

Silk Test 15.0

Silk4NET Tutorial

Micro Focus
575 Anton Blvd., Suite 510
Costa Mesa, CA 92626

Copyright © Micro Focus 2014. All rights reserved. Portions Copyright © 1992-2009 Borland Software Corporation (a Micro Focus company).

MICRO FOCUS, the Micro Focus logo, and Micro Focus product names are trademarks or registered trademarks of Micro Focus IP Development Limited or its subsidiaries or affiliated companies in the United States, United Kingdom, and other countries.

BORLAND, the Borland logo, and Borland product names 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.

2013-12-17

Contents

Getting Started with Silk4NET	4
Creating a Silk4NET Project	5
Adding a Silk4NET Test to a Project	6
Recording a Silk4NET Test	7
Running Silk4NET Tests	8
Enabling TrueLog	9
Analyzing Test Results	10

Getting Started with Silk4NET

Perform the following actions to use Silk4NET:

1. Create a Silk4NET project.
2. Add Silk4NET tests to your project. A project can include any combination of recorded tests and manually scripted tests.
3. Execute the tests.
4. Analyze the test results.

Creating a Silk4NET Project

1. Click **Silk4NET > New Project** or **File > New Project** . The **New Project** dialog box displays.
2. Under **Installed > Templates**, click **Visual Basic** or **Visual C#**, select **Test** and then select **Silk4NET Project**.
3. Type a name for the project into the **Name** field.
4. *Optional:* Type a name for the solution into the **Solution** field.
5. Click **OK**. The **Create a Silk4NET Test** dialog box opens.
6. Select how you want to create your Silk4NET test by clicking one of the following option buttons:

Record a Silk4NET test	Record actions and verifications against your application under test and generate a new test containing the recorded automation statements.
Create an empty Silk4NET test	Create an empty test that can be filled with automation statements later on.
7. Click **OK**. If you have selected to create an empty Silk4NET test, a new solution containing the Silk4NET project is created. Additionally, a Silk4NET test is created in the project with the following language-specific file name:
 - UnitTest1.vb
 - UnitTest1.cs
8. If you have selected to record a new Silk4NET test, the **Select Application** dialog box opens. Select the type of application that you want to test by clicking on a tab and then selecting the application in the list.
9. To test a Web application, type the URL of the Web application into the **Browse to URL** field.
10. Click **OK**. If you have selected an existing instance of Google Chrome on which you want to replay a test method, Silk4NET checks whether the automation support is included. If the automation support is not included, Silk4NET informs you that Google Chrome has to be restarted. The application and the **Recording** dialog box or the **Mobile Recording** dialog box open.



Note: You can also use the context menu in the **Solution Explorer** to add Silk4NET projects to an existing solution.

Adding a Silk4NET Test to a Project

You can only add Silk4NET tests to an existing Silk4NET or Test project. If no Silk4NET or Test project exists, create a Silk4NET or Test project before you try to create a Silk4NET test.

1. Click **Silk4NET > New Test** or **Project > Add New Item** .



Note: If your solution contains more than one Silk4NET projects, select the project to which you want to add the new test from the list in the **Project Selector**.

The **Add New Item** dialog box opens.

2. Under **Installed**, click one of the following:

- If your project is a Visual Basic project, click **Common Items > Silk4NET Test**.
- If your project is a Visual C# project, click **Visual C# Items > Silk4NET Test**.

3. Type a name for the test into the **Name** field and click **Add**. The **Create a Silk4NET Test** dialog box opens.

4. Select how you want to create your Silk4NET test by clicking one of the following option buttons:

Record a Silk4NET test Record actions and verifications against your application under test and generate a new test containing the recorded automation statements.

Create an empty Silk4NET test Create an empty test that can be filled with automation statements later on.

5. Click **OK**. If you have selected to create an empty Silk4NET test, a new solution containing the Silk4NET project is created. Additionally, a Silk4NET test is created in the project with the following language-specific file name:

- UnitTest1.vb
- UnitTest1.cs

6. Click **OK**. If you have selected an existing instance of Google Chrome on which you want to replay a test method, Silk4NET checks whether the automation support is included. If the automation support is not included, Silk4NET informs you that Google Chrome has to be restarted. The application and the **Recording** dialog box or the **Mobile Recording** dialog box open.

If you have selected to record the test, the recorded test is added to your project. If you have selected to add an empty test, an empty Silk4NET test is added to your project.



Note: You can also use the context menu in the **Solution Explorer** to add Silk4NET tests to your Silk4NET or Test project.

Recording a Silk4NET Test

1. Click **Silk4NET > New Test** or **Project > Add New Item** .



Note: If your solution contains more than one Silk4NET projects, select the project to which you want to add the new test from the list in the **Project Selector**.

The **Add New Item** dialog box opens.

2. Under **Installed**, click one of the following:

- If your project is a Visual Basic project, click **Common Items > Silk4NET Test**.
- If your project is a Visual C# project, click **Visual C# Items > Silk4NET Test**.

