JavaTest™ Harness 4.5

Graphical User Interface User's Guide

ORACLE®

Please
Recycle

Adobe PostScript™

Contents

Preface

This manual describes how to use the Graphical User Interface (GUI) provided by the JavaTest™ harness (the harness) to run tests of the test suite, browse results, and write reports test results.

The harness provides two User's Guides, the *Graphical User Interface User's Guide* and the *Command-Line Interface User's Guide* . If your test suite uses the JavaTest agent to run tests, the *JavaTest Agent Users' Guide* might also be included.

This User's Guide is also provided by the harness in an online version. The online version of the User's Guide differs from the PDF in the following areas:

- In the online version, all of the harness User's Guides are merged into a single document.

- In the online version, the search function provides a list and ranking of all matching text strings found in the complete harness documentation set.

- The online version can be opened either from the command line in a viewer (without opening the harness GUI) or from within the harness GUI.

- In the online version, hypertext links and navigation bars are used instead of page and section references.

Note – Displaying the online version of the User's Guide does not require the installation of any additional software (such as a web browser). The viewer is provided by the harness.

Before You Read This Book

To fully use the information in this document, you must have a thorough knowledge of the topics discussed in the documentation delivered with your test suite.

How This Book Is Organized

Chapter 1 introduces the harness, describes the basic topics that users should be familiar with before starting the harness on a test system, describes how to start the harness.

Chapter 2 describes the Quick Start process.

Chapter 3 describes the GUI menus, the tools, and operations that can be performed from the harness desktop.

Chapter 4 describes how to create and manage configurations used by the harness to run tests.

Chapter 5 describes how to create and edit templates used .

Chapter 6 describes how to use templates to create configurations.

Chapter 7 describes how to use the Test Manager tool to run tests and monitor results

Chapter 8 describes how to use the Test Manager tool to browse test information.

Chapter 9 describes how to use the Test Manager tool to generate test reports.

Chapter 10 describes how to use the Report Converter tool to merge test reports.

Chapter 11 describes how to use the Test Results Auditor tool to audit test results.

Chapter 12 describes how to use the GUI tools and components to troubleshoot a test run.

Using System Commands

This document does not contain information on basic system commands and procedures such as shutting down the system, booting the system, and configuring devices.

See one or more of the following for this information:

- *Solaris Handbook for Sun Peripherals*
- AnswerBook2™ software online documentation for the Solaris™ operating environment
- Other software documentation that you received with your system

Typographic Conventions

This User's Guide uses the following typographic conventions:

Typeface	Meaning	Examples
`AaBbCc123`	The names of commands, files, and directories; on-screen computer output	Edit your `.login` file. Use `ls -a` to list all files. `% You have mail.`
`AaBbCc123`	What you type, when contrasted with on-screen computer output	`%` **`su`** `Password:`
AaBbCc123	Book titles, new words or terms, words to be emphasized	Read Chapter 6 in the *User's Guide*. These are called *class* options. You *must* be superuser to do this.
	Command-line variable; replace with a real name or value	To delete a file, type `rm` *filename*.

Shell Prompts

Examples in this User's Guide contain the following shell prompts:

Shell	Prompt
C shell	*machine_name*`%`
C shell superuser	*machine_name*`#`
Bourne shell and Korn shell	`$`
Bourne shell and Korn shell superuser	`#`

Related Documentation

The following documentation provides additional detailed information about the harness:

Application	Title
JavaTest harness command-line interface	*Command-Line Interface User's Guide*
JavaTest agent (optional)	*JavaTest Agent User's Guide*

Accessing Documentation Online

The Oracle Technology Network enables you to access Java ME technical documentation on the Web:

http://download.oracle.com/javame/index.html

We Welcome Your Comments

We are interested in improving our documentation and we welcome your comments and suggestions. Provide feedback at `javamedocs_us@oracle.com`.

1

Introduction

The harness provides two interfaces, a Graphical User Interface (GUI) and a command-line interface. The GUI provides a desktop that contains a set of menus and interactive tools that you use to configure and execute tests, control and monitor agents, and create reports.

The harness also provides a command-line interface that enables users to accomplish many of the tasks performed in the GUI.

The command-line interface also enables the user to display the online version of the User's Guide without opening the GUI. One means of displaying the online User's Guide (test suites might provide other means) is to type the following command:

```
java -jar testsuite/lib/javatest.jar -onlinehelp
```

In the command line, *testsuite*/lib/ represents the location of the directory where the test suite installed the javatest.jar file.

See the *Command-Line Interface User's Guide* for a description of the command-line interface.

GUI Features

The features of the GUI include a set of interactive tools and utilities that enable the user to perform the following tasks:

- Configure, run, and monitor tests on a variety of test platforms (such as servers, workstations, browsers, and small devices) with a variety of test execution models (such as API compatibility tests, language tests, compiler tests, and regression tests).
- Evaluate and analyze test results.
- Create template configuration files.

- Generate and view test reports that summarize test runs.
- View the contents of log files.
- Monitor agents (only enabled when the test suite provides an agent). The JavaTest harness supports the use of an agent to run tests on systems that can't run the JavaTest harness.
- Display online user documentation in a viewer that provides context sensitive help, full-text search, and keyword search information.

