Silk Test 14.0

Release Notes

Contents

Silk Test 14.0 Release Notes

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

For the latest version of this document, see *http://supportline.microfocus.com/productdoc.aspx*.

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

What's New in Silk Test

This section lists the significant enhancements and changes that were made for Silk Test.

Silk Test 14.0 supports the following features:

All Clients

The following features are new for all clients:

Mozilla Firefox Support

Silk Test now includes playback support for applications running in:

- Mozilla Firefox 17
- Mozilla Firefox 18
- Mozilla Firefox 19
- Mozilla Firefox 20
- Mozilla Firefox 21

Google Chrome Support

Silk Test now includes playback support for applications running in:

- Google Chrome 22
- Google Chrome 23
- Google Chrome 24
- Google Chrome 25
- Google Chrome 26
- Google Chrome 27

Rumba 9.0 Support

Silk Test now supports Rumba 9.0.

SAPGUI Client 7.30

Silk Test now supports SAPGUI client 7.30.

Invoking Static Methods of the Application Under Test

In addition to dynamically invoking any method that the underlying object in the application under test defines, for a Windows Forms or a WPF control you can now dynamically invoke all static methods that are defined by the object in the application under test, as well as all user defined static methods.

Usability Enhancements

You can now define whether you want to match attribute names case-sensitive or not

Use the new OPT_LOCATOR_ATTRIBUTES_CASE_SENSITIVE option to specify whether you want to match attribute names case-sensitive or not. The default setting is case insensitive.

New Borland YouTube Channel

You can now access the new *Borland Channel on YouTube* to see videos about Silk Test.

Silk Test Knowledge Base on Silk Test Community

The *Silk Test Knowledge Base* is now located in the Silk Test community. If you have a question about Silk Test, you can now use the advanced search functionality in the Silk Test community to find an answer.

Silk Test Workbench

The following features are new for Silk Test Workbench:

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.

Adding Code to the Application Under Test

When you are testing a WPF or Windows Forms application, you can now add code, with static methods, that interacts with the actual controls in your application under test (AUT). You can inject the code into the AUT during script playback or you can compile the code into the AUT. Then you can call the code from your script during playback to ease the interaction with custom controls.

Microsoft SQL Server 2012 SP1 Support

Silk Test Workbench now supports Microsoft SQL Server 2012 Service Pack 1 (SP1).

Opening an Asset from a Script

When you are editing a script, you can open an asset by right clicking it and selecting **Open Asset.** This will open the asset in the GUI.

Locator Spy Displays All Attributes

The locator attributes table of the **Locator Spy** now displays all attributes that you can use in the locator. For Web applications, the table now also includes any attributes that you have defined to be ignored during recording.

Filter Verification Assets

You can now filter verification assets from the **Result Window** toolbar.

Silk Test Classic

The following features are new for Silk Test Classic:

Enhanced Workflow for Configuring Applications

While creating an application configuration with the Open Agent, you can specify whether you want to additionally create a base state for the application under test or only hook the application under test. Not creating the base state can add efficiency and transparency to complex testing scenarios and when testing multiple applications.

Silk4NET

The following features are new for Silk4NET:

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.

New Silk4NET Start Page

When you start Silk4NET, the new start page provides an entry point from where you can comfortably discover the functionality that is available in Silk4NET. The start page enables you to continue by choosing one of the following options:

- View a list of new features.
- Directly access common Silk4NET functionality.
- View videos on commonly used features.
- Open sample projects.
- Access Help topics which describe how you can start working with Silk4NET.
- View the linked resources for additional support and training.
- View the latest forum entries and blog posts on the Silk Test community.

The start page remains open until you close it or deselect the option to **Show page on startup**.

Opening an Asset from a Script

When you are editing a script, you can open an asset by right clicking it and selecting **Open Asset.** This will open the asset in the GUI.

