

Silk Test 14.0 Silk4NET Web Edition

Release Notes

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

Copyright © Micro Focus 2013. 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.

2013-05-28

Contents

Release Notes	4
What's New in Silk4NET Web Edition	5
Image Recognition Support	5
Object Maps in Silk4NET	5
Mozilla Firefox Support	5
Google Chrome Support	5
Licensing Information	6
System Requirements and Prerequisites	7
Hardware Requirements	7
Platform Support	7
Adobe Flex Support	8
Java Support	8
Installation Instructions	9
Known Issues	10
General Issues	10
Silk4NET	11
Internet Explorer	12
Mozilla Firefox	12
Google Chrome	13
Resolved Issues	15

Release Notes

This file contains important information that might not appear in the Help. Read this file in its entirety.

For information about installation and licensing issues, refer to the [Installation Guide](#).

For the latest version of this document, see [Release Notes](#).

Navigate to supportline.microfocus.com for further assistance from Customer Care.

What's New in Silk4NET Web Edition

This section lists the significant enhancements and changes that were made for Silk4NET Web Edition.

Image Recognition Support

Image recognition methods now enable you to conveniently interact with test applications that contain highly customized controls, which cannot be identified using object recognition or text recognition. Instead of clicks with relative coordinates, you can now use *image clicks* to click on a specified image. Additionally, you can use the image representation of an object in the UI of your application under test to verify that the object exists and looks as expected.

Object Maps in Silk4NET

You can now use object maps in Silk4NET. An object map is a test asset that contains items that associate a logical name (an alias) with a control or a window, rather than the control or window's locator. Once a control is registered in an object map asset, all references to it in scripts are made by its alias, rather than by its actual locator name.

Mozilla Firefox Support

Silk Test now includes playback support for applications running in:

- Mozilla Firefox 19
- Mozilla Firefox 20
- Mozilla Firefox 21

Google Chrome Support

Silk Test now includes playback support for applications running in:

- Google Chrome 24
- Google Chrome 25
- Google Chrome 26
- Google Chrome 27

Licensing Information

Unless you are using a trial version, Silk Test requires a license.

The licensing model is based on the client that you are using and the applications that you want to be able to test. The available licensing modes support the following application types:

Licensing Mode	Application Type
Web	<p>Web applications, including Java-Applets.</p> <p> Note: If you are using a trial license, visit http://www.borland.com/store/silktestwebedition/silk4NET-webedition to buy a Web license.</p>
Web plus Flex	<p>Web applications, including the following:</p> <ul style="list-style-type: none"> • Adobe Flex • Java-Applets <p> Note: To upgrade your license to a Web plus Flex license, visit www.borland.com.</p>
Full	<ul style="list-style-type: none"> • Web applications, including the following: <ul style="list-style-type: none"> • Adobe Flex • Java-Applets • Adobe Flex • Java AWT/Swing • Java SWT and Eclipse RCP • .NET, including Windows Forms and Windows Presentation Foundation (WPF) • Rumba • Windows API-Based <p> Note: To upgrade your license to a Full license, visit www.borland.com.</p>
Premium	<p>All application types that are supported with a <i>Full</i> license, plus SAP applications.</p> <p> Note: To upgrade your license to a Premium license, visit www.borland.com.</p>

 **Note:** To use a Full or a Premium license, you require a different setup package. The information in this document applies only to the Web Edition of Silk4NET. For a detailed overview of the capabilities of the Full and Premium offering, see [Release Notes](#).

System Requirements and Prerequisites

The following section identifies the system requirements for installing and running Silk Test.

Hardware Requirements

We recommend the following hardware requirements:

System Area	Requirement
Processor	Intel or AMD DualCore processor running at 2 GHz
RAM	2 GB
Hard disk space	2 GB of free disk space

Platform Support

Platforms

Silk Test 14.0 is designed to run on the following platforms:

- Microsoft Windows XP SP3
- Microsoft Windows Vista SP2*
- Microsoft Windows 7
- Microsoft Windows 7 SP1
- Microsoft Windows 8

Technology Domains and Browsers

For the Open Agent, Silk Test 14.0 supports all applications that use the following browsers or are developed in:



Note: You can only test Adobe Flex applications with Silk4NET if you have a Web plus Flex license for Silk4NET. For additional information on the licensing modes, see *Licensing Information*.

Technology Type	Supported Version
Adobe Flex	4.0, 4.1, 4.5.1, 4.6
Mozilla Firefox (playback only)	17, 18*, 19*, 20*, 21*
Google Chrome (playback only)	24*, 25*, 26*, 27*
Internet Explorer	8, 9, 10
Java Applets	Java 1.6, Java 1.7

* Indicates a rapid release version of a browser. Such versions have a short product life-cycle and are frequently replaced by newer versions. Silk Test supports rapid release versions as long as the issuing vendor supports them. This means that the next Silk Test version will probably not continue to support the indicated versions, but might support newer versions.

