

Borland[®]

THE OPEN ALM COMPANY

SilkTest[®] 2008 R2

**SilkTest Quick Start Tutorial for Dynamic
Object Recognition**

Borland Software Corporation
8310 North Capital of Texas Hwy
Building 2, Suite 100
Austin, Texas 78731
<http://www.borland.com>

Borland Software Corporation may have patents and/or pending patent applications covering subject matter in this document. Please refer to the product CD or the About dialog box for the list of applicable patents. The furnishing of this document does not give you any license to these patents.

Copyright © 2009 Borland Software Corporation and/or its subsidiaries. All Borland brand and product names are trademarks or registered trademarks of Borland Software Corporation in the United States and other countries. All other marks are the property of their respective owners.

Contents

- Quick Start Tutorial for Dynamic Object Recognition.....4**
- Starting SilkTest.....4
- Creating a New Project.....4
- Enabling Extensions for the GMO Web Application.....5
- Creating a Base State for the GMO Web Application.....6
- Creating a Testcase for the GMO Web Application.....8
- Running a Testcase for the GMO Web Application.....10

Quick Start Tutorial for Dynamic Object Recognition

This tutorial provides a step-by-step introduction to using SilkTest 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. Dynamic object recognition uses a `Find` or `FindAll` method to identify an object in a test case.

For the sake of simplicity, this guide assumes that you have installed SilkTest and are using the GMO sample web application.

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

Related Topics

- [Starting SilkTest](#) on page 4
- [Creating a New Project](#) on page 4
- [Enabling Extensions for the GMO Web Application](#) on page 5
- [Creating a Base State for the GMO Web Application](#) on page 6
- [Creating a Testcase for the GMO Web Application](#) on page 8
- [Running a Testcase for the GMO Web Application](#) on page 10

Starting SilkTest

Choose **Start > Programs > Borland > SilkTest <version> > SilkTest**.
SilkTest opens.

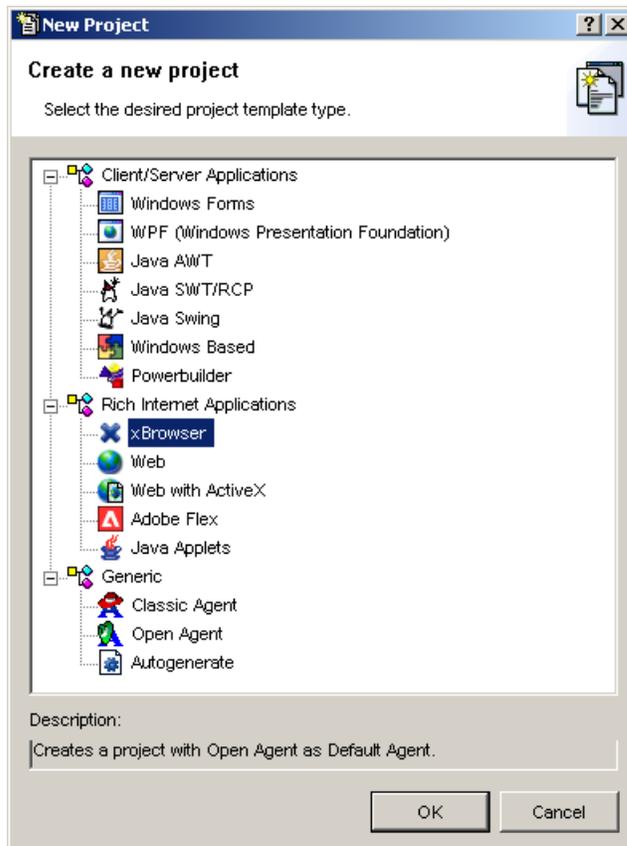
Related Topics

- [Quick Start Tutorial for Dynamic Object Recognition](#) on page 4

Creating a New Project

The type of project that you select determines the default SilkTest Agent. For xBrowser projects, the Open Agent is automatically set as the default agent. SilkTest uses the default agent when recording window identifiers or enabling extensions.

1. Choose **File > New Project**.
The **New Project** dialog box opens.
2. Under **Rich Internet Applications**, click **xBrowser**.



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 creates your project and displays nodes on the **Files** and **Global** tabs for the files and resources associated with this project.

