

## **Silk TrueLog Explorer 10.0**

Help for Silk Test

**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.

2013-05-21

# Contents

<b>Getting Started</b> .....	<b>4</b>
TrueLog Explorer Overview .....	4
Tour of the TrueLog Explorer UI .....	4
Viewer Perspective .....	4
Menu Tree .....	5
Source Pane .....	6
Information Pane .....	6
Supported Applications .....	6
<b>Working with TrueLog Explorer</b> .....	<b>8</b>
Enabling TrueLog and Setting TrueLog Options in Silk Test Classic .....	8
TrueLog Limitations and Prerequisites .....	8
Setting TrueLog Options .....	9
Configuring TrueLog Options Using 4Test .....	10
Toggle TrueLog at Runtime Using a Script .....	10
Viewing Results Using the TrueLog Explorer .....	11
TrueLog Options - Classic Agent Dialog Box .....	11
TrueLog Options - Open Agent Dialog Box .....	14
Enabling TrueLog and Setting TrueLog Options in Silk4J .....	15
Setting TrueLog Options in the Silk Test Client .....	15
Replaying Test Methods from Eclipse .....	15
Replaying a Test Method from the Command Line .....	16
Enabling TrueLog and Setting TrueLog Options in Silk4NET .....	16
Enabling TrueLog .....	16
Setting TrueLog Options in the Silk Test Client .....	17
Opening a TrueLog in TrueLog Explorer .....	17
Finding Errors in a TrueLog .....	18
Enabling Compare Mode .....	18
Comparing Two TrueLogs .....	19
Closing a TrueLog .....	19
Silk Test Performance Considerations .....	19
Tips for Working with TrueLogs .....	20
Why is TrueLog Not Displaying Non-ASCII Characters Correctly? .....	20
<b>Customizing TrueLog Explorer</b> .....	<b>21</b>
Specifying Which Toolbars are Displayed .....	21
Customizing Toolbar Command Buttons .....	21

# Getting Started

TrueLog Explorer for Silk Test provides enhanced capabilities for test-script debugging. It also allows for root cause analysis of unexpected events and errors that occur during testing. TrueLog Explorer presents a graphical rendering of the activities of your testcases and the state of the application under test.

TrueLogs are history files in which extensive detail of testcase actions and system state are recorded during testing. In Silk Test, you configure TrueLog to capture actions, events, and images of interest. This information is collected while tests are run and is selected based on the settings you configure within Silk Test.

Once a test run is complete, use TrueLog Explorer to examine the contents of the corresponding TrueLog.

## TrueLog Explorer Overview

TrueLog Explorer enables you to:

- **Capture key information** – Log test and application information in TrueLogs.
- **Capture screenshots** – Capture bitmaps showing application state after specified script actions. To conserve storage space and processing power, you can limit screenshot capture to occur only following error conditions.
- **Compare TrueLogs** – Systematically compare TrueLogs from new test runs to TrueLogs from baseline test runs in which the application behaved as intended. Because bitmaps of sessions can be included, you can easily compare the differences between new test runs and baseline test runs.
- **Link directly to errors and lines in 4Test scripts** – Link directly from nodes in TrueLog Explorer to the corresponding lines in your 4Test code. This simplifies the process of locating errors and other issues in your code.

TrueLog Explorer presents a coordinated, multi-faceted view of all the information you choose to log.

TrueLog Explorer also supports results analysis for Silk Performer load tests. For details about this functionality, see Silk Performer help.

## Tour of the TrueLog Explorer UI

This section explains the main elements of the TrueLog Explorer interface. Note that interface elements vary based on the application type under test.



**Note:** Silk TrueLog Explorer supports TrueLogs that are created for both Silk Test and Silk Performer. When working with TrueLogs created for Silk Test, you may encounter certain menu items and tabs that are grayed out. These elements offer functionality that is only supported for Silk Performer TrueLogs.

The three major areas of the TrueLog Explorer UI are the **menu tree**, the **Source pane**, and the **Information pane**.

## Viewer Perspective

The version of TrueLog Explorer that is available with Silk Test only supports the Viewer perspective. Viewer perspective enables only a subset of the context-menu commands across all TrueLog Explorer views, and it disables the workflow bar. The intent of this perspective is to offer a simplified view for users who will not be customizing their scripts and to support TrueLog types for which script customization is not available. By default, Silk Test tests produce Viewer-perspective TrueLogs.

Explorer perspective (available with Silk Performer TrueLog types) offers the full range of script customization features and the workflow bar provided by TrueLog Explorer. For TrueLog types that offer Explorer perspective, it is possible to toggle back and forth between Explorer and Viewer perspectives.

## Menu Tree

When you open a TrueLog file, the menu tree on the left side of the interface lets you expand and collapse the TrueLog data that was saved while your testcase ran. Each open TrueLog file is displayed here and includes links to all captured information. The information is configured according to your selections of Silk Test TrueLog options.

If you open TrueLog Explorer without a TrueLog, a message displays that says, “Start by loading a TrueLog.”

The menu tree shows a hierarchical structure of nodes. Nodes exist for the scripts and testcases contained in the TrueLog. Within a testcase node, child nodes exist for each scripted action performed by the testcase, such as clicking a push button or entering text. Nodes also exist for every window that becomes active during the testcase.

The nodes that appear in the menu tree include the following:

- TrueLog files (a node exists for each open TrueLog file).
- Root (the execution node).
- Test plans (if available).
- Scripts.
- Testcases.
- Active windows. These are displayed only when you are using Silk Test Classic and the Classic Agent.
- Actions (user-selected script actions). These are displayed only when you are using Silk Test Classic and the Classic Agent.