Adding Code to the Application Under Test

When you are testing a WPF or Windows Forms application, you can now add code, with static methods, that interacts with the actual controls in your application under test (AUT). You can inject the code into the AUT during script playback or you can compile the code into the AUT. Then you can call the code from your script during playback to ease the interaction with custom controls.

Locator Spy Displays All Attributes

The locator attributes table of the **Locator Spy** now displays all attributes that you can use in the locator. For Web applications, the table now also includes any attributes that you have defined to be ignored during recording.

Web Based Training for Silk4NET

A new cost-free Web-based training for Silk4NET is available from the *Micro Focus Training Store*.

One-Click Browser Switching in Silk4NET

To change the browser that is used for test replay and locator recording, you can just click on the icon of the browser that you want to use.

What's New in Silk4J

The following features are new for Silk4J:

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 Silk4J

You can now use object maps in Silk4J. 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.

Opening an Asset from a Script

When you are editing a script, you can open an asset by right clicking it and selecting **Open Asset.** This will open the asset in the GUI.

Adding Code to the Application Under Test

When you are testing a WPF or Windows Forms application, you can now add code, with static methods, that interacts with the actual controls in your application under test (AUT). You can inject the code into the AUT during script playback or you can compile the code into the AUT. Then you can call the code from your script during playback to ease the interaction with custom controls.

Locator Spy Displays All Attributes

The locator attributes table of the **Locator Spy** now displays all attributes that you can use in the locator. For Web applications, the table now also includes any attributes that you have defined to be ignored during recording.

Web Based Training for Silk4J

A new cost-free Web-based training for Silk4J is available from the *Micro Focus Training Store*.

One-Click Browser Switching in Silk4J

To change the browser that is used for test replay and locator recording, you can just click on the icon of the browser that you want to use.

Silk Test Product Suite

The Silk Test product suite includes the following components:

- Silk Test Workbench – Silk Test Workbench is the native quality testing environment that offers .NET scripting for power users and easy to use visual tests to make testing more accessible to a broader audience.
- Silk4NET – The Silk4NET Visual Studio plug-in enables you to create Visual Basic or C# test scripts directly in Visual Studio.
- Silk4J – The Silk4J Eclipse plug-in enables you to create Java-based test scripts directly in your Eclipse environment.
- Silk Test Classic – Silk Test Classic is the traditional, 4Test Silk Test product.
- Silk Test Agents – The Silk Test Agent is the software process that translates the commands in your tests into GUI-specific commands. In other words, the Agent drives and monitors the application you are testing. One Agent can run locally on the host machine. In a networked environment, any number of Agents can run on remote machines.

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 able to test. The available licensing modes support the following application types:

Licensing Mode	Application Type

Full	• Web applications, including the following:
	    • 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
	
	Note: To upgrade your license to a Full license, visit *www.borland.com*.

Premium	All application types that are supported with a *Full* license, plus SAP applications. **Note:** To upgrade your license to a Premium license, visit *www.borland.com*.

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

Software Requirements

To install and execute Silk Test, the following software needs to be installed:

* Microsoft .NET Framework 4 (Full setup)

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 Server 2008
* Microsoft Windows Server 2008 R2
* Microsoft Windows 7
* Microsoft Windows 7 SP1
* Microsoft Windows 8
* Microsoft Windows Server 2012

Technology Domains and Browsers

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

Technology Type	Supported Version
Adobe Flex	4.0, 4.1, 4.5.1, 4.6
Mozilla Firefox (playback only)	10*, 17, 18*, 19*, 20*, 21*
Google Chrome (playback only)	22*, 23*, 24*, 25*, 26*, 27*
Internet Explorer	8, 9, 10

Technology Type	Supported Version
Java AWT/Swing	Java 1.6, Java 1.7
Java SWT	3.6, 3.7
SAP	SAPGUI client 7.10, SAPGUI client 7.20, SAPGUI client 7.30
Silverlight*	3 (Silverlight Runtime 4), 4 (Silverlight Runtime 4 and Silverlight Runtime 5)
Rumba	8.1, 8.2, 8.3, 9.0
Win 32	any
WinForms	.NET 3.0, 3.5, 3.5 SP1, 4.0, 4.5
WPF	.NET 3.5 SP1, 4.0, 4.5

