



# **SilkTest®**

## **SilkTest 2010 Release Notes**

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

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

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

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

**All other marks are the property of their respective owners.**

# Contents

- SilkTest 2010 Release Notes.....4**
- SilkTest Product Suite.....4
- What's New in SilkTest Workbench.....4
- What's New in Silk4NET.....6
- What's New in Silk4J.....6
- What's New in SilkTest Classic.....7
- System Requirements and Prerequisites.....8
  - Hardware Requirements.....8
  - Platform Support.....8
  - Tested Configurations.....10
  - Adobe Flex Support.....10
  - Java Support.....11
  - SilkTest Classic DSN Support.....11
- End-Of-Life (EOL) Components.....12
- Known Issues.....12
- Installation Instructions.....12

# SilkTest 2010 Release Notes

## July 2010

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 [SilkTest Installation Guide](#).

Navigate to [supportline.microfocus.com](http://supportline.microfocus.com) for further assistance from Customer Care.

## SilkTest Product Suite

The SilkTest product suite includes the following components:

- SilkTest Workbench – SilkTest Workbench is the new, native quality testing environment that offers .NET scripting for power users and innovative storyboard-based visual tests to make testing more accessible.
- 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.
- SilkTest Recorder – SilkTest Recorder enables you to record and replay tests using a GUI and then export those tests to SilkTest Classic, Silk4J, or Silk4NET.
- SilkTest Classic – SilkTest Classic is the traditional, 4Test SilkTest product.
- SilkTest Agents – The SilkTest 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.

## What's New in SilkTest Workbench

### Visual Tests

SilkTest Workbench lets you quickly record and playback visual tests. Visual tests comprise the basic building blocks of an automated testing solution. SilkTest Workbench uses visual tests to mimic the actions that are performed while testing an application. With visual tests, you can:

- Record visual tests automatically, manually edit them, or use a combination of automatic and manual tests.
- Visually modify or create test steps and access step properties without having to access the live test application.
- Use the Visual Navigator to view information displayed in four panes that collectively provides a comprehensive view of each step in a visual test.

The Screen Preview, Storyboard, and Properties panes are synchronized with the Test Steps pane and display information specific to a selected step in the Test Steps pane. In this way, you can easily view all aspects of a step by selecting a step in the Test Steps pane, and then viewing information about the step in the other panes.

- Use the **Identify Object** dialog box to identify the locators for an application's objects and the properties for each object.
- Create test logic, including conditions, expressions, verifications, and error handling using the Test Logic Toolbox.
- Add synchronization steps to delay or wait for objects during playback.
- Configure the settings that dictate how and what actions are recorded and replayed.
- Review detailed test results, which provide a step-by-step record of the actions executed.

### Embedded Scripting Language

SilkTest Workbench's scripting language is Microsoft's Visual Basic, a robust programming language that gives you total control over any application running in the Microsoft .NET framework. .NET scripts contain the functionality of a high-level programming language as well as features designed specifically for software control and testing. With scripts, you can:

- Record scripts automatically, manually edit them, or use a combination of automatic and manual tests.
- Leverage .NET scripting features to make manual scripting easier. These features include:
  - **Identify Objects** dialog box – View how locators identify the GUI objects within the test application and which properties are available for those objects.
  - Auto-completion and Syntactical Assistance – These facilitate manual coding. Using auto-completion and syntactical assistance, you can automatically access, display, and enter .NET language elements, including keywords, enumerations, Boolean values, code syntax, available methods, and available .NET properties for any object.
- Create scripts to cover individual parts of a whole test flow, then use a driver script to call each part.
- Add .NET assemblies from the .NET framework as references for a .NET script.
- Add a file that the script references to logically segment code for distinct purposes.
- Compile and run scripts and use the **Output** window to review automation code and append additional user-defined output to provide supplemental information about a script.
- Find and replace code in scripts to speed script modification.
- Use bookmarks to navigate to key portions of the script.

### Supported Technologies

SilkTest Workbench provides support for testing applications developed in a wide variety of development tools including:

- Adobe Flex
- Java SWT
- SAP
- Windows API-based client/server (Win32)
- Windows Forms
- Windows Presentation Foundation (WPF)
- xBrowser (Web applications)

### Integration with Additional Micro Focus Products

The SilkTest product suite includes two plugins, Silk4NET and Silk4J, which both work with SilkTest Recorder. Additionally, SilkTest Workbench works with SilkCentral Test Manager (SCTM).

