Setup (Automated) UI Testing for the Smart Client Using QFTest

QF-test Documentation:

http://www.qfs.de/en/qftest/tutorial.html **- html**

http://www.qfs.de/qftest/tutorial_en.pdf **- pdf**

http://www.qfs.de/en/qftest/demos.html **- video**

Before You Start

- Download QF-Test: download page
- Licensing : in the qftest install folder (`../qfs/qftest/`) copy these files: license, license.old, license.new
- On first startup of QF-Test and/or the System Under Test (SUT) via QF-Test you might get a security warning from the Windows firewall asking whether to block Java or not. As QF-Test communicates with the SUT by means of network protocols, this must not be blocked by the local firewall in order to allow automated testing.

JRE Instrumentation

- make sure JRE is installed;
- instrumentation of the JRE on which the SUT will run is now the primary method of providing a way for QF-Test to interact with the SUT. Normally you don't need to instrument standard JREs because QF-Test passes the same information to the SUT's JRE via environment variables. However, for special cases instrumentation may be required anyway
    - in main menu select **Extras** > **Manage Instrumentation** > **Instrument** (if the path is not shown, click **Search** first and point to the jre install folder)
    - the State changes to: **Instrumentation current**

Servoy

- Install with default settings (Servoy Developer)

> If Servoy is installed on Windows Xp, make sure that Microsoft Microsoft Visual C++ 2008 Redistributable Package (x86) is preinstalled – otherwise Postgres won't be installed correctly

- Create/export the solution for testing - System Under Test (SUT)
- Start AppServer and import the SUT

Setup

- Start QF-Test
- Main menu **Extras** > **Quickstart Wizard** > Select **A java application launched through WebStart**
- When asked for the Java Webstart executable: 'javaws' - default is fine
- When asked for jnlp file, path add the path to SC: example for crm_sample: http://localhost:8080/servoy-client/solutions/solution/servoy_sample_crm.jnlp

- Next step give a proper client name: for ex: 'svy_sample' - This name serves as central reference to identify your client
- Next step: **Start automatically** is checked – QF-Test will launch SmartClient
- Wait for the client to launch
- Save the script (File-Save) with a desired name

Main Window Description

- The left part of the main window contains the tree structure that represents the test-suite. The right side displays the details of a selected node. (In case you don't see the details view, you can switch it on with the View! Details menu option.) At the bottom of the main window you'll see the terminal output area, which displays stan-dard messages and communications in between your test-suite and the client applica-tion you are testing.
- The basic structure of a test-suite and thus the child nodes of theTest-suite root node is fixed. An arbitrary number of Test-set, Test-case or Test nodes are followed by the **Procedures**, **Extras** and **Windows and components** nodes. The **Procedures** node holds **Packages** and **Procedures**.

Starting the Application

- If QF-Test is configured – JRE instrumentation is done, jnlp path is added, select Setup node and hit Enter
- (When qf-test is run the first time, there is a check - **Start automatically**)

Our first step will be to examine the **Setup** node:

- Set variable: client='client'
- Start SUT client: javaws [$(client)]- starts the application for the client you will be testing, called the System Under Test (SUT).
- Wait for client to connect [$(client)]- waits until the new Java Virtual Machine, in which your SUT will run, binds itself to QF-Test.
- If the jnlp path was correct, the solution will open in SmartClient and the QF-Test's buttons bar a couple of buttons will become enabled (like: **Start Recording** – red circle, **Recording checks** – green check ...)
- In the first field, labeled **Client** is an ambiguous name for the SUT application. You can give the SUT any name you wish, but the same name will be used as a reference to the SUT in other nodes. In other words, this name must be used consistently.

Context Help

While working with various nodes in QF-Test you may, for example, require some asistance in remembering what purpose they are used for. To serve this need, QF-Test comes with context-sensitive help, driven simply by your mouse. Move the mouse pointer over an element you would like to have assistance with, then click the right mouse-button. In the now appearing popup menu, you'll see an entry labeled '**What's this?**' By selecting this option from the popup menu, the appropriate section in the user's manual will be brought up and displayed in your standard browser.

Recording Events

- Click **Record** button
- Go to SC, do the actions needed, typing data / adding records / delete
- When done go to qf-test and hit **Stop** button
- A sequence is created which can be renamed properly-select seq.-hit F2 key-give it a name
- Multiple sequences can be packed in Procedures by multiple selecting sequences, right click, transform node into procedure

Recording Checks

- Click **Record** button
- Click green – **check** button
- In SC left click the fields that you want checked in the desired order
- When finished, go back to qf-test main page and hit **Stop** recording-square button, the sequence was recorded
- Multiple sequences can be packed in procedures by multiple selecting sequences, right click, **Transform node into** - **Procedure**, which can be renamed properly-select proc.-hit F2 key-give it a name
- Procedures can be directly recorded by hitting **Record** button and then

Playback - Running the Script

- If only the Setup part of the script is tested, then select **Setup** node of the tree and hit **Enter**
- If only a sequence is tested, or multiple by shift-selecting, then select that sequence and hit **Enter**
- If all the script is tested then select **Extras** node, and hit **Enter**
- TIP: for easier following of the test run, add a default delay: go to **qf-test** > **main menu** > **edit** > **options** > **replay** > **delays** > **default delays** add something like 350 in the before and after fields.

Tip: use the links at the top for more detailed documentation and video tutorials.