Each active window or action node in a TrueLog optionally contains screenshots and a list of GUI controls (with the actual values) that existed at the time the window became active or an action occurred. Click a node to view its contents in the **Source** pane and its history details in the **Information** pane. Double-click nodes in the menu tree to open or collapse subfolders. By selecting a node and pressing the down arrow, you can sequentially step through the actions of all your tests. As you do so, the other two views display information related to the selected action.

For example, when you are using Silk Test Classic and the Classic Agent and you are viewing a Web application TrueLog, selecting nodes in the menu tree displays a screenshot of the application in the **Source** pane (the content that appeared at that moment during the test run). The Web application properties that were captured at that moment are displayed in the **Information** pane.

Clicking an Action node highlights the control in both the **Controls** tab of the **Information** pane and the **Before Action** tab of the **Source** pane.



**Note:** For 4Test, there are several things to note about the information that appears in the menu tree:

- If a script includes a `Main` function, the script actions that reside in main are displayed within the script node.
- Recovery state calls are displayed as Application State nodes.
- Several elements in 4Test test scripts have no particular action associated with them. A node is created in the menu tree for these elements; the type of node depends on the first action of the test script.
- For browser testing, the TrueLog creates a Window Active node for each document complete event.

# Source Pane

TrueLog Explorer provides multiple views of received data. The **Source** pane, displayed at the top of the TrueLog Explorer, displays the contents of the node selected in the menu tree. Source pane views that do not contain information relevant to Silk Test are inactive.

If you configured TrueLog Explorer to capture bitmaps, the **Source** pane displays a screenshot of the window (or desktop) at the time each action occurred. The view has a **Before Action** tab that shows the image prior to the action and an **After Action** tab that shows the resulting image.

- Before Action** This page displays the screenshots that are captured just before Silk Test executes each action in a testcase. To specify that bitmaps are to be logged while a testcase is running, use the **TrueLog Options** dialog box in Silk Test Classic or the **TrueLog** tab of the **Script Options** dialog box in Silk4J. When you are using Silk Test Classic and the Classic Agent, clicking a control in a screenshot in a **Before Action** view results in that control being highlighted on the **Controls** page, and vice versa.
- After Action** This page displays the screenshots that are captured just after Silk Test executes each action in a testcase. These bitmaps are logged while testcases are running if you specify that bitmaps are to be logged (using Silk Test's **TrueLog Options** dialog box).

# Information Pane

The **Information** pane is a panel that appears at the bottom of the TrueLog Explorer. It shows the data that is captured while testcases are running. Like the **Source** pane, the **Information** pane offers different views based on the application under test. Views that do not contain information relevant to Silk Test are inactive.

- Info** General information about the open TrueLog file and the selected node, including Silk Test file name, Time, and Completion status (including severity, time, and description). You can open successful image verifications by clicking **Open Verification**. You can open failed image verifications by clicking **Show Differences**. For failed image verifications, if a matching with a lower accuracy level would be successful, the highest accuracy level that would have succeeded is suggested.
- Call Stack** The call stack information of the current call. Click a line of code on this page to open Silk Test, if it is not already open. If you are using 4Test code, Silk Test opens the corresponding line within the testcase.
- Controls** This view is available only for Silk Test Classic and the Classic Agent. The view provides information about the controls that are logged (configurable using Silk Test's **TrueLog Options** dialog box). This information includes each control's name, value, type, and one or more of the following:
- CID – Control ID value
  - PID – Parent ID value
  - TL – Distance (in pixels) from the top left of the window
  - BR – Distance (in pixels) from the bottom right of the window

# Supported Applications

When you are using Silk Test Classic and the Classic Agent, you can use TrueLog Explorer for Silk Test to record test data for the following application types:

- client/server
- Web-based

- Java (Windows- and browser-based applications and applets)
- .NET (Windows- and browser-based applications)

When you are using the Open Agent with Silk Test Classic, Silk4J, or Silk4NET, you can use TrueLog Explorer for Silk Test to record test data for all supported application types.

# Working with TrueLog Explorer

This section explains how to work with TrueLog files using TrueLog Explorer. TrueLog is a powerful technology that simplifies root cause analysis of testcase failures using visual verification.

The results of test runs can be examined in TrueLog Explorer.

When you are working with Silk Test Classic and the Classic Agent and an error occurs during a test run, TrueLog Explorer enables you to easily locate the line in your 4Test script that generated the error. To do this, click the TrueLog action node that is associated with the error. Silk Test will open the script file and jump to the correct script line so that you can address the issue.

Compare mode enables side-by-side comparison of TrueLogs and the ability to step through the compared TrueLogs in parallel, node-by-node. This is helpful when comparing the results of different test runs that have been generated from the same test script.

## Enabling TrueLog and Setting TrueLog Options in Silk Test Classic

Before you can generate TrueLogs and view their contents using TrueLog Explorer, you must enable TrueLog for your Silk Test client.

You can enable or disable TrueLog Explorer:

- For all test cases using the **TrueLog Options** dialog box.
- Each time you run a specific test case using the **Run Testcase** dialog box.
- At runtime using the test script.

When you enable or disable TrueLog Explorer in the **Run Testcase** dialog box, Silk Test Classic makes the same change in the **TrueLog Options** dialog box. Likewise, when you enable or disable TrueLog Explorer in the **TrueLog Options** dialog box, Silk Test Classic makes the same change in the **Run Testcase** dialog box.