- The Silk4NET Visual Studio plug-in enables you to create Microsoft Visual Basic or C# test scripts directly in Visual Studio.
- The Silk4J Eclipse plug-in enables you to create Java-based test scripts directly in your Eclipse environment.

- SilkTest Recorder enables you to record and replay tests using a GUI and then export those tests to SilkTest Classic, Silk4J, or Silk4NET.
- SilkCentral Test Manager (SCTM) users can schedule, execute (including the ability to pass parameters), and get the results of SilkTest Workbench functional tests within SCTM test management software.

Users can also develop their own integrations with other test management software or their own custom automation frameworks.

## What's New in Silk4NET

This topic briefly describes the main features of Silk4NET.

Silk4NET is SilkTest's plug-in for Microsoft Visual Studio that allows you to efficiently create and manage functional, regression, and localization tests directly in the Visual Studio. With Silk4NET you can develop tests using either Visual Basic .NET or C#, run the tests as a part of a test plan in Microsoft's test environment or as a part of your build process, and view the test results all from within Visual Studio.

You can also use SilkTest Recorder to create Visual Basic or C# test scripts and then export them to Visual Studio. Or, you can use SilkTest Recorder to create Visual Studio projects that you export to Visual Studio.

Reference the online help in Visual Studio or in SilkTest Recorder for more information about Silk4NET.

## What's New in Silk4J

This topic lists the significant enhancements and changes that were made for Silk4J.

### Windows Presentation Foundation (WPF) Application Record and Replay Support

Silk4J includes record and replay support for Windows Presentation Foundation (WPF) controls. In Silk4J 2009, WPF replay support was provided. However, with the release of Silk4J 2010, the earlier WPF controls, which were prefixed with MSUIA, are deprecated and users should use the new WPF technology domain instead. When you record new test cases, Silk4J automatically uses the new WPF technology domain.

If you recorded tests with Silk4J 2009 that use the earlier MSUIA technology domain, the tests will continue to work. However, if you manually included the constant `TechDomain.WPF` (for the `Desktop.attach` or the `Desktop.executeBaseState` method) with tests the use the earlier MSUIA classes, you need to change the value to `TechDomain.MSUIA` to run the tests successfully.

All Silk4J classes that work with the earlier version of the MSUIA technology domain are marked as deprecated. Note that Eclipse strikes through all occurrences of deprecated APIs.

### SAP Record and Replay Support

Silk4J includes record and replay support for SAP controls.

### Dynamic Object Recognition Records Locators Differently

Previously, when you recorded a locator, Silk4J included a `Find` query for each level of the object's hierarchy, such as, `Find(/Window).Find(/Control)`. The resulting script included cascading `Find` queries. With this release, locators no longer include cascading `Find` queries. For example, instead of:

```
desktop.<Window>find("//Window").<PushButton>find
  ("//PushButton[@caption='ok']").select()
```

Silk4J records:

```
desktop.<PushButton>find("//Window//PushButton[@caption='ok']").select()
```

### Setting Custom Attributes

Silk4J includes a sophisticated locator generator mechanism that guarantees locators are unique at the time of recording and are easy to maintain. Depending on your application and the frameworks that you use, you might want to modify the default settings to achieve the best results. You can use any property that is available in the respective technology as a custom attribute given that they are either numbers (integers, doubles), strings, item identifiers, or enumeration values.

A well-defined locator relies on attributes that change infrequently and therefore requires less maintenance. Using a custom attribute is more reliable than other attributes like caption or index, since a caption will change when you translate the application into another language, and the index might change when another object is added.

Previously, you could set custom attributes for xBrowser and Java SWT applications using the **Script Options** dialog box. Now, you can set custom attributes for all technology domains with the exception of the Adobe Flex and Windows API-based client/server (Win32) technology domains.

## What's New in SilkTest Classic

 **Note:** SilkTest Classic will continue to be supported. However, new features are included in SilkTest Workbench rather than in SilkTest Classic. Refer to the SilkTest Workbench help for details about new features.

This topic lists the significant enhancements and changes that were made for SilkTest Classic.