Configuration Information

The harness uses the Configuration Editor to collect configuration information about how tests are run on a specific test platform. The Configuration Editor saves the information in a configuration file (`.jti` file) that the harness uses to run tests. By using configuration files, the harness can run programs on a variety of Java platforms. The harness writes the test results in the work directory associated with that configuration and test suite. See the Glossary for detailed descriptions of the terms `.jti` file, *work directory*, and *configuration file*.

Before Starting the Harness

Before you start the harness on a test system, you must have a valid test suite and Java Development Kit 6.0 or higher installed on your test system. See your test suite documentation for information about installing the test suite on your test system. Refer to `http://www.oracle.com/products` for information about installing the current Java Development Kit on your test system.

You must also understand how your test group uses or intends to use the harness in its test system. For example, consider the following questions:

- Does your test group use the harness and one or more agents to run distributed tests?

  If you use an agent to run tests, you must also install the agent on the platform being tested. If you intend to use the JavaTest agent, see *JavaTest Agent User's Guide* for detailed information about installing the harness agent on a test platform. If you use a custom agent, refer to your test suite documentation for detailed information about installing the agent.

- Does your test group use configuration files and templates from a central location, or does it use individual configuration files customized for each test run?

   If your group intends to use configuration files and templates from a central location, the location must be configured for the harness to access the template files.

- Does your test group run the harness from a central location or from local installations in the test system?

If your group uses a central location for the test system, you must be able to access the test system at that location.

Installation and Runtime Security Guidelines

It is extremely important to note that the JavaTest installation and runtime system is fundamentally a developer system that is not specifically designed to guard against any malicious attacks from outside intruders. If sample code or tests you author make a network call, you can expose the JavaTest operating environment during execution. For this reason, it is critically important to observe the precautions outlined in the following security guidelines when installing and running JavaTest.

The harness itself is self-contained in `javatest.jar`. If desired, the following optional parts of the binary distribution can be deleted:

- The directories `linux`, `solaris`, and `win32` each contain a subdirectory named `bin`, which contains a script named `javatest`. All of these directories and files are provided for convenience and can be deleted without harm. (If the `javatest` launch script is deleted you can start the harness by pointing a JVM at the `javatest.jar` file.)

- Examples in the `examples/` directory, which is an optional binary component and may not be present.

- Documentation in the `doc/` directory.

- `lib/jt-junit.jar` which is an optional binary component and may not be present.

To maintain optimum network security, JavaTest can be installed and run in a "closed" network operating environment, meaning JavaTest is not connected directly to the Internet, or to a company Intranet environment that could introduce unwanted exposure to malicious intrusion. This is the ideal secure operating environment when it is possible.

JavaTest does not require an "Intranet" connection that supports network connections to systems outside the JavaTest architecture to intra-company resources, but, for example, some Java ME applications in a test suite might use an HTTP

connection. If JavaTest or applications launched from JavaTest are open to any network access you must observe the following precautions to protect valuable resources from malicious intrusion:

Install JavaTest behind a secure firewall that strictly limits unauthorized network access to the Java Runtime file system and services. Limit access privileges to those that are required for JavaTest usage while allowing all the I-directional local network communications that are necessary for JavaTest functionality. The firewall configuration must support these requirements to run JavaTest while also addressing them from a security standpoint.

Follow the principle of "least privilege" by assigning the minimum set of system access permissions required for installation and execution of JavaTest.

Do not store any data sensitive information on the same file system that is hosting JavaTest.

To maintain the maximum level of security, make sure the operating system patches are up-to-date on the JavaTest host machine.

Starting the Harness

The harness provides a set of command-line options that you can use to specify the configuration and properties of the GUI when starting the harness. See the *Command-Line Interface User's Guide* for detailed information about command-line options used to specify how the harness starts.

See your test suite documentation for specific information required to start the harness on your test system.

Online User's Guide

The harness also provides an online version of the User's Guide. The online version can either be displayed without starting the harness or from inside the GUI. The online User's Guide merges all of the following PDF User's Guides into one document:

- *Graphical User Interface User's Guide* contains information about using the GUI.
- *Command-Line Interface User's Guide* contains command-line and utility information.

- *JavaTest Agent User's Guide* (available only if supplied by the test suite) contains information about the optional agent.

Throughout the online version, the icon is used to identify additional information that might be required to fully understand a topic. In the PDF format, this information is presented in the form of notes.

The following sections describe the different actions used to display the online version of the User's Guide and other online documentation.

Accessing Online Documentation

The harness enables you to display the User's Guide and other forms on online documentation without starting the GUI. The viewer is provided by the harness and requires no additional software. To display the online User's Guide, type the following command from the directory containing the `javatest.jar` file:

```
java -jar javatest.jar -onlinehelp
```

You can also display the User's Guide and other forms on online documentation inside the GUI. The following table describes user actions in the GUI that display the User's Guide and other information about the harness.