**Note:** By default, TrueLog Explorer is enabled when you are using the Open Agent, and disabled when you are using the Classic Agent. When TrueLog Explorer is enabled, the default setting is that screenshots are only created when an error occurs in the script and only test cases with errors are logged.

## TrueLog Limitations and Prerequisites

When you are using TrueLog with Silk Test Classic, the following limitations and prerequisites apply:

<b>Remote agents</b>	When you are using a remote agent, the TrueLog file is also written on the remote machine.
<b>Suites</b>	TrueLog is not supported when you are executing suites.
<b>Mixed-agent scripts</b>	TrueLog is not supported when you are executing mixed-agent scripts, which are scripts that are using both agents.
<b>Multiple-agent scripts</b>	TrueLog is supported only for one local or remote agent in a script. When you are using a remote agent, the TrueLog file is also written on the remote machine.
<b>Open Agent scripts</b>	To use TrueLog Explorer with Open Agent scripts, set the default agent in the toolbar to the Open Agent.



## Classic Agent scripts

To use TrueLog Explorer with Classic Agent scripts, set the default agent in the toolbar to the Classic Agent.

# Setting TrueLog Options

Use the TrueLog options to enable TrueLog and to customize the test result information that the TrueLog collects.

Logging bitmaps and controls in a TrueLog may adversely affect performance. Because capturing bitmaps and logging information can result in large TrueLog files, you may want to log test cases with errors only and then adjust the TrueLog options for test cases where more information is needed.

1. Click **Options > TrueLog** to open the **TrueLog Options** dialog box.
2. To capture TrueLog data and activate logging settings, check the **Enable TrueLog** check box and then choose to capture data for:

**All Testcases** Logs activity for all test cases, both successful and failed. This setting may result in large TrueLog files.

**Testcases with errors** Logs activity only for test cases with errors. This is the default setting.

3. In the **TrueLog File** field, specify the location and name of the TrueLog file.

This path is relative to the machine on which the Silk Test Classic Agent is running. The name defaults to the name used for the results file, with an `.xlg` extension. The location defaults to the same folder as the test case `.res` file.



**Note:** If you provide a local or remote path in this field, the path cannot be validated until script execution time.

4. Only when you are using the Classic Agent, choose one of the following to set pre-determined logging levels in the **TrueLog Presets** section:

**Minimal** Enables bitmap capture of desktop on error; does not log any actions.

**Default** Enables bitmap capture of window on error; logs data for Select and SetText actions; enables bitmap capture for Select and SetText actions.

**Full** Logs all control information; logs all events for browsers except for MouseMove events; enables bitmap capture of the window on error; captures bitmaps for all actions.

If you enable Full logs and encounter a `Window Not Found` error, you may need to manually edit your script.

5. Only when you are using the Classic Agent, in the **Log the following for controls** section, specify the types of information about the controls on the active window or page to log.
6. Only when you are using the Classic Agent, in the **Log the following for browsers** section, specify the browser events that you want to capture.
7. Specify the amount of time you want to allow Windows to draw the application window before a bitmap is taken.
  - When you are using the Classic Agent, specify the delay in the **TrueLog Delay** field.
  - When you are using the Open Agent, specify the delay in the **Delay** field in the **Screenshot mode** section.

The delay can be used for browser testing. You can insert a `Browser.WaitForReady` call in your script to ensure that the `DocumentComplete` events are seen and processed. If `WindowActive` nodes are missing from the TrueLog, you need to add a `Browser.WaitForReady` call. You can also use the delay to optimize script performance. Set the delay as small as possible to get the correct behavior and have the smallest impact on script execution time. The default setting is 0.

8. To capture screenshots of the application under test:

- When you are using the Classic Agent, check the **Enable Bitmap Capture** check box and then choose to capture bitmaps.
  - When you are using the Open Agent, determine how Silk Test Classic captures screenshots in the **Screenshot mode** section.
9. Only when you are using the Classic Agent, click the **Action Settings** tab to select the scripted actions you want to include in the TrueLog.  
When enabled, these actions appear as nodes in the Tree List view of the TrueLog.
  10. Only when you are using the Classic Agent, in the **Select Actions to Log** section, check the **Enable** check box to include the corresponding 4Test action in the log. Each action corresponds to a 4Test method, except for `Click` and `Select`.
  11. Only when you are using the Classic Agent, in the **Select Actions to Log** section, from the **Bitmap** list box, select the point in time that you want bitmaps to be captured.
  12. Click **OK**.

## Configuring TrueLog Options Using 4Test

This functionality is supported only if you are using the Classic Agent.

While it is recommended that you set TrueLog options using the **TrueLog Options** dialog box, you can also manipulate TrueLog options within a testcase with the `OPT_TRUELOG` Agent option and the `TRUELOGOPTIONS` record type.

For information about using `Agent.SetOption`, `Agent.GetOption()`, and `BindAgentOption()`, see Silk Test help.

1. Within a testcase, call `truelog.inc`.
2. Set the value of a particular option.  
The following example enables bitmaps.

```
[ ] use "truelog.inc"
[ ]
[ ] TRUELOGOPTIONS  rTLOpts
[ ] TRUELOGOPTIONS  rOldTLOpts
[ ]
[-] testcase TestTrueLogOptions () appstate none
  [ ] rTLOpts = Agent.GetOption (OPT_TRUELOG)
  [ ] print (rTLOpts)           //print the original TrueLog options
  [ ] rTLOpts.iActionSetText = TL_ACTION_ENABLE_BEFORE_BITMAPS
  [ ] rTLOpts.iActionSelect = TL_ACTION_ENABLE_BOTH_BITMAPS
  [ ] rOldTLOpts = Agent.SetOption (OPT_TRUELOG, rTLOpts)
  [ ] print (rTLOpts)         // print the new TrueLog options
```

3. Choose **File > Save** .

## Toggle TrueLog at Runtime Using a Script

This functionality is supported only if you are using the Classic Agent.