Related Topics

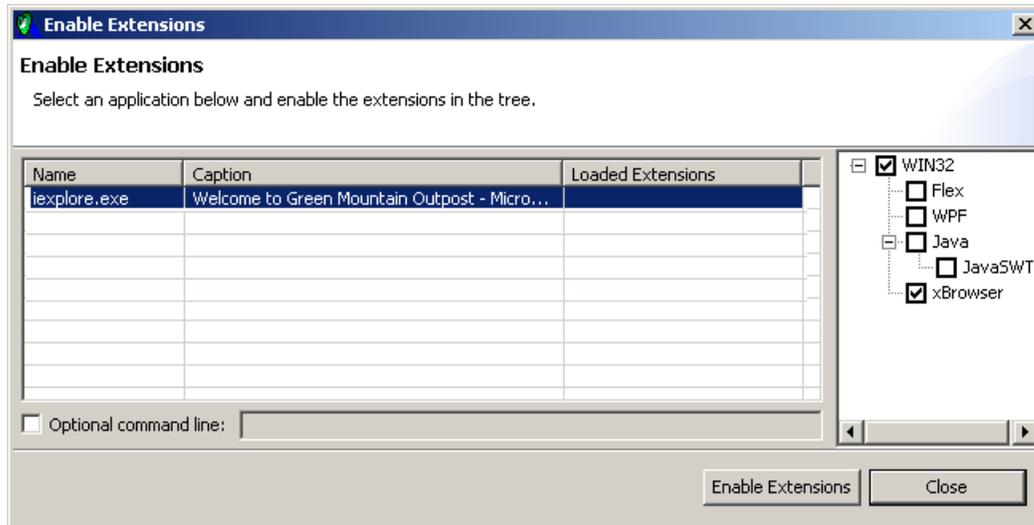
- [Quick Start Tutorial for Dynamic Object Recognition](#) on page 4

Enabling Extensions for the GMO Web Application

An extension is a file that serves to extend the capabilities of, or data available to, a more basic program. SilkTest provides extensions for testing applications that use non-standard controls in specific development and browser environments.

1. Open Internet Explorer and navigate to the GMO web site at: <http://demo.borland.com/gmopost/>.

- In SilkTest, click **Enable Extensions** on the workflow bar.
The **Enable Extensions** dialog box opens.
- Select **ieexplore.exe** with the **Welcome to Green Mountain Outpost** caption from the list of running processes.



SilkTest automatically checks the WIN32 and xBrowser check boxes.

- Click **Enable Extensions**.
After a few moments, the WIN32 and xBrowser technology types display in the **Loaded Extensions** column in the dialog box.
- Click **Close**.

Related Topics

- [Quick Start Tutorial for Dynamic Object Recognition](#) on page 4

Creating a Base State for the GMO Web Application

You must enable extensions for the GMO web application before performing this step.

The base state makes sure that the application that you want to test is running and in the foreground. This ensures that tests will always start with the same application state, which makes them more reliable. In order to use the base state, it is necessary to specify what the main window looks like and how to launch the application that you want to test if it is not running. Creating a base state is optional. However, it is recommended as a best practice.

This task illustrates how to set up SilkTest to automatically launch Internet Explorer and open the GMO sample application before each testcase.

- Click **Set Recovery System** on the workflow bar.
The **Set Recovery System** dialog box opens. The text boxes are automatically populated with the GMO sample application information.
- Accept the default settings and click **OK**.
A **Recovery System** message box opens when the recovery system has been configured.

3. Click **OK**.

A new 4Test include file, `frame.inc`, opens that includes the default base state.

4. In `frame.inc`, add an application state that identifies the browser page in Internet Explorer and opens the GMO sample application.

An application state is the state you want your application to be in after the base state is restored but before you run one or more testcases. By creating an application state, you are creating reusable code that saves space and time. Furthermore, if you need to modify the Setup stage, you can change it once, in the application state routine.

a) Specify the application state name as: `appstate GMO() basedon DefaultBaseState`

b) To identify the browser page in Internet Explorer, type:

```
WINDOW browser = InternetExplorer.Find("../BrowserWindow")
```

 **Note:** `"../BrowserWindow"` is the XPath query string.

For more information about the syntax of XPath query strings, see "Supported XPath Subset" in the *SilkTest online help*.

c) To open the GMO web application, type:

```
browser.navigate("http://demo.borland.com/gmopost/")
```

This code enables you to use the default base state and the main window instead of Desktop to dynamically find child controls.

The code looks like the following:

```
appstate GMO() basedon DefaultBaseState
    WINDOW browser = InternetExplorer.Find("../BrowserWindow")
    browser.navigate("http://demo.borland.com/gmopost/")
```

5. Choose **File** ► **Save** to save the frame file.

Review your work to ensure that the `frame.inc` includes the following:

```
const wMainWindow = InternetExplorer

window BrowserApplication InternetExplorer
    multitag TAG_IS_OPEN_AGENT

    // The working directory of the application when it is invoked
    const sDir = "C:\<desktop directory>\Desktop"

    // The command line used to invoke the application
    const sCmdLine = ""C:\Program Files\Internet Explorer\iexplore.exe" "

    // The first window to appear when the application is invoked
    // const wStartup = ?

    // The list of windows the recovery system is to leave open
    // const lwLeaveOpen = {?}

appstate GMO() basedon DefaultBaseState
    WINDOW browser = InternetExplorer.find("../BrowserWindow")
    browser.navigate("http://demo.borland.com/gmopost/")
```

Related Topics

- [Quick Start Tutorial for Dynamic Object Recognition](#) on page 4

Creating a Testcase for the GMO Web Application

You must enable extensions and create a default base state for the GMO web application before performing this step.