TABLE 1   Accessing the Online Guide

Action	Description

TABLE 1   Accessing the Online Guide

F1 key	Press the F1 key to display information about the window that has keyboard focus.
	You must establish keyboard focus in a window before pressing the F1 key. To establish focus, you might have to highlight something in the window.
Help menu	The Help menu lists the following types of online documentation:
	• Information for the window that has focus. To display information for the window that has focus (such as the Test Manager window), choose *window_name* from the menu.
	• Information for the harness. To display the User's Guide for the harness, choose JavaTest Harness Online Help from the menu.
	• Optional information (such as, users' guide and release notes) provided by the test suite.
	• Information consists of information about the harness and the virtual machine. This information is collected and provided by the harness.
Help buttons	Click the Help button in a dialog box for information about how to use that dialog box.

2

Important Security Information

It is important to note that the harness is fundamentally a developer system that is not specifically designed to guard against malicious attacks. This section describes known harness vulnerabilities that can be exploited by authorized or unauthorized users or others who have access to the network where the harness runs. When installing and operating the harness, consider these vulnerabilities and take action to mitigate potential threats.

Note: See the *Release Notes* for additional security information.

File Access Risks

The harness reads, writes, and deletes files and can therefore potentially expose or damage information stored in files.

Path Vulnerabilities

The harness and its utilities do not prevent users from accessing sensitive files that are accessible from the computer on which the harness is running - including attached/mounted remote filesystems. A harness user can therefore directly or indirectly examine file contents, overwrite contents, or delete them. To mitigate against accidental or intentional misuse of sensitive files, consider setting file permissions to give the least feasible access to harness users.

There are no direct options in the harness for removing a file, directory, or folder, but the harness can automatically remove files that it believes it owns and are corrupt or in need of replacement. Automatic removal or replacement generally occurs only

inside the harness preferences metadata storage area (`~/.javatest` or `user_home\.javatest`) and inside work directory locations directly designated by the user.

There are at least two ways a user can indirectly give the harness access to an arbitrary file that the harness might automatically remove or replace.

- A configuration (`.jti`) file can contain a "hint" that enables the harness GUI or CLI to find what the harness *assumes* is a work directory.
- A desktop file in the preferences metadata storage area can contain a path to what the harness *assumes* is a work directory. The harness GUI or CLI automatically opens the desktop file.

File Creation Permissions

The harness is intended to be run by users who are trusted to properly handle files they have access to. The harness does not attempt to override any file permissions that are in place to protect files or restrict the user. Additionally, the harness creates files with whatever permissions are the default for the user running the harness (on Unix systems, the `umask`, etc). If you want to restrict access to files created by the harness, ensure that user default file creation permissions are set accordingly.

Network Access Risks

The harness and related components are intended to be run in a semi-trusted environment. Never expose the harness or its utilities directly to the Internet, which provides a path for malicious intrusion. The most secure environment is a standalone machine with no network access, if this is feasible for the testing you are performing. However, because the test harness agent configuration and some test suites require a network, a more realistic environment is a closed intranet, ideally one that is physically isolated from organizational intranets and sensitive information.

Remote Agent Risks

The test harness includes the JavaTest agent, which is a remote execution framework. The agent requires open communication ports on the JavaTest host computer and on the remote computer (device); therefore you must ensure that both machines are protected from malicious attack. For the most secure operation, connect the host and remote computers only to a protected intranet on a physically isolated network.

Host Name (DNS) Lookup Vulnerabilities

The harness sometimes uses Domain Name Services (DNS), or similar from your system, to transform a host name into an IP address. Invalid information returned by this lookup might cause the harness to connect to an address you did not intend. When possible, the harness GUI displays the returned IP address. Users should verify that the address is appropriate, for example, lies within a range the corresponds to your network. If you need assistance validating IP addresses, contact your network administrator.

Command Injection Vulnerability

As part of the normal process of harness configuration and testing, there are places where the user is allowed to specify arbitrary commands (with parameters) to execute - for example, to start a test, to start a Java virtual machine being tested, or to start a service. The harness does not screen these commands for malicious or accidental usage. The harness offers no protection from the effects of any command initiated by a user or a test. Be sure that users are trained and trusted to specify proper commands properly, and to verify that commands issued by test suites are harmless. An incorrect or malicious command can cause serious damage.

The harness does not attempt to run commands with higher privileges than that of the current user or the privileges inherited when the new command/process executes.

3

Quick Start

The harness includes a Quick Start wizard that enables users to quickly create combinations of test suite, work directory, and configuration files and open them in properly configured Test Manager windows. The harness opens the wizard when one of the following conditions occur:

- The harness cannot restore an existing desktop.
- A user includes a command-line option in the command to start the GUI.
- A user opens a new Test Manager window.

In each case, the wizard enables the user to specify one of the following tasks:

- Start a New Test Run.
- Resume Work on a Test Run.
- Browse the Contents of a Test Suite.

Because each of these tasks has a different set of test suite, work directory, and configuration file requirements, the wizard presents the user with a set of questions that collect the information required for that specified task. When the wizard has collected the required information, it opens a new Test Manager window that is properly set up to perform the specified task.