Databases

Silk Test 14.0 supports the following databases for Silk Test Workbench asset storage:

- Microsoft SQL Server 2005 (SP3) + Express*
- Microsoft SQL Server 2008 (SP2) + Express
- Microsoft SQL Server 2012
- Microsoft SQL Server 2012 SP1
- Microsoft Access 2000 (as a single-user database)

Eclipse Environment

You can integrate Silk4J in Eclipse 3.7.2 or later. Version 3.7.2 is shipped with Silk Test.

Visual Studio Environment

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

* These operating systems, features, and integrations are planned to be no longer supported in future Silk Test versions. For additional information, see *End-Of-Life (EOL) Components*.

Adobe Flex Support

Silk Test supports testing Adobe Flex 4.0, 4.1, 4.5.1, and 4.6 using all supported browsers and Adobe AIR 2.0 or later (built with Adobe Flex 4.x).

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 the following Java components for the Open Agent:

- SWT 3.6 and 3.7 standalone and Rich Client Platform (RCP) applications (SWT standalone applications executed as an applet in the browser not supported)
- Java version 1.6 and 1.7
- AWT
- Swing (including Java Foundation Classes)
- Applets (Internet Explorer and Mozilla Firefox)

Silk Test Classic DSN Support

Silk Test Classic's Data Driven Workflow supports:

- Text files and comma separated value files (*.txt and *.csv files)
- MS Excel
- MS Access
- MS SQL Server
- Oracle (partial support)
- SyBase SQL Anywhere

Silk Test Classic accesses the preceding databases using ODBC and therefore supports the versions of those databases that have a working ODBC driver.

Note: The Silk Test Classic **Select Data Source** dialog box allows you to choose either the Silk DDA Excel or the Segue DDA Excel data source. For new data driven testcases, choose the Silk DDA Excel data source. Choose the Segue DDA Excel data source for backward compatibility. This allows existing .g.t files that reference Segue DDA Excel to continue to work.

Installation Instructions

For information about installation and licensing issues, refer to the installation guide, available from *http://supportline.microfocus.com/productdoc.aspx*.

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.

Testing Silverlight applications with Mozilla Firefox does not work well on some machines

The Silverlight support in Silk Test might not work well on some machines when used in Mozilla Firefox. A deadlock can occur if the Mozilla Firefox plugin process calls the host Mozilla Firefox process at the same time that the host Mozilla Firefox process calls the plugin process. For additional information, see *https:// bugzilla.mozilla.org/show_bug.cgi?id=686861* or *https://bugzilla.mozilla.org/show_bug.cgi?id=558986*. This known issue might occur in Silk Test because Silk Test is loading a .Net framework into the plugin process to test Silverlight. In this case, Silk Test cannot control when the .Net framework triggers a garbage collection, which calls into the host Mozilla Firefox process when freeing COM objects.

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 replying 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">mscoree.dll</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.

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.

Google Chrome 21, 26, 27: Invalid or missing Accessibility objects in dialog boxes

When you are recording locators for a Web application in Google Chrome, some Accessibility objects in
dialog boxes might not be recognized or the recorded hierarchy of the objects might be invalid. This issue
applies to all Javascript dialog boxes, for example prompt and the HTTP authentication dialog box. Existing
tests, which are using these dialog boxes, might fail when they are replayed on Google Chrome.

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.

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.

Microsoft Silverlight Applications

Some Microsoft Silverlight applications cause Internet Explorer to hang when interacting with Silk Test. On
32-bit platforms, refer to MS KB 2564958 (an update to Active Accessibility) to help prevent the issue.

Locators recorded with Silk Test versions prior to Silk Test 13.5 might no longer work in Internet Explorer

In Silk Test 13.5, we have adapted the normalization of white spaces of the `textContents` attribute in
Internet Explorer. This change was made to improve the cross-browser capabilities of Silk Test, and might

affect locators which rely on the `textContents` attribute, and which are used in scripts that were recorded with releases prior to Silk Test 13.5.

