



SilkTest® Classic

Using the Basic Workflow with the Open Agent

**Borland Software Corporation
4 Hutton Centre Dr., Suite 900
Santa Ana, CA 92707**

Copyright 2009-2010 Micro Focus (IP) Limited. All Rights Reserved. SilkTest contains derivative works of Borland Software Corporation, Copyright 1992-2010 Borland Software Corporation (a Micro Focus company).

MICRO FOCUS and the Micro Focus logo, among others, are trademarks or registered trademarks of Micro Focus (IP) Limited or its subsidiaries or affiliated companies in the United States, United Kingdom and other countries.

BORLAND, the Borland logo and SilkTest are trademarks or registered trademarks of Borland Software Corporation or its subsidiaries or affiliated companies in the United States, United Kingdom and other countries.

All other marks are the property of their respective owners.

Contents

- Using the Basic Workflow with the SilkTest Open Agent.....4**
 - Starting SilkTest4
 - Creating a New Project.....4
 - Configuring the Insurance Company Web Application.....4
 - Recording a Testcase for the Insurance Company Web Site.....5
 - Replaying the Testcase for the Insurance Company Web Site.....6

Using the Basic Workflow with the SilkTest Open Agent

This tutorial provides a step-by-step introduction to using SilkTest for BMC Transaction Management Application Response Time by Borland Software to test a Web application that uses dynamic object recognition. Dynamic object recognition enables you to write test cases that use XPath queries to find and identify objects.

For the sake of simplicity, this guide assumes that you have installed SilkTest and are using the Insurance Company sample Web application, <http://demo.borland.com/InsuranceWebExtJS/index.jsf>.

For additional information about SilkTest Classic, including information about sample scripts and applications, refer to the SilkTest Classic Help. To view the help, choose **Start > Programs > Silk > SilkTest <version> > Documentation > SilkTest Classic > SilkTest Classic Online Help**.

Starting SilkTest

Choose **Start > Programs > Silk > SilkTest <version> > SilkTest Classic**.
SilkTest Classic opens.

Creating a New Project

The type of project that you select determines the default SilkTest Agent. For Web application projects, the Open Agent is automatically set as the default agent. SilkTest Classic uses the default agent when configuring an application and recording a testcase.

1. Choose **File > New Project**.
The **New Project** dialog box opens.
2. Under **Rich Internet Applications**, click **Web**.
3. Click **OK**.
The **Create Project** dialog box opens.
4. Type a project name and a description in the appropriate text boxes.
5. Click **OK** to save your project in the default location, `<SilkTest installation directory>\Project`.
If you do not want to save your project in the default location, click **Browse** and specify the folder in which you want to save your project.
SilkTest Classic creates your project and displays nodes on the **Files** and **Global** tabs for the files and resources associated with this project.

Configuring the Insurance Company Web Application

Configure the application to set up the environment that SilkTest Classic will create each time you record or replay a testcase.

When you configure an application, SilkTest Classic automatically creates a base state for the application. An application's base state is the known, stable state that you expect the application to be in before each test begins execution, and the state the application can be returned to after each test has ended execution.

1. Click **Configure Application** on the Basic Workflow bar.

If you do not see **Configure Application** on the workflow bar, ensure that the default agent is set to the Open Agent.

2. Double-click **Web Site Test Configuration**.

The **New Web Site Configuration** page opens.

3. From the **Browser Type** list box, select **Internet Explorer**.

You can use Firefox to replay tests but not to record them.

4. Perform one of the following steps:

- **Use existing browser** – Click this option button to use a browser that is already open when you configure the test. For example, if the Web page that you want to test is already displayed in a browser, you might want to use this option.
- **Start new browser** – Click this option button to start a new browser instance when you configure the test. Then, in the **Browse to URL** text box specify the Web page to open.

For this tutorial, close all open browsers and then click **Start new browser** and specify <http://demo.borland.com/InsuranceWebExtJS/>.

5. Click **Finish**.

The Choose name and folder of the new frame file page opens. SilkTest Classic configures the recovery system and names the corresponding file `frame.inc` by default.

6. Navigate to the location in which you want to save the frame file.

7. In the **File name** text box, type the name for the frame file that contains the default base state and recovery system. Then, click **Save**.

SilkTest Classic automatically creates a base state for the application. By default, SilkTest Classic lists the caption of the main window of the application as the locator for the base state.

