Silk TrueLog Explorer 2009 Release 2

Help for SilkTest

Borland®
(A MICRO FOCUS COMPANY)

MICRO FOCUS®
Leading the Evolution™

Contents

Getting Started

TrueLog Explorer for SilkTest 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 SilkTest, 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 SilkTest.

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 SilkPerformer load tests. For details about this functionality, see SilkPerformer 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 SilkTest and SilkPerformer. When working with TrueLogs created for SilkTest, you may encounter certain menu items and tabs that are grayed out. These elements offer functionality that is only supported for SilkPerformer 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 SilkTest 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, SilkTest tests produce Viewer-perspective TrueLogs.

Explorer perspective (available with SilkPerformer 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 SilkTest 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
- Actions (user-selected script actions)

   Note: The results of Print statements are logged on the Testcase node and displayed in the **Information** pane.

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 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.

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 SilkTest 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 SilkTest executes each action in a testcase. These bitmaps are logged while testcases are running if you specify that bitmaps are to be logged (using SilkTest's **TrueLog Options** dialog box). Clicking a control in a screenshot in a **Before Action** view results in that control being highlighted on the **Controls** page, and vice versa.
Source (HTML)	This page displays HTML code that is used to generate Web content. Contains the same information as the **HTML Content** page in the **Information** pane.

> **Note:** If TrueLog Explorer detects non-printable characters in an HTML page while "auto view" mode is enabled, the **Source** page displays the HTML source code in non-readable binary format.

After Action	This page displays the screenshots that are captured just after SilkTest executes each action in a testcase. These bitmaps are logged while testcases are running if you specify that bitmaps are to be logged (using SilkTest'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 SilkTest are inactive.

Info	General information about the open TrueLog file and the selected API node, including SilkTest file name and path, Function, Line number, Time, and Completion status (including severity, time, and description).
4Test	The call stack information of the current call. Click a line of code on this page to open SilkTest (if it is not already open) and jump to the corresponding line within the testcase.
Controls	Information about the controls that are logged (configurable using SilkTest'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

Result   Logs the result information that is typically logged to SilkTest result (`.res`) files.

HTML Content   Displays the HTML code that is used to generate Web content. Contains the same information as the **Source** view in the **Source** pane.

Statistics   Includes process handle, process threads, and virtual memory information.

Supported Applications

You can use TrueLog Explorer for SilkTest 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)

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.

Note: TrueLog Explorer only works with SilkTest Classic Agent. Use the Difference Viewer to analyze results for testcases that use SilkTest Open Agent.

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

Before you can generate TrueLogs and view their contents using TrueLog Explorer, you must enable TrueLog for SilkTest. You can enable TrueLog using the **Run Testcase** dialog box or the **TrueLog Options** dialog box. Using the **Run Testcase** dialog box enables you to enable or disable TrueLog each time you run a testcase. When you enable or disable TrueLog using the **Run Testcase** dialog box, SilkTest makes the same change in the **TrueLog Options** dialog box. Likewise, when you enable or disable TrueLog using the **TrueLog Options** dialog box, SilkTest makes the same change in the **Run Testcase** dialog box.

Note: TrueLog recording in SilkTest is disabled by default.

Enabling TrueLog Using the TrueLog Options Dialog Box

Before you can generate TrueLog and view TrueLog content using TrueLog Explorer, you must enable TrueLog for SilkTest.

1. In SilkTest, choose **Options ➤ TrueLog**.
   The **TrueLog Options** dialog box opens.
2. Check the **Enable TrueLog (for Classic Agent only)** check box.
3. Specify how data should be captured for testcases by clicking one of the following options:

   By default, the **Testcases with errors** option is selected.

   - **All testcases** – Logs activity for all testcases, both successful and failed. This setting may result in large TrueLog files.
   - **Testcases with errors** – Logs activity for only those testcases that have errors. This is the recommended setting as it limits the size of TrueLog files.

4. In the **TrueLog presets** group box, configure logging options by clicking one of the following buttons:

- **Minimal** – Logs testcases that have errors; enables bitmap capture of the desktop upon error; does not log actions.
- **Default** – Logs testcases that have errors; enables bitmap capture of the window upon error; logs data for `Select` and `SetText` actions; enables bitmap capture for `Select` and `SetText` actions.
- **Full** – Logs all testcases; logs all control information; logs all events for browsers except for `MouseMove` events; enables bitmap capture of the window upon error; captures bitmaps for all actions.

5. Click **OK**.

Enabling TrueLog Using the Run Testcase Dialog Box

With the **Run Testcase** dialog box you can enable or disable TrueLog each time you run a testcase.

1. Within SilkTest, ensure that the testcase you want to run is in the active window.

2. Click **Run Testcase** on the workflow bar.
The **Run Testcase** dialog box opens and lists all the testcases that are contained in the current script.

3. Select a testcase, and specify arguments if necessary, in the **Arguments** box.

Separate multiple arguments with commas.