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.

SAP Applications

`HierarchyHeaderWidth` and `ColumnOrder` Properties of the SAPTree Class are write only

Other than the automation documentation indicates, the `HierarchyHeaderWidth` and `ColumnOrder` properties of the `SAPTree` class are write only and cannot be read.

Ensure that your scripts use write rather than read with these properties.

`GetColumnIndexFromName()` of the SapTree Class May Fail with an "unspecified error"

`GetColumnIndexFromName()` of the `SapTree` class may fail with an "unspecified error". This is a known issue in SAP automation.

Check the SAP web site to see if the issue has been resolved.

Calling the `Select()` method of the SAPTree Class on a Context menu item may fail

Calling the `Select()` method of the `SAPTree` class on a Context menu item may fail.

Call `SelectContextMenuItem` on the parent control instead. This problem is a known issue in the SAP automation.

The position property for a horizontal scrollbar always returns 1

The position property for a horizontal scrollbar always returns 1. This is a known issue in SAP automation.

Check the SAP web site to see if the issue has been resolved.

The SAPNetPlan class is not supported

This issue will be resolved in a future release.

Replay error occurs when executing an SAP script in fast mode

In certain cases, if you record an SAP test using the Silk Test Recorder and then replay it using the **Replay speed: Fast** setting, an error occurs. The error says, `The data necessary to complete this operation is not yet available.`

Change the replay speed to a slower setting. Or, change the script to use SAP automation to replay the problematic action instead of xBrowser. For example, you might change the action from `DomLink.Select` to `SapHTMLViewer.SapEvent`.

The method getCurrentRow returns a wrong value with SAPGUI client 7.30

If you use SAPGUI client 7.30 and you call the method `getCurrentRow`, the method might falsely return -1 instead of the row number.

The method resizeWorkingPane is not working correctly with SAPGUI client 7.30

If you use SAPGUI client 7.30 and you call the method `resizeWorkingPaneEx`, the method will not resize the `workingPane` and calling `getSapWindow().getWidth()` will return a wrong value for the window width.

Silk Test Classic

Ensure that the agent.exe and partner.exe processes have been killed before attempting a Silk Test Monitor execution

When running Silk Test monitors or GUI-level tests, `Agent.exe` or `Partner.exe` may not close correctly after a monitor execution and the temporary results folder may remain opened. When this happens during a monitor execution, all subsequent monitors fail with the error `LoadtestController: 3302 - The results directory could not be cleared` and do not clear.

Kill the `Agent.exe` or `Partner.exe` process manually on the execution server or create an Action Essential which will automatically kill the processes whenever needed.

Unknown behavior occurs if a window class is declared without a tag

If a customer implements a window class in Silk Test without a tag, unknown errors and possible crashes occur. For example, in the following code, the `tag` line is mandatory.

```
[-] winclass MyDialogBox : DialogBox
 [ ] tag "[DialogBox]"
 [ ] locator "//DialogBox[@caption='Check Box']"
```

If the tag line is present, the code works as it should. However, if the tag line is missing, this leads to unknown errors and possible crashes. For example, the following code is problematic:

```
[-] winclass MyDialogBox : DialogBox
 [ ] locator "//DialogBox[@caption='Check Box']"
```

Silk Test Workbench

Replaying a Visual Test on a Large Site

You may encounter performance problems when running a visual test on a large site when using xBrowser. In order to avoid this, set the option **Playback** > **Results** > **VisualTest** > **ControlCapture** to `No`.

Message boxes display in the background when a script plays back (31314)

If you include a message box statement, such as `MsgBox ("Hello")` in .NET script, it displays in the background when you play back the script.