Starting a New Test Run

The harness requires that you have a test suite, a work directory, and a valid configuration loaded in the Test Manager before running tests. To properly set up the Test Manager, when you choose Start a New Test Run, the wizard displays a series of questions that prompt you to choose the type of configuration used (load a template or create a new configuration), specify a test suite, and choose a work directory. See the following topics for additional information about templates, test suites and work directories:

- [Working With Templates](#)
- [Opening a Test Suite](#)
- [Opening a Work Directory](#)
- [Creating a Work Directory](#)

In the last panel, the wizard displays two options: automatically open the Configuration Editor and automatically begin the test run.

If you choose not to automatically open the Configuration Editor when the wizard closes, the harness displays the Test Manager. In either case, before the harness begins to run tests, it verifies that the configuration is complete. If configuration values are missing or invalid, the harness displays a dialog box describing the problem before opening the Configuration Editor.

See the following topics for additional information about working with configurations:

- [Creating a Configuration](#)
- [Editing a Configuration](#)

Resuming Work on a Test Run

If you choose to resume work on a test run, the wizard only prompts you for the work directory. Because each work directory is mapped to a test suite and a configuration file, you do not provide any additional information to the harness. See [Opening a Work Directory](#) for additional information about using an existing work directory.

In the last panel, the wizard displays two options: automatically open the Configuration Editor and automatically begin the test run.

If you choose not to automatically open the Configuration Editor when the wizard closes, the harness displays the Test Manager. In either case, before the harness begins to run tests, it verifies that the configuration is complete. If configuration values are missing or invalid, the harness displays a dialog box describing the problem before opening the Configuration Editor.

Browsing the Contents of a Test Suite

If you want to view the tests in a test suite without running them, the wizard only requires that you specify the test suite. While the wizard provides you with the option of creating a configuration file for the test suite, this is not required. If you choose not to create a configuration file for the test suite, the harness closes the wizard and opens a Test Manager window containing the test suite.

See Opening a Test Suite for a additional information about loading a test suite.

Note – Until you provide both a work directory and a valid configuration file, you cannot run tests.

Opening a Test Suite

You can open a test suite from either the Test Manager or the Quick Start wizard (if it is enabled and you are starting a new test run).

Note: Some test suites do not implement the optional Quick Start wizard. It won't be available if the test suite architect disabled it.

▼ Open a Test Suite

1. Click the Browse button in the Quick Start wizard Test Suite panel or choose File > Open Test Suite from the Test Manager menu bar.

2. Use the dialog box to navigate to the test suite.

3. Click the test suite icon or enter its name in the text field.

4. Click the Open button.

The harness loads the test suite in a new Test Manager.

Before you can run tests, you must open or create a work directory and load or create a configuration. If you did not use the Quick Start wizard to open the Test Manager, you must perform these operations separately. See the following topics for detailed information about using work directories and configurations:

- [Opening a Work Directory](#) provides information about using an existing work directory for the test run.
- [Creating a Work Directory](#) provides information about creating a new work directory for the test run.
- [Working With Templates](#) provides information about using a template in the test run.
- [Creating a New Configuration](#) provides information about creating a new configuration for the test run.
- [Editing a Configuration](#) provides information about changing configuration values used for the test run.

Opening a Work Directory

Each work directory is associated with a specific test suite, a configuration, and possibly a template. Each time the harness runs tests, it creates test result files that store the information collected during the test run. The harness stores these files in the work directory for all test runs of the test suite with that configuration.

You can open a work directory from either the Quick Start wizard (when resuming work on a test run) or the Test Manager.

Use File > Recent Work Directory to quickly open a directory you have used before.

▼ Open a Work Directory with the Quick Start Wizard

1. Click the Browse button in the Quick Start wizard Work Directory panel.

   The harness displays a Work Directory chooser.

2. Use the chooser to navigate to the location of the work directory.

3. Click on the work directory icon or enter the name of the work directory in the text field.

4. Click the Open button.

5. Complete the Quick Start wizard.

When you complete the Quick Start wizard, the harness opens a Test Manager window containing the test results for all test runs of the associated test suite and configuration.

▼ Open a Work Directory with the Test Manager

1. Choose File > Open Work Directory from the Test Manager menu bar.

   The harness displays the Open Work Directory dialog box.

2. Use dialog box to navigate to the location of the work directory.

3. Enter the name of the work directory in the Path field or click its name in the tree.

4. Click the Open button.

   The harness performs one of the following actions:

- Displays the test results from the work directory for all test runs of the current test suite in the Test Manager *if* both of the following conditions are true:

- The current test suite and configuration are associated with the work directory.

- The current test suite is not associated with any other open work directory.

- Opens a new Test Manager window for the work directory, test suite, and configuration if the work directory is not associated with the current test suite in the Test Manager.

Creating a Work Directory

Every work directory is associated with a specific test suite and a configuration file after one is created or opened. Each time the harness runs tests, it creates test result files that store the information collected during the test run. The harness stores these files in the work directory for all test runs of the test suite with that configuration.

You must create a new work directory for each new test suite. You might also choose to create a new work directory to perform multiple test runs of the same test suite but with different configuration files. If you try to use an archive work directory that is no longer considered valid by the harness you must also create a new work directory.