### Windows Presentation Foundation (WPF) Application Record and Replay Support

SilkTest Classic includes record and replay support for Windows Presentation Foundation (WPF) controls. In SilkTest Classic 2009, WPF replay support was provided. However, with the release of SilkTest Classic 2010, the earlier WPF controls, which were prefixed with MSUIA, are deprecated and users should use the new WPF technology domain instead. When you record new test cases, SilkTest Classic automatically uses the new WPF technology domain.

If you have an existing project that includes scripts that use the earlier MSUIA technology domain, the testcases will continue to work. However, if you use .OPT files with the earlier MSUIA technology domain, manual changes are necessary to run the testcases successfully.

You can also set up your testcases to use the deprecated MSUIA technology domain and the new WPF technology domain side-by-side. This enables you to migrate to the new WPF technology domain because both the new WPF and deprecated MSUIA locators appear in the Locator Spy. You can then use the Locator Spy to migrate scripts to the new technology domain on a step-by-step basis. For example, you can change an application's main menu that was identified by `//MsuiaWindow//MsuiaMenu` previously to `//WPFWindow//WPFMenu`. You can enable side-by-side functionality by manually changing your OPT file (if you use option files rather than project files) or the `projectname.ini` file.

### SAP Record and Replay Support

SilkTest Classic includes record and replay support for SAP controls.

## Dynamic Object Recognition Records Locators Differently

Previously, when you recorded a locator, SilkTest Classic included a `Find` query for each level of the object's hierarchy, such as, `Find(/Window).Find(/Control)`. The resulting script included cascading `Find` queries. With this release, locators no longer include cascading `Find` queries. For example, instead of:

```
Desktop.Find("//Window").Find("//PushButton[@caption='ok']").Select()
```

SilkTest Classic records:

```
Desktop.Find("//Window//PushButton[@caption='ok']").Select()
```

## Setting Custom Attributes

The Open Agent includes a sophisticated locator generator mechanism that guarantees locators are unique at the time of recording and are easy to maintain. Depending on your application and the frameworks that you use, you might want to modify the default settings to achieve the best results. You can use any property that is available in the respective technology as a custom attribute given that they are either numbers (integers, doubles), strings, item identifiers, or enumeration values.

A well-defined locator relies on attributes that change infrequently and therefore requires less maintenance. Using a custom attribute is more reliable than other attributes like `caption` or `index`, since a `caption` will change when you translate the application into another language, and the `index` might change when another object is added.

In `xBrowser`, `WPF`, and `Java SWT` applications, you can also retrieve arbitrary properties (such as a `WPFButton` that defines `myCustomProperty`) and then use those properties as custom attributes.

Previously, you could set custom attributes for `xBrowser` and `Java SWT` applications. Now, you can set custom attributes for all technology domains with the exception of the `Adobe Flex` and `Windows API-based client/server (Win32)` technology domains.

# System Requirements and Prerequisites

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

## 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

SilkTest 2010 runs on the following platforms:

- Windows XP SP3
- Windows Vista SP1 or SP2
- Windows 7
- Windows 2008
- Windows 2008 R2

## Technology Domains and Browsers

SilkTest 2010 supports applications that use the following browsers or are developed in:

Technology Type	Supported Version for SilkTest Workbench, SilkTest Recorder, Silk4NET, Silk4J, SilkTest Classic using the Open Agent	Supported Version for SilkTest Classic using the Classic Agent
Adobe Flex	3.0, 3.2, 3.3, 3.4, 3.5	not supported
Firefox	3.0, 3.5, 3.6 (playback only)	not supported
Internet Explorer	6, 7, 8	6, 7, 8 (playback only)
Java AWT/Swing	not supported	Java 1.5, 1.6
Java SWT	3.2, 3.3, 3.4, 3.5, 3.6	not supported
SAP	SAPGUI client 7.10	not supported
Win 32	any	any
WinForms	.NET 2.0, 3.0, 3.5, 3.5 SP1, 4.0	not supported
WPF	.NET 3.5 SP1, 4.0	not supported

## Databases

SilkTest 2010 supports the following databases with the noted operating systems:

- SQL Server 2008 (SP1) + Express
- SQL Server 2005 (SP3) + Express
- Oracle 11.1
- Oracle 11.2

 **Note:** If you choose to install SQL Server 2008 Express SP1 during the SilkTest installation, ensure that you have one of the supported operating systems:

- Windows 7
- Windows Server 2008
- Windows Server 2008 R2
- Windows Vista SP1 or SP2

 **Note:** For Windows Vista SP1, you must install the full version of .NET 3.5 SP1 or 4.0 on your computer before you can install SQL Server. Otherwise, SQL Server Express might not install. To install the full version of .NET 3.5 SP1 or 4.0, download it from the Microsoft Web site.