Include `MsgBoxStyle.MsgBoxSetForeground` in the `MsgBox` statement to have the message box display in the foreground. For example:

```
MsgBox ("Hello", MsgBoxStyle.MsgBoxSetForeground)
```

Installer fails to install SQL Server Native Client and register Silk Test Workbench as a COM server

If the operating system performs an automatic update or the user initiates a Windows update during or before installation of Silk Test, the installer fails to install SQL Server Native Client and register Silk Test Workbench as a COM server.

Both of these problems affect Silk Central Test Manager (SCTM) integration, since SCTM uses the native client driver installed by SQL Server Express to create Silk Test Workbench DSNs. Furthermore, SCTM needs Silk Test Workbench registered as a COM server in order to use its COM interface for automation.

Choose one of the following solutions:

1. Do not install Silk Test while running a Windows update.
2. Ensure that Windows updates do not install automatically on your machine.
3. Install any pending Windows updates prior to installing Silk Test and reboot the system before installing Silk Test.

Modifying the hot key combination to include multiple modifiers causes additional keys to be recorded

If you configure a hot key for insert verification or start/stop record that contains multiple modifiers, occasionally the modifier press key action is recorded. For instance, if you specify a hot key combination of `Alt+Ctrl+F9`, `Alt` may be recorded as a press key action instead of being ignored as part of the hot key combination. This is a problem during playback because the appropriate release key actions are missing.

Manually remove the actions related to the modifiers, or only use one hot key modifier.

SQL Server DSN does not work on 64-bit systems

A 64-bit DSN cannot be used with Silk Test Workbench. To create a DSN for a 64-bit machine, click **Start** > **Silk** > **Silk Test** > **Administration** > **Data Sources (ODBC)** and create a 32-bit DSN. You can also use the WOW64 tools located at `C:\WINDOWS\SysWOW64\odbcad32.exe`.

Silk Test Workbench requires a network connection to start

Silk Test Workbench requires a network connection to start. If no network connection is available, a `"Failed to connect to OpenAgent"` error occurs. Other clients, such as Silk Test Classic and Silk Test Recorder, do not require a network connection to start.

Ensure that a network connection is available before starting Silk Test Workbench.

SQL Server Express might not install on Windows Vista SP1 machines

On Windows Vista SP1 machines, ensure that the full version of .NET 3.5 SP1 is installed if you plan to install SQL Server 2008 Express. To install the full version of .NET 3.5 SP1, download it from the Microsoft Web site.

Identify from Screen Preview creates an incorrect object map item for a migrated database

After a database is migrated from Silk Test Workbench 2010 to Silk Test Workbench 2010 R2, using the **Identify from Screen Preview** creates incorrect object map items.

Use the **Update Screen** command to recapture the screen before using **Identify from Screen Preview**.

When a child script is called from a parent script, the child script can access the object map out of scope

If a parent and child script reside in different projects and use different object maps, the child script runs successfully even though the object map is out of scope for the child script.

Since the script runs successfully, there is no resolution necessary. However, this behavior will not work in future releases.

When multiple users attempt to simultaneously edit the same asset on a SQL 2008 database, issues with the asset locking mechanism might occur

To test for the problem, open an asset, like a visual test, for editing in the Silk Test Workbench. While the asset is open, attempt to open the same asset from a different instance of Silk Test Workbench. If the problem occurs, either the second instance of Silk Test Workbench will be able to simultaneously edit the same asset, or the second instance will receive a message with incomplete data. The user name and machine name will both be blank.

In SQL Server, allocate database admin rights to the SQL database users that use that database.

Unexpected error while exporting assets in a SQL database

The unexpected error, `Conversion failed when converting from Character string to Unique identifier`, occurs when exporting assets in a SQL database.

Import the exported database and press OK if any error messages occur. The import finishes successfully and the assets are available for use.

Unexpected behavior while using an Access database

Compact the database using the Silk Test **Database Maintenance** tool. Click **Start** > **Programs** > **Silk** > **Silk Test** > **Administration** > **Database Maintenance**. Then, open the database and click **Tools** > **Compact Database**.