4. *Optional*: To create a one-second delay after each script line is executed, check the **Animated Run Mode (Slow-Motion)** check box.

Typically, you only use this check box when you want to watch a testcase run. For example, when you want to demonstrate a testcase to someone.

5. Check the **Enable TrueLog (for Classic Agent only)** check box.

6. *Optional*: Click **TrueLog Options** to set the options that you want to use when recording TrueLogs.

When you do not click **TrueLog Options**, SilkTest uses the most recent settings specified on the **TrueLog Options** dialog box. By default, SilkTest logs activity for only those testcases that have errors; enables bitmap capture of the window upon error; logs data for `Select` and `SetText` actions; and enables bitmap capture for `Select` and `SetText` actions.

a) Specify how data should be captured for testcases by clicking one of the following options:

- **All testcases** – Logs activity for all testcases, both successful and failed. This setting may result in large TrueLog files.
- **Testcases with errors** – Logs activity for only those testcases that have errors. This is the recommended setting as it limits the size of TrueLog files.

b) In the **TrueLog presets** group box, configure logging options by clicking one of the following buttons:

- **Minimal** – Logs testcases that have errors; enables bitmap capture of the desktop upon error; does not log actions.
- **Default** – Logs testcases that have errors; enables bitmap capture of the window upon error; logs data for `Select` and `SetText` actions; enables bitmap capture for `Select` and `SetText` actions.
- **Full** – Logs all testcases; logs all control information; logs all events for browsers except for `MouseMove` events; enables bitmap capture of the window upon error; captures bitmaps for all actions.

c) Click **OK**.

7. Click **Run**.
SilkTest runs the testcase and generates a results file.

If required, refine your TrueLog-generation settings.

TrueLog Options

Before you can create TrueLogs, you must specify the types of information and actions that you want to have logged.

There are two ways to configure TrueLog options:

- Using the **TrueLog Options** dialog box.
- Within individual testcases using `SetAgent` options.

Each SilkTest project retains its TrueLog settings, unless you change these settings within the project.

TrueLog settings are not available through option sets.

The `Results` value TrueLog option is only available through 4Test; all other options may be set on the **TrueLog Options** dialog box. To change the level of the results that are captured by TrueLog, you must change the `Results` value using `SetOption` to:

- `TL_RESULTS_ERROR` or `0` to log only errors
- `TL_RESULTS_WARNING` or `1` to log warnings and errors
- `TL_RESULTS_INFORMATIONAL` or `2` to log print statements, warnings, and errors

Configuring Options Using the TrueLog Options Dialog Box

Specify the types of information and actions that are to be logged.

1. In SilkTest, choose **Options ➤ TrueLog**.
   The **TrueLog Options** dialog box opens.
2. On the **Logging Settings** page, check the **Enable TrueLog (for Classic Agent only)** check box to activate logging settings.

   > 🖉 **Note:** When TrueLog is enabled, testcases run on the AWT test application will hang. The hang occurs when the testcase attempts to select an item from the **Controls** menu. To work around this, place a `SetActive()` call before the **Controls** menu pick.

3. Specify how data should be captured for testcases by clicking one of the following options:
   By default, the **Testcases with errors** option is selected.

   - **All testcases** – Logs activity for all testcases, both successful and failed. This setting may result in large TrueLog files.
   - **Testcases with errors** – Logs activity for only those testcases that have errors. This is the recommended setting as it limits the size of TrueLog files.

4. *Optional*: Specify the TrueLog filename or path.
   If a path is supplied, it is relative to the machine the SilkTest Agent runs on. If you do not configure this setting, the file will default to the name used for the results file, with an `.xlg` extension. (If you enter a filename with no extension, the default is `.xlg`.) If you do not enter a path, the `.xlg` file will be saved to the same directory as the testcase `.res` file. Click **Browse** to specify a different location for the `.xlg` file. If you enter a path (either local or remote), the location cannot be validated until the script is executed.

5. In the **TrueLog presets** group box, configure logging options by clicking one of the following buttons:

- **Minimal** – Logs testcases that have errors; enables bitmap capture of the desktop upon error; does not log actions.
- **Default** – Logs testcases that have errors; enables bitmap capture of the window upon error; logs data for `Select` and `SetText` actions; enables bitmap capture for `Select` and `SetText` actions.
- **Full** – Logs all testcases; logs all control information; logs all events for browsers except for `MouseMove` events; enables bitmap capture of the window upon error; captures bitmaps for all actions.

6. To log the GUI control's hierarchy, name, type, and other attributes for the active window or page, check the **Control information** check box in the **Log the following for controls** group box.

   The **Control information** check box is checked by default when you enable TrueLog.

   Optional: If you choose to log control information, you can check the following check boxes to log the following information:

   - **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. This option can also increase the size of the TrueLog, especially when testing tabbed windows.
   - **Include static text controls** – Includes the static text controls in the logged hierarchy. Leave this option turned off for browser testing. Otherwise, you will dramatically increase the size of TrueLogs for Web testing.