To use dynamic object recognition, a testcase must use the `Find` or `FindAll` command and an XPath query string that identifies the object you want to test. In this testcase, we will use a `Find` command and an XPath query string to test clicking the **Enter GMO OnLine** button in the GMO sample application.

1. Choose **File > New**.

The **New File** dialog box opens.

2. Click **4Test script** and then click **OK**.

A new **4Test Script** window opens.

3. Create a testcase called `EnterGMOOnLine` that uses the GMO application state.

For more information about creating the GMO application state, see "Creating a Base State for the GMO Web Application."

The code looks like the following:

```
testcase EnterGMOOnLine () appstate GMO
```

4. To identify the browser page in which you want to launch the GMO application, type:

```
WINDOW browser = InternetExplorer.find("../BrowserWindow")
```

The browser page variable is a local variable and must be included in the testcase even though it is specified in the application state.

5. Identify the XPath query strings to include in the `Find` or `FindAll` functions in the test script.

You must use a supported construct to create the query string.

For more information about the syntax of XPath query strings, see "Supported XPath Subset" in the *SilkTest online help*.

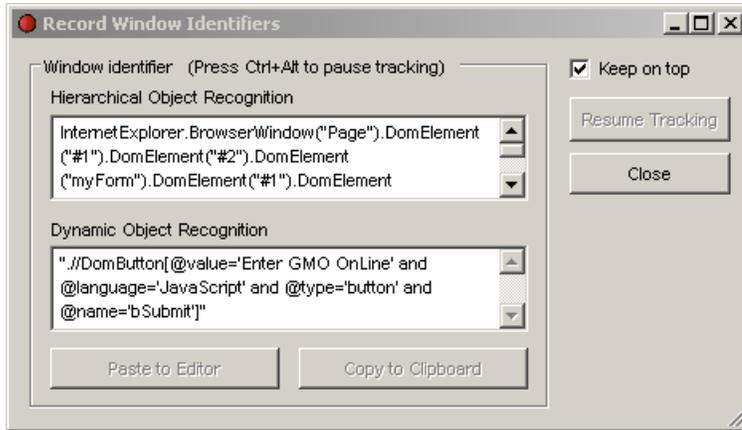
- a) Choose **Record > Window Identifiers**

Using the **Record Window Identifiers** dialog box ensures that the query string is valid. However, it does not verify whether the locator string is unique. We recommend that you ensure that the string is unique. Otherwise additional objects might be found when you run the test.

The **Record Window Identifiers** dialog box opens.

- b) In the GMO web application, move the mouse over the **Enter GMO OnLine** button.

The **Dynamic Object Recognition** text box displays the XPath query string for the button.



- c) Press Ctrl+Alt to capture the **Enter GMO OnLine** button in the **Dynamic Object Recognition** text box.
- d) Click the **Dynamic Object Recognition** text box to ensure that the text box is active and then click **Paste to Editor**.

The content of the currently active text box is copied into the **4Test Script** window.

The code looks like the following:

```
"/../DomButton[@value='Enter GMO OnLine' and @language='JavaScript'
and @type='button' and @name='bSubmit']"
```

6. In the **4Test Script** window, include the query string in the **Find** command.

The code looks like the following:

```
browser.Find("/../DomButton[@value='Enter GMO OnLine' and
@language='JavaScript' and @type='button' and @name='bSubmit']")
```

7. Remove all the attributes, except the `@name='bSubmit'` attribute.

In this case, the name attribute uniquely identifies the button, so no other attributes are necessary.

 **Note:** It is a good practice to include only the required attributes in a testcase.

8. To include the click button functionality, add `Click ()`.

The code looks like the following:

```
browser.Find("/../DomButton[@name='bSubmit']").Click ()
```

9. Choose **File** ► **Save** to save the script.

The **Save As** dialog box opens.

10. In the **File Name** text box, type `SimpleBrowserTest` and then click **Save**.

Review your work to ensure your test script looks like the following:

```
testcase EnterGMOonline () appstate GMO
WINDOW browser = InternetExplorer.find("/../BrowserWindow")
browser.Find("/../DomButton[@name='bSubmit']").Click ()
```

Related Topics

- [Quick Start Tutorial for Dynamic Object Recognition](#) on page 4

Running a Testcase for the GMO Web Application

1. In the **Files** tab in the **Project Explorer**, right-click **SimpleBrowserTest.t** and choose **Run Testcase**. The **Run Testcase** dialog box opens.
2. In the **Testcase** box, select **EnterGMOOnline** and then click **Run**. SilkTest runs the testcase and generates a results file.

Related Topics

- [Quick Start Tutorial for Dynamic Object Recognition](#) on page 4

Index

B

base state
 creating for GMO web application 6

P

projects
 creating xBrowser 4

Q

Quick Start tutorial
 running test 10
 testcase 8

R

Record Window Identifiers dialog box 8

S

SilkTest
 launching 4
 quick start tutorial 4

T

test case
 creating base state 6
testcase
 running 10
 sample script 8
tutorial
 quick start for dynamic object recognition 4

X

xBrowser
 creating project 4