QSpin
Quality & SPI Network

QTaste Quick start guide		Rev. 1.2
QSpin Tailored automated system test environment	*Page i*	

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
1.2	24/09/2012	Requirements module + bug fixing release 1.2	All

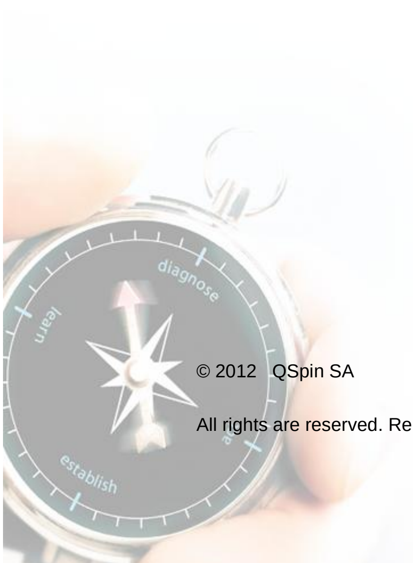


TABLE OF CONTENTS

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 (OpenSuse 11.04).

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.)
- On Linux system, Java VM requires some graphical gnome libraries. These libraries have to be installed.

  https://sourceforge.net/apps/mantisbt/qtaste/view.php?id=3

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.

*To start the installer, just click on the “*QTaste_1.2-installer.jar*”.*

Warning: *It is not recommended to install QTASTE in a directory containing spaces (like Program Files/QTaste) due to limitations in some shell scripts. (http://sourceforge.net/apps/mantisbt/qtaste/view.php?id=6)*

2. QTASTE DEMO QUICK START

2.1. About the demos

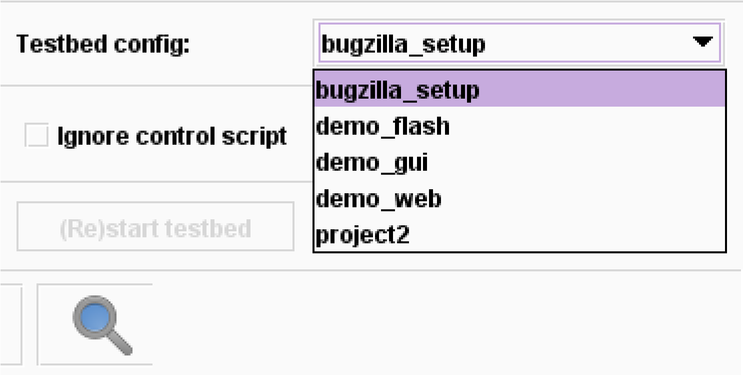
The goal of demos are to demonstrate that QTaste can be used in different kind of environments like Web, Windows GUI, Databases, embedded systems or others. For that reason, each demo required the installation of the test environment. It comes with some pre-requisites of each demo described in the README.txt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

2.2. Starting the calculator demo

Unfortunately, this demo is only working on XP 32 bits platform.
https://sourceforge.net/apps/mantisbt/qtaste/view.php?id=4

Keep in mind that this demo is only working on Windows XP 32 bits 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.