You can create a work directory from either the Test Manager or the Quick Start wizard (if it is enabled and you are starting a new test run). The harness also prompts you to specify a work directory if you begin a task requiring a work directory without having specified one.

▼ Create a Work Directory with the Quick Start Wizard

Note: Some test suites do not implement the optional Quick Start wizard. It won't be available if the test suite architect disabled it.

1. Click the Browse button in the Quick Start wizard Work Directory panel.

The harness displays a Work Directory chooser.

2. Use the chooser to navigate to the location of the new work directory.

3. Enter the name of the work directory in the text field.

4. Click the Create button.

5. Complete the Quick Start wizard.

When you complete the Quick Start wizard, the harness opens a new Test Manager window and creates a work directory for the test suite, template, and configuration file combination.

▼ Create a Work Directory with the Test Manager

1. Choose File > Create Work Directory from the Test Manager menu bar.

The harness displays the Create Work Directory dialog box.
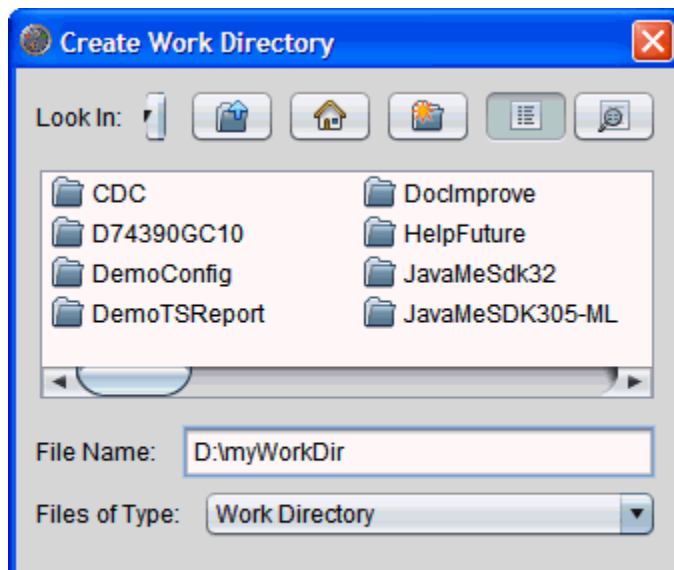
2. Enter the name of the new work directory in the Work Directory Name text field.

3. Click the Browse button and use the File Chooser to select a location for the work directory.

Note – Do not enter the directory name in the file chooser. The harness uses the name from the Work Directory Name text field to create it for you.

Depending on the test suite, the template option might not be allowed (templates not enabled), might be required (cannot select "No Template"), or might be optional (operates as described) in the following step. If you select a template, it will be associated with that work directory for all configurations that are created for that work directory.

4. Use the No Template or Choose a Template buttons to set the template options for the work directory.

5. Check the Launch Configuration Editor box to open the Configuration Editor after the dialog box closes.

6. Click the Create button.

When you create the work directory, the harness opens a new Test Manager window and creates a work directory for the test suite, template, and configuration file combination.

4

Graphical User Interface

The GUI consists of a desktop containing a set of tools that enable the user to configure and run tests, monitor test status, evaluate and analyze test results, generate reports, and view log file contents.

The harness attempts to restore the previous desktop when it displays the GUI. If you used a command-line option (such as, -newdesktop) to start the harness or if a previous desktop does not exist, the harness displays the Quick Start wizard. See the *Command-Line Interface User's Guide* for detailed information about using specific values and setup commands in the command line to start the GUI.

The Quick Start wizard guides you through the process of setting up the GUI for a test run. See Quick Start for information about using the Quick Start wizard.

For detailed information about desktop functions and features, see the following topics:

- Layout Styles
- Menus
- Setting Preferences
- Displaying Online Information
- Keyboard access
- Tools

For information about using desktop tools to perform tasks, see the following topics:

- Configuring a Test Run
- Creating and Editing Templates
- Working with Templates
- Running Tests
- Browsing Test Run Information
- Creating and Displaying Reports
- Merging and Converting XML Test Reports

- Troubleshooting with the GUI

Menus

The GUI provides two types of menus: desktop and tool. The layout style determines how the menus are displayed in the GUI. The following table describes the two menu types and how the desktop style determines how they are displayed.

TABLE 2   GUI Menus

Menu Type	Description
Desktop	Desktop menus are always available to the user in the desktop menu bar. See GUI Layout for a detailed description of the GUI layout options available to the user. They include the following menus: • File • Tool • Windows • Help
Tool	These menus are unique to specific tools and are only available to the user when that tool window is open. The appropriate tool menus are merged with the desktop menus in a single menu bar. See Agent Monitor Tool for a description of the Agent Monitor tool menus. If your test suite uses the harness agent, see the *JavaTest Harness Agent User's Guide* for a description of the Agent Monitor. See Report Converter Tool for a description of the Report Converter tool menus. See Test Manager Tool for a description of the Test Manager tool menus.