3. Type a name for the test into the **Name** field and click **Add**. The **Create a Silk4NET Test** dialog box opens.

4. Select **Record a Silk4NET test** and click **OK**.

5. Select the tab that corresponds to the type of application that you are testing:

- If you are testing a standard application that does not run in a browser, select the **Windows** tab.
- If you are testing a Web application or a mobile Web application, select the **Web** tab.

6. To test a standard application, select the application from the list.

7. To test a Web application or a mobile Web application, select one of the installed browsers or mobile browsers from the list.

You can specify the Web page to open in the **Browse to URL** text box. For the tutorial, select **Internet Explorer** and specify <http://demo.borland.com/InsuranceWebExtJS/> in the **Browse to URL** text box.

8. Click **OK**. If you have selected an existing instance of Google Chrome on which you want to replay a test method, Silk4NET checks whether the automation support is included. If the automation support is not included, Silk4NET informs you that Google Chrome has to be restarted. The application and the **Recording** dialog box or the **Mobile Recording** dialog box open.

9. Perform the interactions, which you want to record, with your application under test.

For additional information about recording a mobile Web application, see *Recording Mobile Web Applications*. For additional information about recording an action against a mobile device, see *Interacting with a Mobile Device*.

10. When you are finished with recording, click **Stop Recording**. The **Recording Complete** dialog box opens. From this dialog box, you can click **Playback** to replay the recorded test.

- If you are using Visual Studio 2010, you can also access the **Test View** in Visual Studio, where you can replay and manage your tests.
- If you are using Visual Studio 2012, you can also access the **Test Explorer** in Visual Studio, where you can replay and manage your tests.

The recorded interactions are added as a file to your project. The default file name of the generated file is `UnitTest<Index>.cs` or `UnitTest<Index>.vb`, depending on the default programming language of your project. For example, if you are recording the first test for a Visual Basic project, the name of the generated file is `UnitTest1.vb`



Note: You can also create a new project and record the new test into the new project.

Running Silk4NET Tests

This topic describes how you can run your Silk4NET tests in Visual Studio.

1. To view all the tests that are available in the selected project or solution:
 - In Visual Studio 2010, click **Test > Windows > Test View**.
 - In Visual Studio 2012, click **Test > Windows > Test Explorer**.
2. In the **Test View** or the **Test Explorer**, depending on which version of Visual Studio you are using, select the tests that you want to execute.
3. Right-click on your selection and click one of the following:
 - In Visual Studio 2010, click **Run Selection**.
 - In Visual Studio 2012, click **Run Selected Tests**.

To run all tests in the selected project or solution, click **Run All** in the **Test View** or the **Test Explorer**, depending on which version of Visual Studio you are using.

4. If you are testing a Web application and multiple browsers that are supported for replay are installed on the machine, the **Select Browser** dialog box opens. Select the browser and click **Run**.
5. When the test execution is complete, the **Playback Complete** dialog box opens. Click **Explore Results** to examine the results of the tests with TrueLog or click **OK** to close the dialog box.



Note: When you execute your tests, and Visual Studio starts the components that are needed for the execution of the tests, Visual Studio will clean up everything when the test execution is finished, terminating the Open Agent and all open browser windows.

Enabling TrueLog

For new Silk4NET scripts, TrueLog is enabled by default. To enable TrueLog for existing Silk4NET scripts, which are using the Visual Studio Unit Testing Framework, you have to replace the `TestClass` attribute of all test classes in the script with the `SilkTestClass` attribute.

To enable TrueLog:

1. Open the script that contains the test class for you want to enable TrueLog.
2. Add the `SilkTestClass` attribute to the test class.

The TrueLog is created in the `TestResults` sub-directory of the directory, in which the Visual Studio solution file and the results of the Visual Studio Unit Testing Framework are located. The Visual Studio solution file is the file in which the Silk4NET scripts are located. When the Silk4NET test execution is complete, a dialog box opens, and you can click **Explore Results** to review the TrueLog for the completed test.

Examples

To enable TrueLog for a class in a Visual Basic script, use the following code:

```
<SilkTestClass()> Public Class MyTestClass
  <TestMethod()> Public Sub MyTest()
    ' my test code
  End Sub
End Sub
```

To enable TrueLog for a class in a C# script, use the following code:

```
[SilkTestClass]
public class MyTestClass {
  [TestMethod]
  public void MyTest() {
    // my test code
  }
}
```

Analyzing Test Results

After running a test, you can review the test results and analyze the success or failure of the test run.

1. Run a Silk4NET test. When the execution is finished, the **Playback Complete** dialog box opens.
2. Click **Explore Results** to examine the results of the tests with TrueLog. Silk TrueLog Explorer opens.
3. Click through the results in Silk TrueLog Explorer.

Silk TrueLog Explorer captures a screenshot whenever a test fails.