Toggle the TrueLog Explorer at runtime to analyze test results, capture screen-shots before and after each action, and capture screen-shots when an error occurs.

Use the test script to toggle TrueLog Explorer multiple times during the execution of a test case. For example, if you run a single test case to test multiple user interface menus, you can turn TrueLog on and off several times during the script to capture bitmaps for only a portion of the menus.

1. Set the TrueLog Explorer options to define what you want the TrueLog Explorer to capture.
2. Create or open the script that you want to modify.
3. Navigate to the portion of the script that you want to turn on or off.

4. To turn TrueLog off, type: `SetOption(OPT_PAUSE_TRUELOG, TRUE)`.
5. To turn TrueLog on, type: `SetOption(OPT_PAUSE_TRUELOG, FALSE)`.
6. Click **File** > **Save** to save the script.

## Viewing Results Using the TrueLog Explorer

Use the TrueLog Explorer to analyze test results files, capture screenshots before and after each action, and capture screenshots upon error.

1. Set the TrueLog Explorer options.
2. Run a test case.
3. Choose one of the following:
  - Click **Results** > **Launch TrueLog Explorer**.
  - Click the **Explore Results** button on the **Basic Workflow** or the **Data Driven Workflow** bars.
4. On the **Results Files** dialog box navigate to the file name that you want to review and click **Open**.

By default, the results file has the same name as the executed script, suite, or testplan. To review a file in the **TrueLog Explorer**, open a `.xlg` file. To review a Silk Test Classic results file in Silk Test Classic, open a `.res` file.

## TrueLog Options - Classic Agent Dialog Box

Use the **TrueLog Options - Classic Agent** dialog box to enable TrueLog for the Classic Agent and to customize the information that the TrueLog collects for Silk Test Classic.

Click **Options** > **TrueLog**.

Logging bitmaps and controls in TrueLog may adversely affect the performance of Silk Test Classic. Because capturing bitmaps and logging information can result in large TrueLog files, you may want to log test cases with errors only and then adjust the TrueLog options for test cases where more information is needed.



**Note:** By default, TrueLog Explorer is enabled when you are using the Open Agent, and disabled when you are using the Classic Agent. When TrueLog Explorer is enabled, the default setting is that screenshots are only created when an error occurs in the script and only test cases with errors are logged.

For additional information about TrueLog Explorer, refer to the *Silk TrueLog Explorer User Guide*, located in **Start** > **Programs** > **Silk** > **Silk Test** > **Documentation**.

### Logging Settings tab

Enables the TrueLog capture and configures the TrueLog.

#### Enable TrueLog

Captures TrueLog data and activates logging settings. You can also enable or disable TrueLog each time you run a test case by checking this check box in the **Run Testcase** dialog box. When you enable or disable TrueLog in the **Run Testcase** dialog box, Silk Test Classic makes the same change in the **TrueLog Options** dialog box. Likewise, when you enable or disable TrueLog in the **TrueLog Options** dialog box, Silk Test Classic makes the same change in the **Run Testcase** dialog box.

**All testcases**                      Logs activity for all test cases, both successful and failed.

**Testcases with errors**            Logs activity for only those test cases with errors.

#### TrueLog file

Designates the location and name of the TrueLog file. This path is relative to the machine on which the agent is running. The name defaults to the name used for the

results file, with an `.xlg` extension. The location defaults to the same folder as the test case `.res` file.



**Note:** If you provide a local or remote path in this field, the path cannot be validated until script execution time.

### Log the following for controls

Logs certain types of information about the controls on the active window or page.

#### Control information

Logs the GUI control's hierarchy, name, type, and other attributes for the active window or page. This information appears on the **Controls** tab of the **Information** window in TrueLog Explorer. This is selected by default when you enable TrueLog. If you choose to log control information, you may optionally decide to log:

- Control creation/deletion** Tracks the creation and/or deletion of controls on the active window or page. TrueLog updates the control hierarchy after each action. Choosing this option may adversely affect performance.
- Include static text controls** Includes the static text controls in the logged hierarchy. To keep TrueLogs small, leave this option turned off for browser testing.
- Track low level events** Logs keyboard and mouse events. For each mouse click and key press a new **Action** node is created on the Tree List in the TrueLog. Use caution when selecting this option for browser applications, as you can significantly impact performance.

### Log the following for browsers

Identifies the browser events you want to capture.

- Download events** Captures all events that trigger page downloads. Selected by default.
- Navigate events** Captures all events that cause new pages to appear. Selected by default.
- Terminate events** Captures all events that cause a browser to close. Selected by default.
- New window events** Captures all events that cause a new browser window to appear. Selected by default.
- MouseMove calls** Logs all scripted `MouseMove` calls. This is useful for tracking JavaScript `MouseOver` events. By including this information in TrueLogs, you may significantly affect performance. This information appears as an **Action** node in the Tree List.

### TrueLog delay

Allows Windows time to draw the application window before a bitmap is taken. The delay can be used for browser testing. You can insert a `Browser.WaitForReady` call in your script to ensure that the `DocumentComplete` events are seen and processed. If **WindowActive** nodes are missing from the **TrueLog**, you need to add a `Browser.WaitForReady` call. You can also use `TrueLog Delay` to optimize script performance. Set the delay as small as possible to get the correct behavior and have the smallest impact on script execution time. The default setting is 0.