File

Use the File menu to open files, set user preferences, and exit from the harness. The contents of the File menu change dynamically, based on the context of the desktop. The harness only enables menus when they can be used. The following table describes the contents of the File menu.

TABLE 3   File Menu Contents

Menu Item	Description
Open the Quick Start Wizard	**Note:** Some test suites do not implement the optional Quick Start wizard. It won't be available if the test suite architect disabled it.
	Opens the optional Quick Start wizard for you to use for the following tasks:
	• Start a new test run by selecting a test suite, work directory, and a configuration file. If the configuration file is new or incomplete, the harness opens the Configuration Editor after the wizard closes.
	• Resume work on a previous test run or browse a test suite without running tests.
	• Browse the contents of a test suite.
Create Work Directory	Opens a dialog box that you can use to create a work directory. When you choose to create a new work directory, the harness associates it with the test suite opened in the Test Manager tool window. The harness creates test result files in the work directory that contain all of the information gathered by the harness during test runs of the test suite.
	See Creating a Work Directory for detailed information about creating a work directory.
Open > Work Directory	Opens a dialog box that you can use to open an existing work directory. Each work directory is associated with a specific test suite and contains information from previous test runs. The test result files in the work directory contain all of the information gathered by the harness during test runs.
	When you choose to open an existing work directory, the harness performs the following actions:
	• It associates that work directory with the test suite opened in the Test Manager tool window *only* if the test suite is both a match and has no other work directory already open.
	• If it cannot associate the work directory with the open test suite, the harness opens a new Test Manager and loads both the work directory and its associated test suite.
	See Opening a Work Directory for detailed information about opening a work directory.

TABLE 3 File Menu Contents

Open > Test Suite	Opens a dialog box you can use to choose a test suite. When you open the test suite, the harness loads it into new Test Manager. See <u>Opening a Test Suite</u> for additional information.
Recent Work Directory	Displays a list of work directories that have been opened. Choose a work directory from the list to open a new instance of it in the current session.
Preferences	Opens the Preferences dialog box for you to set the display and functional options of the harness. See <u>Setting Preferences</u> for detailed information about setting preferences in the GUI.
Close	Shown in the File menu. Closes the current window without exiting from the harness. Closing a Test Manager closes the work directory.
Exit	Exits from the harness. If you have set your preferences to save your current desktop, your current desktop is saved so that all open windows, test suites, work directories and configurations are restored the next time that you start the harness. See <u>Setting Preferences</u> for detailed information about setting preferences in the GUI.

Tools

The following table describes the tools provided by the harness.

TABLE 4   Tools

Tool	Description
Report Converter	Opens the Report Converter tool that you use to generate and view reports of results from multiple test reports.
Test Manager	Opens the Test Manager tool that you use to do the following tasks: • Open and create work directories. • Add or modify information that the harness uses when running your tests. • Run the tests of a test suite. • Monitor tests and test results while they are being run. • View test environment settings of your configuration. • View the contents of an exclude list. • Browse completed tests and test results. During a test run, the icons in the Test Manager change to reflect the test status. You can also use the window during and after a test run to browse information about individual tests.
Agent Monitor (Optional)	The Agent Monitor tool is only available for use when the test suite is configured to use an agent. If your test suite includes the JavaTest Harness agent, see the *JavaTest Harness Agent User's Guide* for detailed information about the agent and the Agent Monitor tool.

Windows

The Windows menu also provides a list of all windows currently open in the harness. You can navigate to any open window by clicking on its name in the list.

Help

Use the Help menu to display user information about the tool window and the harness, the available test suite documentation, version information about the harness, and information about the current Java runtime environment.

TABLE 5   Contents of the Help Menu

Menu Item	Description
Active Window Name	Displays user information about the active window. This menu item is only available in an open tool window.
Online Help	Displays the online version of the harness User's Guide. This version combines the GUI, command-line interface, and the agent User's Guides (when applicable) into one document.
Test Suite Documentation (Optional)	If the test suite provides online documentation, the harness lists it here. Click the document name to display it in the viewer.
About the JavaTest Harness	Displays information about this release of the harness.
About the Java Virtual Machine	Displays information about the runtime used to run the harness.

Setting Preferences

Choose File > Preferences from the menu bar to open the harness preferences dialog. Use this dialog to set the display and functional options of the harness.
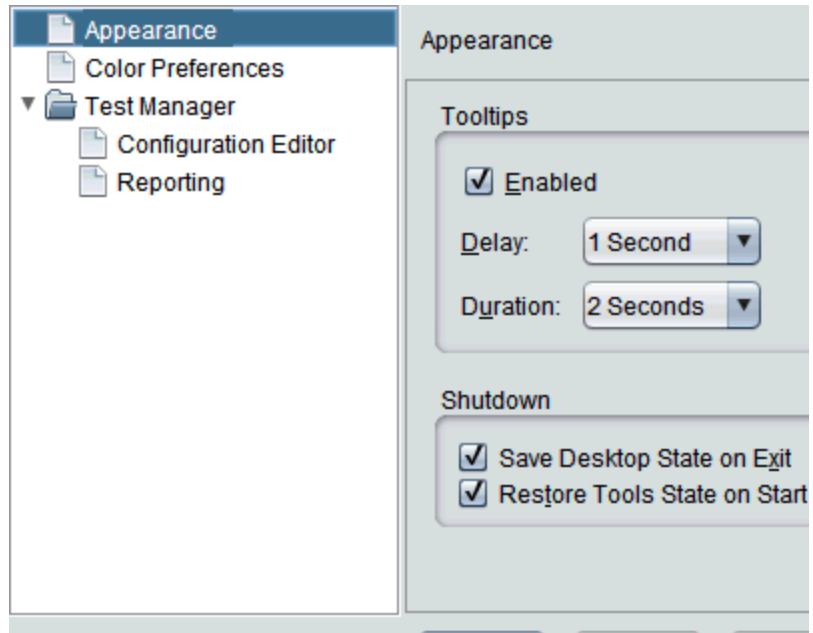
The following table describes the contents of the Preferences dialog box.