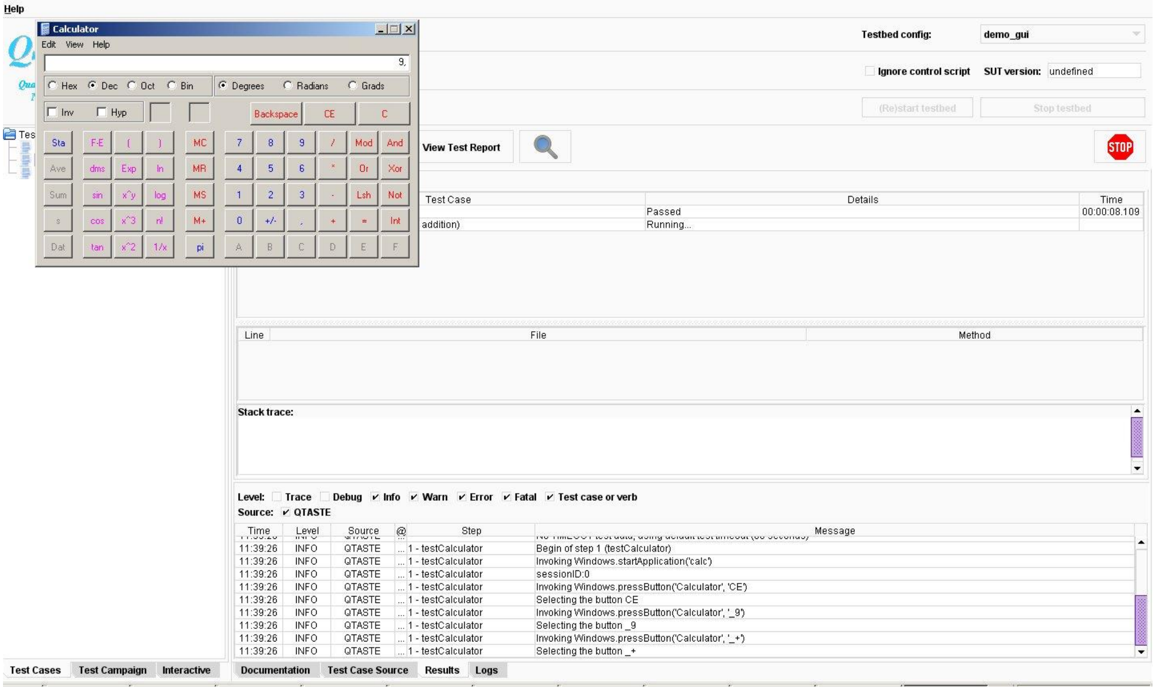
- In this implementation, it requires the installation of python (tested with 2.5) and python has to be available in the PATH environment variable. (http://www.python.org/download/releases/2.5/)
- JDK 1.6 has to be installed as well as it is part of the QTaste pre-requisite.
- Start the Qtaste demo **using 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)



Testbed config: bugzilla_setup

□ Ignore control script

- 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



Testbed config: demo_gui

□ Ignore control script SUT version: undefined

Test Case	Details	Time
	Passed	00:00:08.109
addition)	Running...	

Stack trace:

Level: □ Trace □ Debug ☑ Info ☑ Warn ☑ Error ☑ Fatal ☑ Test case or verb

Source: ☑ QTASTE

```
 Time     Level   Source   @    Step                 Message
11:39:26  INFO    QTASTE   ... 1 - testCalculator   Begin of step 1 (testCalculator)
11:39:26  INFO    QTASTE   ... 1 - testCalculator   Invoking Windows.startApplication('calc')
11:39:26  INFO    QTASTE   ... 1 - testCalculator   sessionID:0
11:39:26  INFO    QTASTE   ... 1 - testCalculator   Invoking Windows.pressButton('Calculator', 'CE')
11:39:26  INFO    QTASTE   ... 1 - testCalculator   Selecting the button CE
11:39:26  INFO    QTASTE   ... 1 - testCalculator   Invoking Windows.pressButton('Calculator', '_9')
11:39:26  INFO    QTASTE   ... 1 - testCalculator   Selecting the button _9
11:39:26  INFO    QTASTE   ... 1 - testCalculator   Invoking Windows.pressButton('Calculator', '_+')
11:39:26  INFO    QTASTE   ... 1 - testCalculator   Selecting the button _+
```

Help
Calculator
Edit View Help
9,
Hex Dec Oct Bin
Degrees Radians Grads
Inv Hyp
Backspace CE C
Sta F-E ( ) MC 7 8 9 / Mod And
Ave dms Exp ln MR 4 5 6 * Or Xor
Sum sin x^y log MS 1 2 3 - Lsh Not
s cos x^3 n! M+ 0 +/- , + = Int
Dat tan x^2 1/x pi A B C D E F
(Re)start testbed
Stop testbed
View Test Report
STOP
Line File Method
Test Cases Test Campaign Interactive
Documentation Test Case Source Results Logs