### Enable bitmap capture

Controls when TrueLog captures screenshots of the application under test. Bitmap files are included in TrueLog (`.xlg`) files. Selected by default.

- As specified on action settings tab** Enables the capture of bitmaps for each action type you select on the **Action Settings** tab. This setting also causes a bitmap

	capture each time a window becomes active in your application.
<b>On error</b>	Captures a bitmap when an error occurs in your test case.
<b>Window only</b>	Saves the bitmaps for the active window.
<b>Desktop</b>	Saves a bitmap of the entire desktop.
<b>Before error bitmap</b>	Captures the bitmap before the error occurred.

### TrueLog Presets

Sets pre-determined logging levels.

<b>Minimal</b>	Logs test cases with errors; enables bitmap capture of desktop on error; does not log any actions.
<b>Default</b>	Logs test cases with errors; enables bitmap capture of window on error; logs data for <code>Select</code> and <code>SetText</code> actions; enables bitmap capture for <code>Select</code> and <code>SetText</code> actions.
<b>Full</b>	Logs all test cases; logs all control information; logs all events for browsers except for <code>MouseMove</code> events; enables bitmap capture of the window on error; captures bitmaps for all actions.

If you enable Full logs and encounter a `Window Not Found` error, you may need to manually edit your script.

### Action Settings tab

The **Action Settings** tab on the **TrueLog Options - Classic Agent** dialog box selects the scripted actions you want to include in the TrueLog. When enabled, these actions appear as nodes in the Tree List view of the TrueLog.

#### Select actions to log

**Enable** Selects the action to log. Each action corresponds to a 4Test method, except for `Click` and `Select`.

- |               |  |
|---------------|--|
| <b>Click</b>  | Records mouse clicks on many controls, such as <code>PushButton</code> , <code>ScrollBar</code> , <code>TextField</code> , and <code>HtmlLink</code> . To record <code>Click</code> methods on a <code>CheckBox</code> , choose <code>Select</code> , not <code>Click</code> .   |
| <b>Select</b> | Records actions on multiple methods of multiple types of controls, including <code>ListBox</code> , <code>TreeView</code> , <code>ComboBox</code> , <code>RadioButton</code> , and <code>CheckBox</code> . Here is a partial list of what <code>Select</code> records: <ul style="list-style-type: none"> <li>• <b>Select (ListBox, TreeView, ComboBox, RadioButton)</b></li> <li>• <b>DoubleSelect, SelectList, SelectRange (ListBox, TreeView)</b></li> <li>• <b>Click (ListBox, TreeView, ComboBox, RadioButton, CheckBox)</b></li> <li>• <b>Check, Uncheck, Toggle, SetState (CheckBox)</b></li> </ul> |

**Bitmap** Selects the point in time you want bitmaps to be captured:

- |               |                                       |
|---------------|---------------------------------------|
| <b>None</b>   | Never captures bitmaps.               |
| <b>Before</b> | Captures bitmaps before errors occur. |
| <b>After</b>  | Captures bitmaps after errors occur.  |

**Both** Capture bitmaps both before and after errors occur.

## TrueLog Options - Open Agent Dialog Box

Use the **TrueLog Options - Open Agent** dialog box to enable TrueLog for the Open Agent and to customize the information that the TrueLog collects for Silk Test Classic.

Click **Options > TrueLog**.

You can additionally use the **Difference Viewer** to analyze results for test cases that use the Open Agent.

Logging bitmaps and controls in a TrueLog may adversely affect the performance of Silk Test Classic. Because capturing bitmaps and logging information can result in large TrueLog files, you may want to log test cases with errors only and then adjust the TrueLog options for test cases where more information is needed.



**Note:** By default, TrueLog Explorer is enabled when you are using the Open Agent, and disabled when you are using the Classic Agent. When TrueLog Explorer is enabled, the default setting is that screenshots are only created when an error occurs in the script and only test cases with errors are logged.

For additional information about TrueLog Explorer, refer to the *Silk TrueLog Explorer User Guide*, located in **Start > Programs > Silk > Silk Test > Documentation**.

### Enable TrueLog

Captures TrueLog data and activates logging settings. You can also enable or disable TrueLog each time you run a test case by checking this check box in the **Run Testcase** dialog box. When you enable or disable TrueLog in the **Run Testcase** dialog box, Silk Test Classic makes the same change in the **TrueLog Options** dialog box. Likewise, when you enable or disable TrueLog in the **TrueLog Options** dialog box, Silk Test Classic makes the same change in the **Run Testcase** dialog box.

**All testcases** Logs activity for all test cases, both successful and failed.

**Testcases with errors** Logs activity for only those test cases with errors.

### TrueLog file

Designates the location and name of the TrueLog file. This path is relative to the machine on which the agent is running. The name defaults to the name used for the results file, with an `.xlg` extension. The location defaults to the same folder as the test case `.res` file.



**Note:** If you provide a local or remote path in this field, the path cannot be validated until script execution time.

### Screenshot mode

Controls when TrueLog Explorer captures test application screens during playback of the application under test. Bitmap files are included in TrueLog (`.xlg`) files.



**Note:** For test cases with errors, TrueLog Explorer always captures a screen, independent of the screenshot mode you have selected.

**None** Captures no screens during playback.

**Active Window** Captures only the active window of the test application during playback.

**Active Application** Captures the test application and any windows within the test application.

**Desktop** Captures the entire desktop area.

## Delay

Allows Windows time to draw the application window before a bitmap is taken. You can also use `Delay` to optimize script performance. Set the delay as small as possible to get the correct behavior and have the smallest impact on script execution time. The default setting is 0.