Visual Studio Environment

You can integrate Silk4NET in Visual Studio 2010 Professional or later.

Adobe Flex Support



Note: You can only test Adobe Flex applications with Silk4NET if you have a Web plus Flex license for Silk4NET. For additional information on the licensing modes, see *Licensing Information*.



Note: Silk Test does not support testing child domain technologies, including Adobe Flex, for Web applications running in Google Chrome.

Silk Test supports Adobe Flash Player 10 or later.

Java Support

Silk Test supports Applets for Internet Explorer and Mozilla Firefox.



Note: You can test Applet contents that are embedded into HTML, but you cannot test external dialogs and frames.

Installation Instructions

For information about installation and licensing issues, refer to the [Installation Guide](#).

Known Issues

This section identifies Silk Test's known issues and their resolutions.

General Issues

When a remote desktop or remote desktop connection (RDC) is minimized, Silk Test does not function

When you connect through the remote desktop protocol (RDP) to a desktop, you take ownership of the desktop by attaching to the desktop with your mouse and keyboard. If the desktop is minimized without ownership of the desktop being released, any playback of mouse clicks or keystrokes is undefined.

The recommended configuration is to run Silk Test on a VMWare-server and connect through VMWare-client software. This allows replay to continue even if the client has been closed.

The Open Agent does not start when the Check Point firewall is installed

When you have a Check Point firewall or a Check Point ZoneAlarm firewall installed on your system, the Open Agent cannot be started, because the firewall interrupts the communication between the Agent and the infoservice.

To start the Open Agent, you have to uninstall the Check Point firewall from your system.

The `modifiers` parameter in the `domDoubleClick` method is ignored

You cannot specify the modifier in the overloaded `domDoubleClick` method. The modifier will not be double-clicked, although you have specified the parameter. The overloaded `domDoubleClick` method, which allows you to specify the modifier, is deprecated. To specify the modifier, you can use the `doubleClick` method, if you are using a client that supports an overloaded method with the `modifiers` parameter, or the `PressKeys` and `ReleaseKeys` methods.

The built-in spell checking in Microsoft Windows 8 might interfere with the replay of tests

The built-in spell checking in Microsoft Windows 8 can be enabled in applications like Internet Explorer 10.

If a word was incorrectly spelled during recording, and you replay typing this word, the spell checker will either mark it, or for commonly misspelled words will automatically fix it, which is the same behaviour a real user would get. If your tests were created on an operating system that did not include the spell checking feature, you might get unexpected results when replaying the tests on Microsoft Windows 8. To disable the spell checking, you can do the following:

1. Press **Windows Key + C**.
2. On the Charm bar, click **Settings**.
3. Select **More PC Settings**.
4. Select **General** to see the Spelling selections.



Note: These are system-wide settings, not settings specific to Internet Explorer 10.

5. Set **Autocorrect misspelled words** to off.
6. Set **Highlight misspelled words** to off.

When a .NET application is started from DevPartner Studio (DPS), Silk Test might not recognize it

To resolve the issue, perform the following steps:

1. Go to the Silk Test installation folder (by default, it's located at: C:\Program Files\Silk\Silk Test).
2. For Windows Forms applications, go to `ng\agent\plugins\com.borland.fastxd.techdomain.windowsforms.agent_<version number>`.
3. For Windows Presentation Foundation (WPF) applications, go to `ng\agent\plugins\com.microfocus.silktest.techdomain.wpf.agent_<version number>`.
4. In Notepad, open the file `plugin.xml`, and add the following line to the `<loadparameters>` section:

```
<param name="frameworkAssembly">mscorlib</param>
```
5. Go to the `ng\agent` folder in the Silk Test installation folder (by default, it's located at: C:\Program Files\Silk\Silk Test\ng\agent).
6. In Notepad, open `openagent.ini` and add `-clean` as the first line of the file.
7. Log out of the computer, and then log back in. Silk Test works as expected with the application that was started by DevPartner Studio.

The highlighting rectangle is out of place when recording clicks on an area of an image

When you record a click on a part of a complex image, for example an area map, the green highlighting rectangle does not highlight the appropriate area of the image. However, the click will be executed correctly during replay.

Silk4NET

If you do not have a TrueLog result file after test execution make sure that:

- A Silk4NET project which is created in Visual Studio 2010 is executed within a Visual Studio 2010 environment (also when executed via the command line or Team Foundation Server).
- A Silk4NET project which is created in Visual Studio 2012 is executed within a Visual Studio 2012 environment (also when executed via the command line or Team Foundation Server).
- The test class contains the `SilkTestClass` attribute.
- You have write access to the location where the TrueLog file will be created.