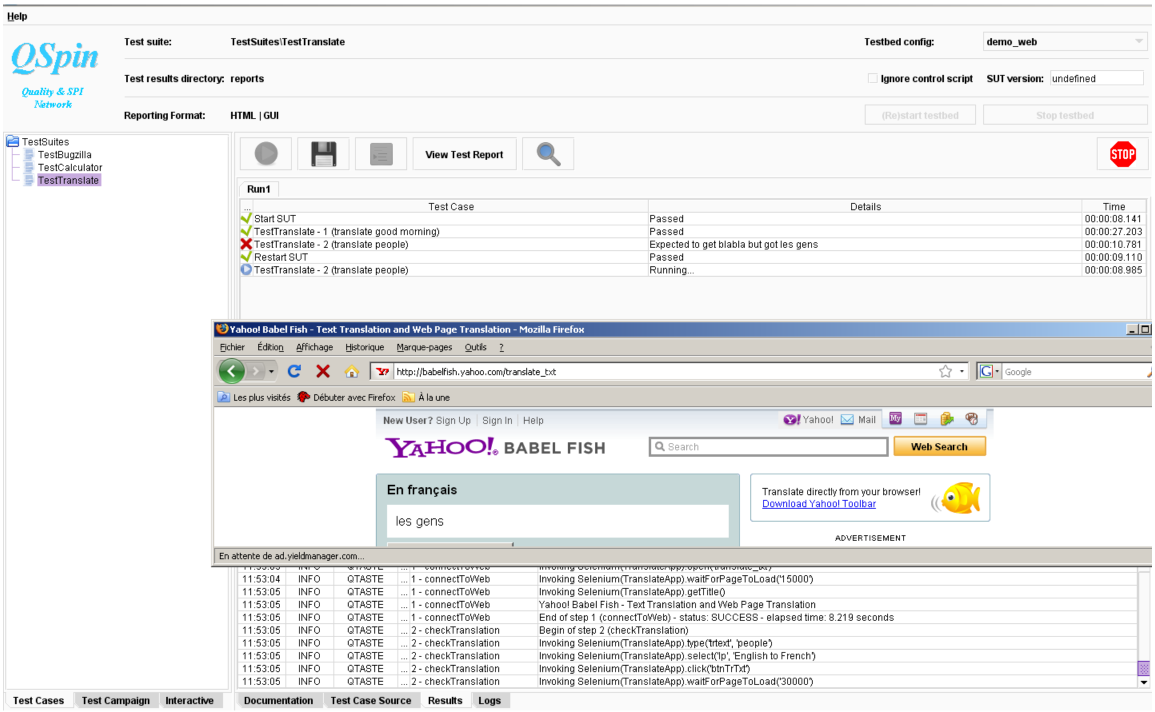
2.3. Starting the translate demo

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



Test suite: TestSuites\TestTranslate

Testbed config: demo_web

Test results directory: reports

□ Ignore control script

SUT version: undefined

Reporting Format: HTML | GUI

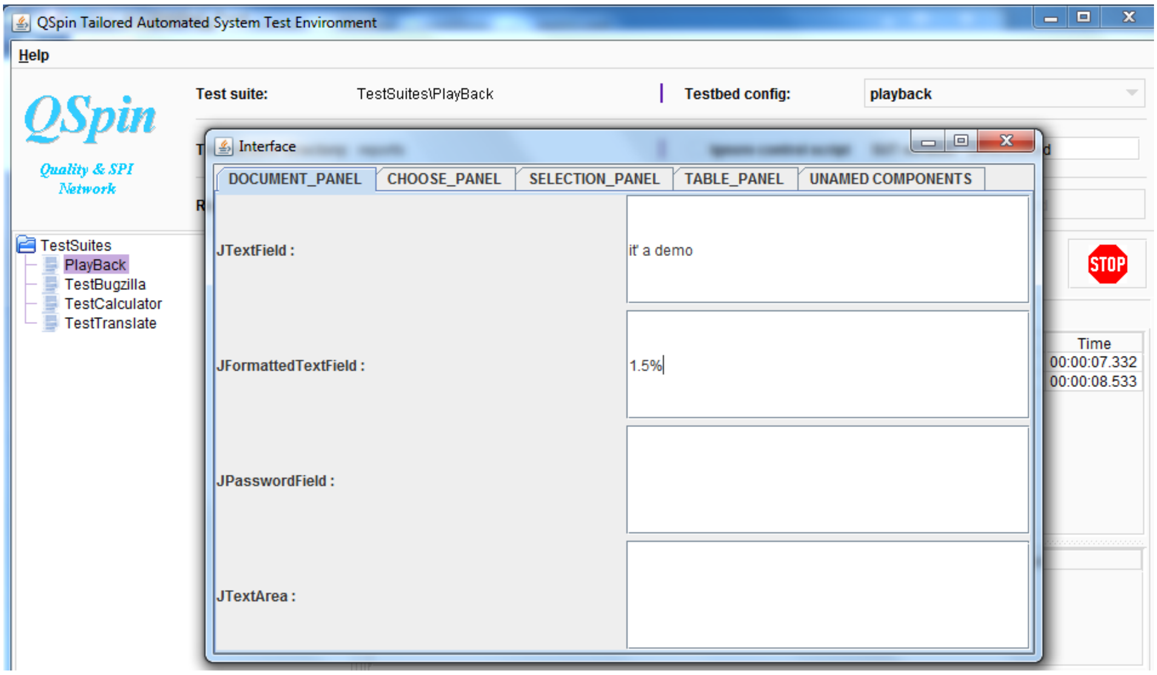
...	Test Case	Details	Time
✓	Start SUT	Passed	00:00:08.141
✓	TestTranslate - 1 (translate good morning)	Passed	00:00:27.203
✗	TestTranslate - 2 (translate people)	Expected to get blabla but got les gens	00:00:10.781
✓	Restart SUT	Passed	00:00:09.110
●	TestTranslate - 2 (translate people)	Running...	00:00:08.985

```
11:53:04  INFO  QTASTE  ... 1 - connectToWeb      Invoking Selenium(TranslateApp).waitForPageToLoad('15000')
11:53:05  INFO  QTASTE  ... 1 - connectToWeb      Invoking Selenium(TranslateApp).getTitle()
11:53:05  INFO  QTASTE  ... 1 - connectToWeb      Yahoo! Babel Fish - Text Translation and Web Page Translation
11:53:05  INFO  QTASTE  ... 1 - connectToWeb      End of step 1 (connectToWeb) - status: SUCCESS - elapsed time: 8.219 seconds