# Enabling TrueLog and Setting TrueLog Options in Silk4J

Before you can generate TrueLogs and view their contents using TrueLog Explorer, you must enable TrueLog for your Silk Test client. Additionally you must run your test cases as Silk4J tests.

When you are working with Silk4J you can enable or disable TrueLog Explorer in the **TrueLog** tab of the **Script Options** dialog box.



**Note:** By default, TrueLog Explorer is enabled when you are using Silk4J.

## Setting TrueLog Options in the Silk Test Client

Enable TrueLogs to capture bitmaps and to log information for your Silk Test client.

Logging bitmaps and controls in TrueLogs may adversely affect the performance of Silk Test. Because capturing bitmaps and logging information can result in large TrueLog files, you may want to log test cases with errors only and then adjust the TrueLog options for test cases where more information is needed.

The results of test runs can be examined in TrueLog Explorer. For additional information about TrueLog Explorer, refer to the *Silk TrueLog Explorer User Guide*, located in **Start > Programs > Silk > Silk Test > Documentation**.

To enable TrueLog and customize the information that the TrueLog collects for your Silk Test client, perform the following steps:

1. In the menu, click on the Silk Test client menu entry and then click **Edit Options**. The **Script Options** dialog box opens.
2. Click the **TrueLog** tab.
3. Select the **Screenshot mode**.  
Default is **None**.
4. *Optional:* Set the **Delay**.  
This delay gives Windows time to draw the application window before a bitmap is taken. You can try to add a delay if your application is not drawn properly in the captured bitmaps.
5. Click **OK**.

## Replaying Test Methods from Eclipse

1. Navigate to the test method that you want to test.
2. *Optional:* Click on a browser icon in the toolbar to set the browser on which you want to replay the test.
3. Perform one of the following steps:
  - Right-click a package name in the Package Explorer to replay all test methods in the project.
  - Right-click a class name in the Package Explorer to replay all test methods in the class. Or, alternatively, open the class in the source editor and right-click in the source editor.
  - Right-click a method name in the Package Explorer to replay a test for only that method. Or, alternatively, open the class in the source editor and select a test method by clicking its name.
4. Choose **Run As > Silk4J Test**. The test results display in the JUnit view as the test runs. If all tests pass, the status bar is green. If one or more tests fail, the status bar is red. You can click a failed test to display the stack trace in the Failure Trace area.

5. When the test execution is complete, the **Playback Complete** dialog box opens. Click **Explore Results** to review the TrueLog for the completed test.

## Replaying a Test Method from the Command Line


You must update the PATH variable to reference your JDK location before performing this task. For details, reference the Sun documentation at: <http://java.sun.com/j2se/1.5.0/install-windows.html>.

1. Set the CLASSPATH to:

```
set CLASSPATH=<eclipse_install_directory>\plugins
\org.junit4_4.3.1\junit.jar;
%OPEN_AGENT_HOME%\JTF\silktest-jtf-nodeps.jar;C:\myTests.jar
```

2. Run the JUnit test method by typing:

```
java org.junit.runner.JUnitCore <test class name>
```

 **Note:** For troubleshooting information, reference the JUnit documentation at: [http://junit.sourceforge.net/doc/faq/faq.htm#running\\_1](http://junit.sourceforge.net/doc/faq/faq.htm#running_1).

3. To run several test classes with Silk4J and to create a TrueLog, use the `SilkTestSuite` class to run the Silk4J tests.

For example, to run the two classes `MyTestClass1` and `MyTestClass2` with TrueLog enabled, type the following code into your script:

```
package demo;
import org.junit.runner.RunWith;
import org.junit.runners.Suite.SuiteClasses;
import com.borland.silktest.jtf.SilkTestSuite;

@RunWith(SilkTestSuite.class)
@SuiteClasses({ MyTestClass1.class, MyTestClass2.class })
public class MyTestSuite {}
```


To run these test classes from the command line, type the following:

```
java org.junit.runner.JUnitCore demo.MyTestSuite
```

## Enabling TrueLog and Setting TrueLog Options in Silk4NET

Before you can generate TrueLogs and view their contents using TrueLog Explorer, you must enable TrueLog for your Silk Test client. Additionally, 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.

When you are working with Silk4NET you can enable or disable TrueLog Explorer in the **TrueLog** tab of the **Script Options** dialog box.