   This information appears on the **Controls** page of the **Information** pane in TrueLog Explorer.

7. To log keyboard and mouse events, check the **Track low level events** check box in the **Log the following for controls** group box.

   Use caution when selecting this option for browser applications (use the **Log the following for browsers** group box of the dialog, as explained below). By including low level events in TrueLogs, you may significantly impact performance.

   For each mouse click and key press, a new **Action** node is created in the TrueLog Explorer menu tree.

8. In the **Log the following for browsers** group box, check the check boxes for the browser events that you want TrueLog to capture.

   - **Download events** – Capture all events that trigger page downloads.
   - **Navigate events** – Capture all events that cause new pages to appear.
   - **Terminate events** – Capture all events that cause a browser to close.
   - **New window events** – Capture all events that cause a new browser window to appear.
   - **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 TrueLog Explorer menu tree.

   These events (except for `MouseMove`) appear on the **Information** pane.

9. In the **Delay** box, specify the amount of time that TrueLog Explorer should wait before capturing a bitmap.

   The **Delay** setting gives Windows time to draw an application before a bitmap is taken. The delay can be used to optimize script performance. For the least impact on script execution time, set the delay as short as possible. The default setting is `0`.

   🖉 **Note:** To use the delay for browser testing, insert a `Browser.WaitForReady` call in your script. This ensures that the `DocumentComplete` events are seen and processed. (If `WindowActive` nodes are missing from the TrueLog, you must add a `Browser.WaitForReady` call.)

10. In the **Enable bitmap capture** group box, check the appropriate check boxes to specify when TrueLog should capture screenshots of the application under test.

- **As specified on action settings tab** – Enables the capture of bitmaps for each action type you select on the **Action Settings** page of the **TrueLog Options** dialog box. This setting also triggers a bitmap capture each time a window becomes active in your application.
- **On error** – Captures a bitmap when an error occurs in your testcase. If you select this option, choose the attributes for the bitmap:
    - **Window only** – Captures a bitmap of the active window following an error.
    - **Desktop** – Captures a bitmap of the entire desktop following an error.
- **Before error bitmap** – Captures an additional bitmap before an error occurs.

  In all cases, bitmaps are linked with the resulting **Error** node in the TrueLog Explorer menu tree. Bitmap files are included in TrueLog (`.xlg`) files.

11. Click the **Action Settings** tab to select the scripted actions in your tests that should be included in TrueLogs.

    None of these actions are selected by default when you enable TrueLog.

    a) Check the check box next to each action that you want to include in TrueLog.

    With the exception of **Click** and **Select**, each action corresponds to a 4Test method.

    Click records mouse clicks on many controls, such as PushButton, ScrollBar, TextField, and HtmlLink. To record **Click** methods on a CheckBox, choose **Select**, not **Click**.

    Select records actions on multiple methods of multiple control types, including ListBox, TreeView, ComboBox, RadioButton, and CheckBox. Here is a partial list of what **Select** records:

    - **ListBox, TreeView** – Select, DoubleSelect, SelectList, SelectRange, Click
    - **Click** – Select, Click
    - **CheckBox** – Check, Uncheck, Toggle, SetState, Click

    b) *Optional*: For each action that you include, in the **Bitmap** column, select the point in time when you want bitmaps to be captured:

    The default setting for all actions is **None**. Be aware that capturing bitmaps increases TrueLog file size.

    - Before the selected actions occur
    - After the selected actions occur
    - Before and after the selected actions occur

    When enabled, these actions appear as nodes in the TrueLog Explorer menu tree.

12. Click **OK**.

Configuring TrueLog Options Using 4Test

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 SilkTest 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**.

Opening a TrueLog in TrueLog Explorer

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

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

1. Choose **Start ➤ Programs ➤ Silk ➤ SilkTest <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 SilkTest testcase of any error that you encounter to correct the error or to update information.

Opening SilkTest Files

You can link directly from any TrueLog API node to the corresponding line in a SilkTest file to correct errors or to update information.

1. Click an action node or error node in an open TrueLog.
2. Click the **4Test** tab.
3. Click a line in the 4Test call stack.
   SilkTest opens to the particular line of the file that corresponds to the clicked line.

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 SilkTest.

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 SilkTest 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**.

SilkTest Performance Considerations

Logging bitmaps and controls with TrueLog may adversely affect the performance of SilkTest. 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.

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

📝 **Note:** TrueLog Explorer for SilkTest 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 date 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.

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.

Disabling HTML Code Binary View

Auto view mode is set by default to automatically select the best data-representation view for captured data. To view HTML source code in a readable format, you must disable auto view mode.

1. Choose **Settings ➤ Options**.
   The **Options** dialog box opens to the **Display** page.
2. Uncheck the **Auto view mode** check box.
3. Uncheck the **Show binary data** check box.
4. Click **OK**.

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