- Windows XP Service Pack 3

# Tested Configurations

Tested configurations, by platform, are listed below. Note that it is not possible to test all combinations. Experience indicates that various combinations can result in different behavior with SilkTest. Micro Focus makes every attempt to test as many of these combinations as possible.

 **Note:** While you can use SilkTest on 64-bit operating systems, SilkTest only supports testing 32-bit applications.

 **Note:** You must have local administrator rights and UAC turned off to use SilkTest.

Platform Type	Tested Configurations for SilkTest Workbench, SilkTest Recorder, Silk4NET, Silk4J, SilkTest Classic using the Open Agent	Tested Configurations for SilkTest Classic using the Classic Agent
Windows 7 (32bit/64bit)	Internet Explorer 8	Internet Explorer 8 (playback only)
	Firefox 3.5, 3.6 (playback only)	Java 1.6 (AWT/Swing)
Windows 2008 SP2 (32bit/64bit)	Flex 3.5	Win 32
	SWT 3.6	
Windows 2008 R2 (64bit)	.NET 4.0 (WinForms, WPF)	
	Win 32	
	SAPGUI client 7.10 patch 13	
Windows Vista SP2 (32bit)	Internet Explorer 7	Internet Explorer 8 (playback only)
	Firefox 3.0	Java 1.6 (AWT/Swing)
	Flex 3.5	Win 32
	SWT 3.6	
	.NET 4.0 (WinForms, WPF)	
	Win 32	
Windows XP SP3 (32bit)	SAPGUI client 7.10 patch 13	
	Internet Explorer 7	Internet Explorer 8 (playback only)
	Firefox 3.0	
	Flex 3.0	
	SWT 3.5	
	.NET 4.0 (WinForms, WPF)	
	Win 32	
	SAPGUI client 7.10 patch 13	

## Adobe Flex Support

SilkTest supports testing Flex 3.0, 3.2, 3.3., 3.4, and 3.5 using the following:

- Internet Explorer 8 (GA)
- Internet Explorer 7 (GA)
- Firefox 3.0, 3.5, 3.6 (playback only)
- Standalone Flash Player 10
- Adobe AIR
- Internet Explorer 6 (SP 1 and SP 2)

 **Note:** Currently, Adobe offers experimental support for Flex applications in AIR. As a result, you may encounter issues when testing Flex in Adobe AIR that are not related to SilkTest.

## Java Support

### Open Agent Support

SilkTest supports the following Java components for SilkTest Workbench, SilkTest Recorder, Silk4NET, Silk4J, SilkTest Classic using the Open Agent:

- Eclipse 3.0, 3.1, 3.2, 3.3, 3.4, 3.5, and 3.6 SWT standalone and Rich Client Platform (RCP) applications (SWT standalone applications executed as an applet in the browser not supported, embedded IE control inside of SWT standalone or RCP application not supported)
- Java version 1.5 and 1.6

### Classic Agent Support

SilkTest supports the following Java components for SilkTest Classic using the Classic Agent:

- IBM JRE 1.5
- Java version 1.5 and 1.6
- AWT
- Java Foundation Classes (JFC, Swing)
- Symantec Visual Café
- Applets

 **Note:** Applets are not supported on Internet Explorer 8.

## SilkTest Classic DSN Support

SilkTest 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

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

 **Note:** The SilkTest 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.

## End-Of-Life (EOL) Components

The following operating systems, features, and integrations are not supported in SilkTest 2010.

- StarTeam integration
- PVCS integration
- SilkTest Classic 4Test outline Editor mode



**Note:** SilkTest Classic will continue to be supported. This change will not affect most SilkTest Classic users.

- Java 1.4
- Windows 2003 Server
- Windows XP SP2

## Known Issues

This section identifies known issues in SilkTest.

SilkTest's known issues and their resolutions are documented in the [Micro Focus Knowledge Base](#).

## Installation Instructions

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