TABLE 6   Preferences Dialog Box Contents

Preference Category	Description
Appearance	Sets how the desktop displays tool windows and tool tips. See Appearance Preferences for detailed information.
Color Preferences	Sets the color preferences for default input and input status warnings. See Color Preferences for detailed information.
Test Manager	Sets the Test Manager tool bar property. See Test Manager Preferences for detailed information.

Appearance Preferences

Change the desktop appearance by opening the Appearance folder and setting the following options:

■ Tool Tip Options

- [Shutdown Options](#)
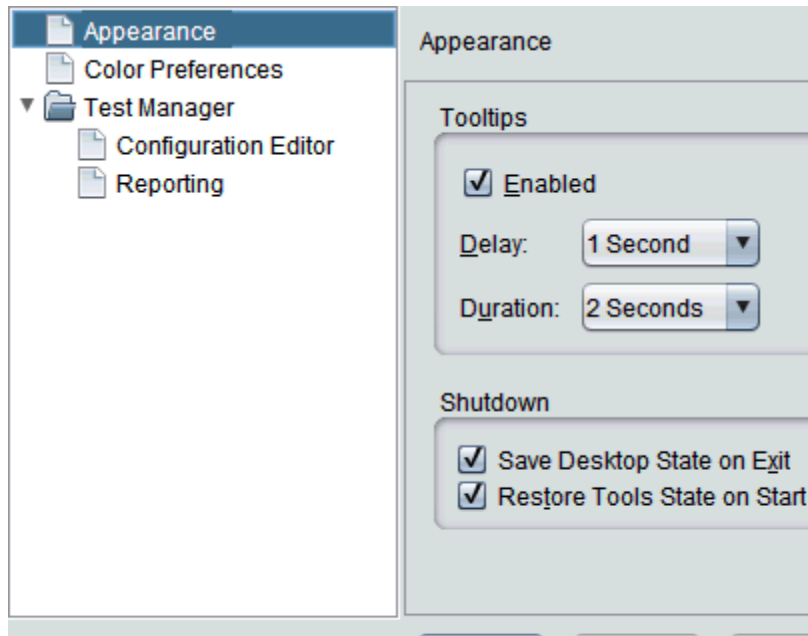
See [GUI Layout](#) for a detailed description of the tabbed interface.

Tool Tip Options

You can set the tool tip options from the Appearance category of the Preferences dialog box.

The Tool Tips area contains combo boxes and a check box that you can use to specify how tool tips function in the desktop. The following table describes the available tool tip options.

TABLE 7   Setting Tool Tip Options

Option	Description
Enabled	Use the check-box to enable or disable tool tips for the GUI.
Delay	Use the combo box to select the delay interval before displaying tool tips.
Persistence	Use the combo box to select the duration that the GUI displays a tool tip.

Shutdown Options

Check the Save Desktop State on Exit option to save your current desktop for use in a future test session. This default is on.

When you exit this option saves the current view filter information, recent work directories, and recent configuration information. If a desktop file exists, it will remain.

If you uncheck Save Desktop State on Exit and close the harness, the current view filter information is not saved. The next time you open a test session the harness opens the Quick Start Wizard (if the test suite author has enabled it) or an empty harness tool.

Check Restore Tools State on Start to restore saved desktop settings. The default is on. If this option is not checked, the recent work directories and configuration information are restored but the tools are not.

If Restore Tools is set **on** and Save Desktop State on Exit is set **off**, the harness will always start with one tools set (although **-newdesktop** can be used to remove all data).

Test Manager Preferences

To change the Test Manager preferences, select File > Preferences > Test Manager.

