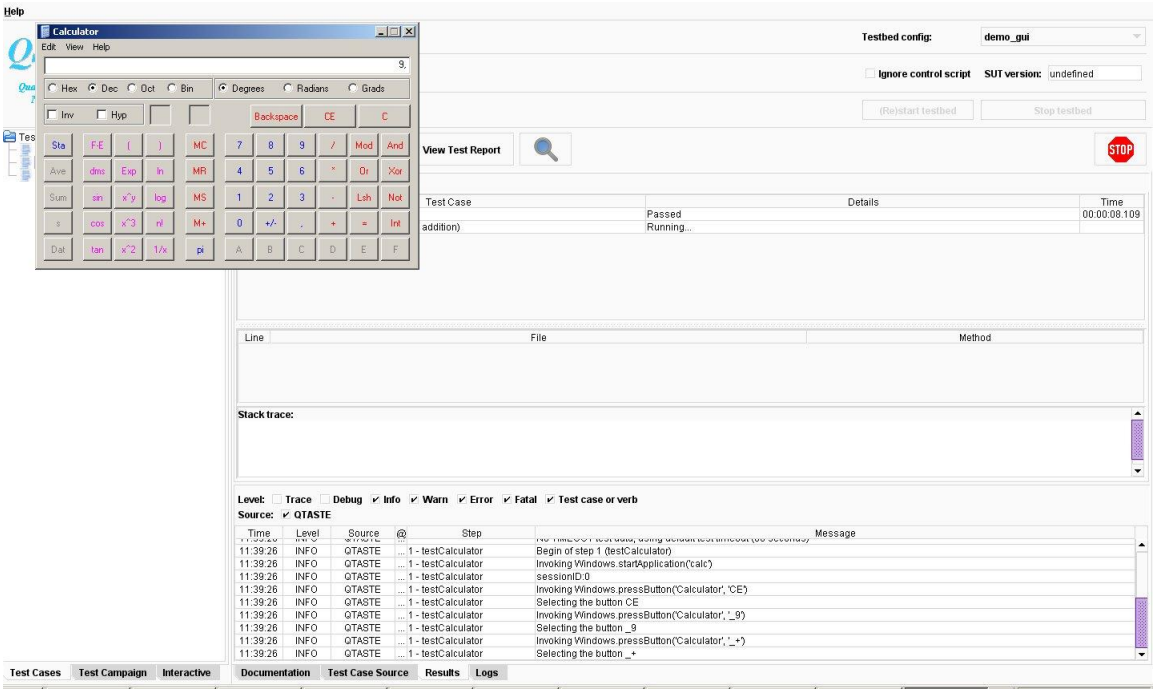
Keep in mind that this demo is only working on Windows platform as this example demonstrates how QTaste can communicate with low-level GUI component controls. It uses pyWinAuto python component (http://pywinauto.openqa.org/) in order to communication with the Windows components.

- In this implementation, it requires the installation of python (tested with 2.5) and python has to be available in the PATH environment variable. (http://www.python.org/download/releases/2.5/)
- JDK 1.6 has to be installed as well as it is part of the QTaste pre-requisite.
- Start the Qtaste demo **using the startUI.cmd (or startUI.sh) script** located in demo directory.
- Select the testbed called "demo_gui" in the testbed configuration drop-down (upper-right corner of the QTaste GUI)



- Select the "TestCalculator" TestSuites (in the treeview on the left).
- Click on the "Run Test" button

- And QTaste will execute the test on the Windows Calculator



2.3. Starting the translate demo

This demo demonstrates an example of Web functional test using the Selenium implementation. (http://seleniumhq.org/)
This demo can be started from Windows or Linux platform.

- This demo required Firefox (tested with version 3.x). More browsers are supported. Please have a look at the selenium web site to know about the supported browser.
- JDK 1.6 has to be installed as well as it is part of the QTaste pre-requisite.
- Start the Qtaste demo **using the startUI.cmd (or startUI.sh) script** located in demo directory.
- Select the testbed called “demo_web” in the testbed configuration drop-down (upper-right corner of the QTaste GUI)
- Select the “TestTranslate” TestSuites (in the treeview on the left).
- Click on the “Run Test” button

- And QTaste will execute the test using Firefox