 **Note:** By default, TrueLog Explorer is enabled when you are using Silk4NET.

### Enabling TrueLog

For new Silk Test scripts, TrueLog is enabled by default. To enable TrueLog for existing Silk Test 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 Silk Test scripts are located. When the Silk Test 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
  }
}
```

## Setting TrueLog Options in the Silk Test Client

Enable TrueLogs to capture bitmaps and to log information for your Silk Test client.

Logging bitmaps and controls in TrueLogs may adversely affect the performance of Silk Test. Because capturing bitmaps and logging information can result in large TrueLog files, you may want to log test cases with errors only and then adjust the TrueLog options for test cases where more information is needed.

The results of test runs can be examined in TrueLog Explorer. For additional information about TrueLog Explorer, refer to the *Silk TrueLog Explorer User Guide*, located in **Start > Programs > Silk > Silk Test > Documentation**.

To enable TrueLog and customize the information that the TrueLog collects for your Silk Test client, perform the following steps:

1. In the menu, click on the Silk Test client menu entry and then click **Edit Options**. The **Script Options** dialog box opens.
2. Click the **TrueLog** tab.
3. Select the **Screenshot mode**.  
Default is **None**.
4. *Optional*: Set the **Delay**.  
This delay gives Windows time to draw the application window before a bitmap is taken. You can try to add a delay if your application is not drawn properly in the captured bitmaps.
5. Click **OK**.

## Opening a TrueLog in TrueLog Explorer

By default, TrueLogs are saved to the same directory as other Silk Test test result (`.res`) files. You can however specify a new location for TrueLog `.xlg` files.

TrueLogs share the same base filename as other Silk Test result files (for example, `test1.res` and `test1.xlg`), but they have the `.xlg` suffix.

1. Choose **Start > Programs > Silk > Silk Test <version> > Analysis Tools > Silk TrueLog Explorer** .
2. Choose **File > Add TrueLogs** or **File > Add TrueLog On Error Files** . The **Open** dialog box opens with the specified file type selected in the **Files of Type** list box.
3. Navigate to the directory of the relevant test and select the file to be explored.
4. *Optional:* Check the **Open in Compare View** check box to open the file in Compare view.
5. Click **OK**.



**Note:** Multiple TrueLogs can be open in TrueLog Explorer simultaneously.

## Finding Errors in a TrueLog

TrueLog Explorer helps you find replay errors in test results. Based on those errors, you can then make any required corrections to your test script.



**Note:** When viewed in the menu tree, nodes that contain errors are tagged with red “X” marks.

1. Select a TrueLog in the TrueLog menu tree.
2. Choose **Edit > Find Errors** . The **Find Errors** dialog box opens.
3. From the **Search in** list box, select one of the following items:
  - **all open TrueLogs** – searches for errors in all open TrueLogs.  
The **all open TrueLogs** option is selected by default.
  - **selected TrueLog only** – searches for errors only within the active TrueLog.
4. In the **Find what** group box, check the check boxes next to the items that you want to find.
  - To include errors in your search, check the **Errors** check box.
  - To include warning messages in your search, check the **Warnings** check box.
  - To include informational messages in your search, check the **Informational** check box.
5. In the **Search strategy** group box, specify how you want to search for errors.
  - Click the **Start from selected node** option button to begin your search at the active node.
  - Click the **Start with first error** option button to begin your search from the first encountered node that includes an error.
6. Click **Find Next** to step through TrueLog result files one error at a time.  
You can link to the corresponding Silk Test testcase of any error that you encounter to correct the error or to update information.

## Enabling Compare Mode

Use compare mode to compare a replay TrueLog alongside a corresponding record TrueLog to analyze playback errors. When you check the **Open in Compare View** check box on the **Open** dialog box, TrueLogs are opened in compare mode automatically. Otherwise, TrueLogs are opened in default view.



**Note:** Compare mode offers the option of having record and replay **Source** pages oriented vertically rather than horizontally. To compare **Source** pages vertically, choose **View > Compare Vertically** .

1. Select a TrueLog in the TrueLog menu tree.
2. Perform one of the following steps:
  - Click **Compare Mode**.
  - Choose **View > Compare Mode** .



**Note:** Multiple entry points and approaches are available for the comparison of TrueLogs.

## Comparing Two TrueLogs

Comparing two TrueLogs enables you to walk through two or more test runs generated by the same test script to identify playback errors. 4Test scripts must be edited within Silk Test.

1. Choose **File > Add TrueLogs** . The **Open** dialog box opens.
2. Select the first TrueLog that you want to compare and click **Open**. The TrueLog opens in the TrueLog menu tree.
3. Choose **File > Add TrueLogs** . The **Open** dialog box opens.
4. Select the other TrueLog that you want to compare.
5. Check the **Open in Compare View** check box.
6. Click **Open**. The second TrueLog opens in the TrueLog menu tree below the first TrueLog.
7. Choose **Edit > Step through TrueLog** . The **Step through TrueLog** dialog box opens with the Silk Test nodes option selected.
8. Click **Find Next** to step through the two TrueLog files one node at a time, side-by-side.

## Closing a TrueLog

Once you have finished analyzing a TrueLog, you may want to close it to clean up the menu tree.

1. In the TrueLog menu tree, select the TrueLog that you want to close.
2. Choose **File > Remove selected TrueLog** .

**Alternative:** To close all open TrueLogs, choose **File > Remove all TrueLogs** .

## Silk Test Performance Considerations


Logging bitmaps and controls with TrueLog may adversely affect the performance of Silk Test. Because capturing bitmaps and logging information can result in large TrueLog files, you may only want to log testcases that result in errors (TrueLog on Error). You can then adjust TrueLog options to accommodate other testcases as required.

If you are running long regression tests that require debugging, you can capture bitmaps for only those testcases that result in errors (TrueLog on Error). This option is set on the **TrueLog Options** dialog box.