If you get the following error message during replay "Unit Test Adapter threw exception..." make sure that:

- A Silk4NET project which is created in Visual Studio 2010 is executed within a Visual Studio 2010 environment (also when executed via the command line or Team Foundation Server).
- A Silk4NET project which is created in Visual Studio 2012 is executed within a Visual Studio 2012 environment (also when executed via the command line or Team Foundation Server).

How to migrate from a Silk4NET project created in VS2010 to VS2012:

- Open the solution in VS2012.
- Change the target framework of your Silk4NET project(s) to .NET Framework 4.5.
- Remove the Reference `SilkTest.Ntf.VisualStudio2010.UnitTestingExtension` from your Silk4NET projects
- Add the reference `SilkTest.Ntf.VisualStudio2012.UnitTestingExtension` to your Silk4NET projects.

When you are using object maps, existing locators that do not start with a slash will no longer work

Locators that include only a class name and that do not start with a slash, for example `PushButton`, will no longer work if object maps exist. This issue might result in breaking existing scripts that were created in a Silk Test version prior to Silk Test 14.0. For the previous example the script will fail with the following error:

Identifier 'PushButton' was not found in the Object Map.

More complex locators that include more than a class name, for example `PushButton[@caption=OK]` will continue to work, even if object maps exist.

To fix this issue, add a `//` to the start of any such locator. For example, if the locator `PushButton` in the following code does no longer work:

```
PushButton button = mainWindow.find("PushButton");
```

Change the code to:

```
PushButton button = mainWindow.find("//PushButton");
```

Internet Explorer

Using Google toolbar interferes with recording Web applications

Using the Google toolbar with Internet Explorer 8 interferes with recording locators for Web applications.

Turn off the Google toolbar before you record Web applications.

Mozilla Firefox

Calls into applications using Adobe FlashPlayer do not properly synchronize in the latest versions of Mozilla Firefox

When you are using Mozilla Firefox with a recent Adobe FlashPlayer version, some calls might not synchronize properly. The following issues might occur:

- Mozilla Firefox might falsely recognize a running script as stalled and might display a confirmation dialog box asking whether you want to continue the execution of the script, even though the script is running properly.
- Typing characters might not work because `SetFocus` is no longer working correctly.
- The Adobe automation might return an old value although the UI already shows a new value.

If you face one or more of these issues with applications using Adobe FlashPlayer, turn off the ProtectedMode in Adobe FlashPlayer. For additional information, see <http://forums.adobe.com/thread/1018071> and read the information provided under *Last Resort*.

Text recognition might have the wrong offset in Mozilla Firefox 17

In Mozilla Firefox 17, text recognition might not use the correct offset. In such a case, the content of Mozilla Firefox is not redrawn.

Text recognition does not work with Mozilla Firefox 18 or later

In Mozilla Firefox 18 or later, text recognition does not work.

Google Chrome

Error in WaitForProperty on Google Chrome

You may get a `ReplayException` error when doing an action that causes navigation to a new page, and then use `waitForProperty` for synchronization on an object that exists on both pages. A workaround for the defect is to do the action, then a `Find` on an object which only exists on the new page, and then do the `WaitForProperty` if still necessary.

Silk Test does not recognize the Log In and Cancel buttons in the authentication dialog box of Google Chrome

Silk Test does not recognize the **Log In** and **Cancel** buttons in the authentication dialog box of Google Chrome. Use one of the following solutions to work around this limitation:

- Specify the username and the password in the URL of the website that you want to test. For example, to log in to the website `www.example.com/loginrequired.html`, use the following code:

```
http://myusername:mypassword@example.com/loginrequired.html
```

- Use `TypeKeys` to enter the username and password in the dialog box. For example, use the following code:

```
desktop.find("//Window[@caption='Authentication Required']/Control[2]").TypeKeys("myusername")
desktop.find("//Window[@caption='Authentication Required']/Control[1]").TypeKeys("mypassword<Enter>")
```



Note: `Control[2]` is the username field, and `Control[1]` is the password field. The `<Enter>` key at the end of the second `TypeKeys` confirms the entries in the dialog box.

OpenTab does not work properly with Google Chrome

When you use `OpenTab` while testing an application in Google Chrome, the automation interface of Google Chrome might stop responding and you might have to restart Google Chrome.

Locator recording in windows fails with Google Chrome

When you are testing a Web application in Google Chrome, locator recording in windows fails when multiple windows are open during application configuration in the Google Chrome instance, in which the application is running. If you close the other Google Chrome windows during application configuration, the error will not appear.