11:53:05  INFO  QTASTE  ... 2 - checkTranslation  Begin of step 2 (checkTranslation)
11:53:05  INFO  QTASTE  ... 2 - checkTranslation  Invoking Selenium(TranslateApp).type('trtext', 'people')
11:53:05  INFO  QTASTE  ... 2 - checkTranslation  Invoking Selenium(TranslateApp).select('lp', 'English to French')
11:53:05  INFO  QTASTE  ... 2 - checkTranslation  Invoking Selenium(TranslateApp).click('btnTrTxt')
11:53:05  INFO  QTASTE  ... 2 - checkTranslation  Invoking Selenium(TranslateApp).waitForPageToLoad('30000')
```

2.4. Starting the JavaGUI demo

This demo demonstrates an example of automation of control of a Java application developed using Swing.

- Start the Qtaste demo **using the startUI.cmd (or startUI.sh) script** located in demo directory.
- Select the testbed called “playback” in the testbed configuration drop-down (upper-right corner of the QTaste GUI)
- Select the “PlayBack” TestSuites (in the treeview on the left).
- Click on the “Run Test” button



Test suite: TestSuites\PlayBack

Testbed config: playback

JTextField : it' a demo

JFormattedTextField : 1.5%

JPasswordField :

JTextArea :

00:00:07.332
00:00:08.533