When you are working with Silk Test Classic and the Classic Agent, each action type you specify for logging (on the **Actions Settings** page of the **TrueLog Options** dialog box) results in a new node in the TrueLog menu tree. It is recommended that you only log those action types that you are interested in.

# Tips for Working with TrueLogs

These tips are only relevant when you are working with Silk Test Classic and the Classic Agent.

 **Note:** TrueLog Explorer for Silk Test does not currently support 4Test script customization or application-protocol specific functionality.

## Enable bitmap capture for "desktop" rather than "active window"

It is recommended that you capture the entire desktop image rather than just the active window if you suspect that another application or dialog box appears on top of the application under test. To do this, on the **TrueLog Options** dialog box, in the **Enable Bitmap Capture** group box, click **On Error**, then click **Desktop**. Any time you suspect an error is caused by a system condition other than the application under test, consider using this option.

## Controls that are out of view

TrueLog Explorer cannot capture images of controls that are out of view at the moment of capture. To ensure that a TrueLog bitmap shows a relevant control related to a particular action, you must scroll the control into view.

## SetText and MultiText actions captured in bitmaps

When you capture bitmaps for `SetText` and `SetMultiText` actions, TrueLog Explorer displays the text that is set for those actions in large red letters that are superimposed on the bitmaps. The position of the text is in the physical center of the bitmap and is unrelated to the control into which the text was entered. This can be misleading when the control is not visible in the captured bitmap.

## Results page contains non-printable characters

If the data on the **Results** page contains non-printable characters, the data will be displayed in both ASCII and hexadecimal characters. You can turn off this behavior using the TrueLog Explorer **Options** dialog box. Choose **Settings > Options** in TrueLog Explorer, then click the **Display** tab. Uncheck all check boxes in the **Data format** group box.

# Why is TrueLog Not Displaying Non-ASCII Characters Correctly?

TrueLog Explorer is a MBCS-based application, meaning that to be displayed correctly, every string must be encoded in MBCS format. When TrueLog Explorer visualizes and customizes data, many string conversion operations may be involved before the data is displayed.

Sometimes when testing UTF-8 encoded Web sites, data containing characters cannot be converted to the active Windows system code page. In such cases, TrueLog Explorer will replace the non-convertible characters, which are the non-ASCII characters, with a configurable replacement character, which usually is '?'.

To enable TrueLog Explorer to accurately display non-ASCII characters, set the system code page to the appropriate language, for example Japanese.

# Customizing TrueLog Explorer

This section shows you how to take advantage of TrueLog Explorer toolbar customization and view features to better adapt TrueLog Explorer to your needs.

## Specifying Which Toolbars are Displayed

You can customize the set of toolbars that appear on the TrueLog Explorer interface.

1. Choose **Settings > Customize** . The **Toolbars** page opens.
2. Check the check boxes next to the toolbars that you want to have displayed.
3. *Optional:* To specify that you want to enable tooltips (roll-over UI control descriptions), check the **Show Tooltips** check box.
4. *Optional:* To specify that you want to enable TrueLog Explorer “cool look,” check the **Cool Look** check box. Cool look replaces Windows 3.1 style drop-shadow buttons with non-drop-shadow buttons.
5. Click **OK**.

*Alternative:* Click **Apply** to immediately apply your selections to the UI.

## Customizing Toolbar Command Buttons

You can define which command buttons display on specific toolbars.



**Note:** The **Menu** toolbar cannot be customized.

1. Choose **Settings > Customize** . The **Toolbars** page opens.
2. Click the **Commands** tab.
3. In the **Categories** list box, select the toolbar for which you want to customize the command buttons.
4. Click available commands in the **Buttons** box to read their descriptions in the **Description** box.
5. Drag the commands that you want to include on the toolbar onto the toolbar name in the **Categories** list box.
6. Click **OK**.

# Index

4Test  
  script customization 20  
  setting options 10

## A

actions 5  
active windows 5  
After Action tab 6  
agents  
  classic 8  
  open 8  
  Silk Test 8  
application state node 5

## B

Before Action tab 6  
bitmaps  
  capturing 20  
BR 6

## C

Call Stack tab 6  
CID 6  
command line  
  running test methods 16  
commands  
  customizing 21  
compare mode 18  
completion status 6  
Controls tab 6  
customizing  
  commands 21  
  toolbars 21  
  TrueLog Explorer 21

## D

Difference Viewer 8

## E

enabling TrueLog  
  TrueLog Explorer 16  
errors  
  finding 18

## I

Information pane 6

## M

Main function 5

menu tree 5  
modes  
  compare 18

## O

opening  
  TrueLog On Error 17  
  TrueLogs 17

## P

panes  
  Information 6  
  Source 6  
performance considerations 19  
perspectives  
  Explorer 4  
  Viewer 4  
PID 6  
product  
  introduction 4  
  overview 4

## R

recovery states 5  
Results tab 20

## S

scripts 5  
setting options  
  TrueLog 9  
  TrueLog Explorer 9  
Silk Test  
  agents 8  
Source pane 6  
Source tab 6  
Statistics tab 6  
supported applications  
  TrueLog 6

## T

tabs  
  After Action 6  
  Before Action 6  
  Call Stack 6  
  Controls 6  
  Results 20  
  Source 6  
  Statistics 6  
test method  
  running 15  
test methods  
  running from command line 16

- running from Eclipse 15
- test plans 5
- tests
  - analyzing 18, 19
- TL 6
- toolbars
  - customizing commands 21
  - displaying 21
- tooltips 21
- TrueLog
  - configuring 15, 17
  - enabling 15–17
  - limitations 8
  - options for the Classic Agent 11
  - options for the Open Agent 14
  - prerequisites 8
  - replacement characters for non-ASCII 20
  - setting options 9
  - supported applications 6
  - wrong non-ASCII characters 20
- TrueLog Explorer
  - configuring 15, 17
  - enabling 15, 17
  - enabling TrueLog 16
  - interface 4
  - setting options 9

- viewing results 11
- viewing results 11
- TrueLog On Error
  - opening 17
- TrueLog Options - Classic Agent
  - dialog box 11
- TrueLog Options - Open Agent
  - dialog box 14
- TrueLogs
  - closing 19
  - comparing 19
  - enabling in Silk Test Classic 8
  - enabling in Silk4J 15
  - enabling in Silk4NET 16
  - finding errors 18
  - introduction 4
  - opening 17
  - tips 20

## V

- Viewer perspective 4
- viewing results
  - TrueLog Explorer 11