You cannot use Google Chrome to test Web sites that are using a Prototype JavaScript framework version previous to version 1.7

This is an issue in the Prototype JavaScript framework, where in framework versions previous to 1.7 arrays are incorrectly JSONified. This affects the internal `domAutomation` object of Google Chrome, on which the Google Chrome testing environment and the Silk Test implementation rely, and leads to a wrong behavior of Google Chrome during testing. This issue has been fixed in Prototype JavaScript version 1.7 or later.

To use Google Chrome to test web sites that are using the Prototype JavaScript framework, update Prototype JavaScript to version 1.7 or later.

Background applications in Google Chrome prevent automation support from loading

When you want to test a Web application with Google Chrome and the **Continue running background apps when Google Chrome is closed** check box is checked, Silk Test cannot restart Google Chrome to load the automation support.

Resolved Issues

The following issues have been resolved:

Issue #	Description
RPI 595210	Automation execution fails at <code>typeKeys</code> method.
RPI 1085703	Silk Test Workbench cannot connect to the application under test if the application uses a different locale than the rest of the system.
RPI 1085706	When closing and re-opening an application which is connecting to multiple agents or is using mixed agent scripts, Silk Test Classic throws a 'Bad channel to agent call' error.
RPI 1085839	Silk Test Workbench crashes when complex nested script references occur.
RPI 1085884	Silk Test Classic: <code>SYS_GetFileContents()</code> method for the Open Agent does not support unicode or UTF-8 characters.
RPI 1085962	The Difference Viewer does not display the actual value, but only the expected value.
RPI 1086020	Silk Test Classic: Registry manipulation methods shipped with the product do not work correctly with the Classic Agent.
RPI 1086153	Silk Test Classic: The Open Agent does not recognize <code>FlexTitleWindow</code> controls in Adobe Air applications.
RPI 1086205	Silk Test Workbench crashes when scripts are referenced multiple times.
RPI 1086286	Cannot identify windows by <code>classname</code> .
RPI 1086363	Java SWT: Controls that have both a single and a double quote in an attribute value are recorded with an empty locator.
RPI 1086479	On some non-English systems, the Information Service is deleted during the installation of Silk Test.
RPI 1086563	Silk Test Classic crashes while scripting.
RPI 1086745	Silk Test Workbench: Only one browser hooking configuration is considered for a Visual Test.
RPI 1087087	Silk Test Classic crashes when the agent name is set to <code>(none)</code> .
RPI 1087325	Browser objects are not scrolled into view.
RPI 1087457	Java AUT freezes while recording or playing back tests with the Open Agent.
RPI 1087461	The <code>OnClick</code> attribute is not being recorded for Google Chrome.
RPI 1087511	The properties <code>AllCellTexts</code> and <code>AllCellValues</code> are added to <code>JTable</code> .
RPI 1087650	Unable to read hidden fields when using Google Chrome.
RPI 1087697	Allow null values for pointers and suitable size types in DLL calling.
RPI 1087733	<code>JTree</code> node recognition in Silk Test Workbench fails if it has a trailing space.
RPI 1087734	Null pointer exception when starting an Adobe Flex application that is compiled with an exception handler.
RPI 1088056	Silk Test does not recognize a WPF application correctly when the application is started as a standalone application.

Issue #	Description
RPI 1088145	Cisco VPN impacts Open Agent communication (improvements to compensate).
RPI 1088225	Performance issues when recording or replaying a JTextField of a Java application.
RPI 1088481	Unhandled exception within a Silk Test Workbench script.
RPI 1088549	Unhandled access violation in Borland.SilkTest.WinForms.NativeTechDomain.4_x86.
DE3904	SAP eCATT: No log is created when a compilation error occurs in a .NET script.
DE3988	Rumba screen verifications do not work correctly.
DE4056	
DE4061	
DE3931	Text fallback with Mozilla Firefox and Microsoft Windows 8.
DE4001	Item identifiers are not normalized. The trailing and leading white spaces are different depending on the browser version.
DE4031	SAP eCATT: Load BLOB for Japanese asset failed by XML parse error.
DE4034	Some owner drawn menu items are recognized with glyphs as caption.
DE4059	Silk Test Workbench: F8/Step Into is not working with breakpoints.
DE4110	Silk4NET: The Open Agent blocks the Team Foundation Server output directory.
DE4116	TypeKeys in Google Chrome do sometimes not replay all keys.
DE4254	Truelog screenshots are dependent on color quality.
DE4955	Recording crashes Internet Explorer 10.
TA19985	Opening a tab fails, even though the tab is successfully opened.
TA20248	Unify whitespace of textcontents property across all browsers.
TA22051	Silk Test Classic crash reported by Windows Error Reporting Service.
TA26707	Improvements for retrieving properties in Internet Explorer.