When you configure an application, SilkTest Classic adds an include file based on the technology or browser type that you enable to the Use files location in the **Runtime Options** dialog. For instance, if you configure an application that uses an Internet Explorer or Firefox browser, SilkTest Classic adds the `xBrowser.inc` file to the **Runtime Options** dialog.

SilkTest Classic opens the Web page. Record the testcase whenever you are ready.

Recording a Testcase for the Insurance Company Web Site

Record a testcase for the Insurance Company sample Web site to see how SilkTest Classic creates a test.

1. Click **Record Testcase** on the Basic Workflow bar.

The **Record Testcase** dialog box opens.

2. Type the name of your testcase in the **Testcase name** text box.

For example, type `zipTest`.

Testcase names are not case sensitive; they can be any length and consist of any combination of alphabetic characters, numerals, and underscore characters.

3. From the **Application State** list box, select **DefaultBaseState** to have the built-in recovery system restore the default base state before the testcase begins executing.

The testcase is recorded in the script file as: `testcase testcase_name ()`.

4. If you do not want SilkTest Classic to display the status window during playback when driving the application to the specified base state, uncheck the **Show AppState status window** check box.

Typically, you check this check box. However, in some circumstances it is necessary to hide the status window. For instance, the status bar might obscure a critical control in the application you are testing.

5. Click **Start Recording**.

SilkTest Classic:

- Closes the **Record Testcase** dialog box
- Starts your application, if it was not already running
- Removes the editor window from the display
- Displays the **Recording** status window
- Waits for you to take further action

6. In the Insurance Company Web site, perform the following steps:

- a) From the **Select a Service or login** list box, select **Auto Quote**.
The **Automobile Instant Quote** page opens.

- b) Type a zip code and email address in the appropriate text boxes, click an automobile type, and then click **Next**.

To follow this tutorial step-by-step, type **92121** as the zip code, **jsmith@gmail.com** as the email address and specify **Car** as the automobile type.

- c) Specify an age, click a gender and driving record type, and then click **Next**.

For example, type **42** as the age, specify the gender as **Male** and **Good** as the driving record type.

- d) Specify a year, make, and model, click the financial info type, and then click **Next**.

For example, type **2010** as the year, specify **Lexus** and **RX400** as the make and model, and **Lease** as the financial info type.

A summary of the information you specified appears.

- e) Point to the **Zip Code** that you specified and press **Ctrl+Alt** to add a verification to the script.

You can add a verification for any of the information that appears.

The **Verify Properties** dialog box opens.

- f) Check the **textContent** check box and then click **OK**.

A verification action is added to the script for the zip code text.

An action that corresponds with each step is recorded.

7. To review what you have recorded, click **Stop Recording** in the **Recording** window.

SilkTest Classic displays the **Record Testcase** dialog box, which contains the code that has been recorded for you.

8. Click **Paste to Editor**.

9. Choose **File > Save**.

- a) Navigate to the location in which you want to save the test.

- b) In the **File name** text box, type the name for the test and then click **Save**.

For example, type **zipTest**.

- c) Click **Yes** when prompted to add the file to the project.

Replay the test to ensure that it works as expected. You can modify the test to make changes if necessary.

Replaying the Testcase for the Insurance Company Web Site

Replay a test to ensure that it works as expected.

1. Make sure that the testcase you want to run is in the active window.
2. Click **Run Testcase** on the Basic Workflow bar.
SilkTest displays the **Run Testcase** dialog box, which lists all the testcases contained in the current script.
3. Select the `zipTest` testcase.
4. To wait one second after each script line is executed, check the **Animated Run Mode (Slow-Motion)** check box.
Typically, you will only use this check box if you want to watch the testcase run. For instance, if you want to demonstrate a testcase to someone else, you might want to check this check box.
5. Click **Run**.
SilkTest Classic runs the testcase and generates a results file. The results file describes whether the test passed or failed, and provides summary information.

Index

P

projects
 configuring web 4
 creating web 4

S

SilkTest
 launching 4
 quick start tutorial 4

T

testcase
 recording 5
 replaying 6

tutorial
 basic workflow 4

W

web applications
 configuring project 4
 creating project 4
 recording testcase 5
 replaying testcase 6

X

xBrowser
 configuring project 4
 creating project 4
 recording testcase 5
 replaying testcase 6