Silk Test Workbench with UAC enabled during configuration cannot connect to an SQL Server

You cannot connect to an SQL server during configuration, when you are using Silk Test Workbench without administrator rights, and with UAC enabled.

To connect to the SQL server during configuration, you have to start Silk Test Workbench with the **Run as administrator** option.

Silk Test Workbench might crash when the connection to a remote SQL database is lost

When you are using a remote SQL database, and that database disconnects from Silk Test Workbench, for example because of a server restart, Silk Test Workbench might crash when trying to communicate with the database.

To reconnect with the database, restart Silk Test Workbench.

The replay of a visual test might get very slow if the Control capture option is set

Set the Control capture option to **No** to improve playback performance.

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");
```

Silk4J

The Silk4J menu does not open correctly in Eclipse 4.1

When you click on the Silk4J menu button in Eclipse 4.1, the menu items are not displayed. This is an Eclipse issue that is fixed for Eclipse 4.2. For additional information, see *https://bugs.eclipse.org/bugs/ show_bug.cgi?id=367159*.

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");
```

Resolved Issues

The following issues have been resolved:

Issue #	Description
RPI 595210	Automation execution fails at `typeKeys` 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: SYS_GetFileContents() 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 FlexTitleWindow controls in Adobe Air applications.
RPI 1086205	Silk Test Workbench crashes when scripts are referenced multiple times.
RPI 1086286	Cannot identify windows by `classname`.
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 `(none)`.
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 `OnClick` attribute is not being recorded for Google Chrome.
RPI 1087511	The properties `AllCellTexts` and `AllCellValues` are added to `JTable`.
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	`JTree` 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 DE4056 DE4061	Rumba screen verifications do not work correctly.
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.

End-Of-Life (EOL) Components

This section lists the operating systems, features, and integrations that will no longer be supported in the current or future Silk Test versions.

EOL Components for Silk Test 14.0

The following operating systems, features, and integrations are no longer supported in Silk Test:

- Windows-based applications (on the Classic Agent)
- Web applications (on the Classic Agent)
- ActiveX/Visual Basic applications (on the Classic Agent)
- Microsoft Accessibility (on the Classic Agent)
- Mozilla Firefox 3.6
- Java SWT 3.5
- Adobe Flex 3
- Internet Explorer 7
- Eclipse versions less than 3.7.2

The following browser versions are rapid release versions. These versions have a short product life-cycle and are frequently replaced by newer versions. Silk Test supports these versions as long as the issuing vendors do. Rapid release versions have a shorter grace period in Silk Test once EOL is declared.

- Mozilla Firefox 12
- Mozilla Firefox 13
- Mozilla Firefox 14
- Google Chrome 19
- Google Chrome 20
- Google Chrome 21

> **Note:** Silk Test applies the normal life-cycle policy to support an Extended Support Release (ESR) of Mozilla Firefox, for example Mozilla Firefox 17.

EOL Components for the Next Silk Test Release

The following operating systems, features, and integrations will no longer be supported in the next Silk Test release:

- Microsoft SQL Server 2005

The following browser versions are rapid release versions. These versions have a short product life-cycle and are frequently replaced by newer versions. Silk Test supports these versions as long as the issuing vendors do. Rapid release versions have a shorter grace period in Silk Test once EOL is declared.

- Mozilla Firefox 18
- Mozilla Firefox 19
- Mozilla Firefox 20
- Mozilla Firefox 21
- Google Chrome 22
- Google Chrome 23

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

Note: Silk Test applies the normal life-cycle policy to support an Extended Support Release (ESR) of Mozilla Firefox, for example Mozilla Firefox 17.

EOL Components for the Silk Test Release After the Next Release

The following operating systems, features, and integrations will no longer be supported in the Silk Test release after the next release:

- Silk Test Recorder. The standalone Silk Test Recorder will no longer be shipped with Silk Test. This change does not affect the internal recording capabilities of any Silk Test client.
- Microsoft Windows Vista
- Mozilla Firefox 10
- Microsoft Silverlight