- Test Manager Preferences
- Configuration Editor
- Reporting

The following preferences are set when the Test Manager folder is selected as follows:
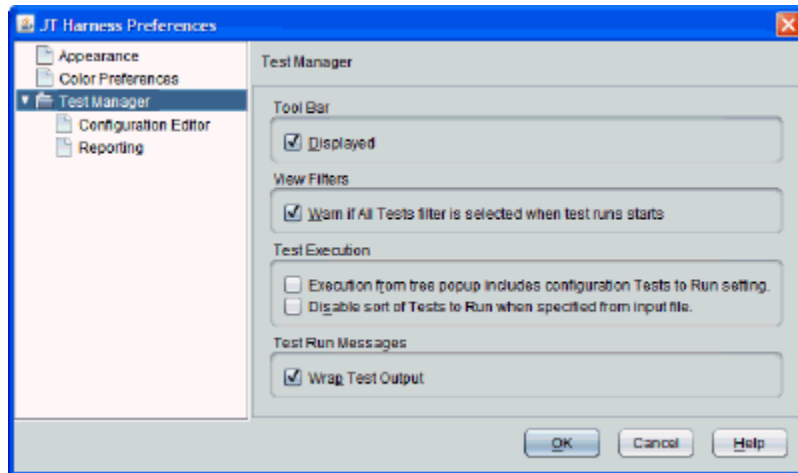
- Tool Bar: Check Displayed to see the toolbar. If you enable or disable this option you must restart JavaTest to see the change.

- View Filters: Enables a warning message that is displayed if the <u>All Tests filter</u> is selected for a test run.
- Test Execution:
- Execution from tree popup includes configuration Test to Run setting:
- Disable sort of Tests to Run when specified from input file: This allows you to set the test execution order. Traditionally tests are run in the order they appear in the main test tree. This order is generally determined by the test suite architect. By selecting this option, the order execution will follow that in the input file for the Tests to Run section in the configuration. That is, the tests will be executed in the order they appear in that file, rather than being sorted and then executed. The default state of this checkbox is unchecked. See
- Test Run Messages: Wrap Test Output in the <u>Test Run Messages Pane</u>.

Configuration Editor

Check the More Info box to view the More Info pane in the configuration editor. Same as choosing View > More Info, except the preference will remain in future sessions.
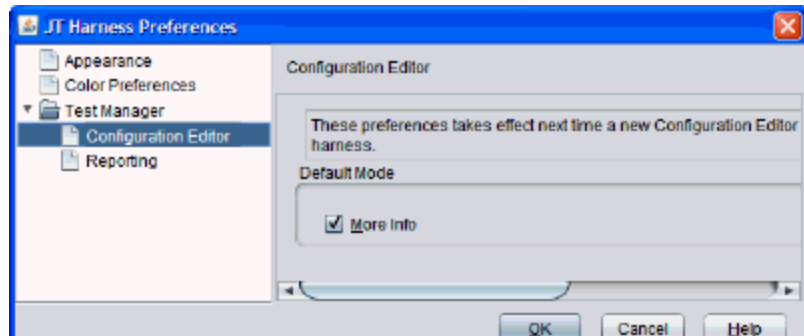
Reporting

Set the bug prefix URL for KFL reports.
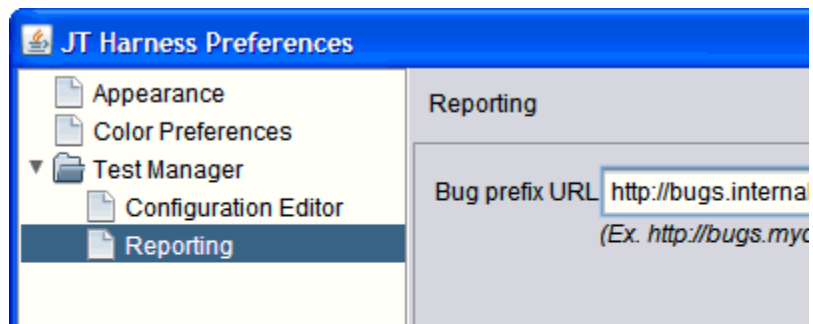
Displaying Online Information

You can display available harness and test suite information in the following ways:

- Help menu

- Help buttons

- F1 key

Help Menu

The Help menu lists the available harness, test suite, and current runtime information.When multiple test suites are opened, the menu displays a menu item for each document provided by the test suite. Choosing an item from the menu opens it in a separate viewer.

Help Buttons ?

The desktop provides a help button on all window tool bars and in all dialog boxes.

In the tool bar, clicking the help button displays the documentation for that window.

In dialog boxes, clicking the help button displays documentation for that dialog box.

F1 Key

Press the F1 key to see detailed information for the window that has keyboard focus. Establish keyboard focus in a window before pressing the F1 key. In some cases, you might have to highlight a component in the window to establish focus.

Keyboard access

The harness uses standard Java programming language key bindings for keyboard traversal and access of the GUI. See the *Java Look and Feel Design Guidelines* at `http://java.sun.com/products/jlf/` for a detailed description of the standard key bindings for keyboard traversal and access of the GUI.

The window or component must have keyboard focus before you can use keyboard navigation, activation, or shortcuts. Keyboard navigation, activation, and shortcut operations are described in the following topics.

Keyboard Focus

When a component has focus, it is generally displayed with a colored border or changes color. However, some components in the GUI cannot be displayed with a focus indicator. In this case, you must continue to use the keyboard to traverse the GUI until focus is displayed.

Keyboard Shortcuts

Keyboard shortcuts perform both navigation and activation in the same action. The following table lists the keys that are used to access menus and online help.

TABLE 8 Keyboard Shortcuts for Navigation and Activation

Activation Keys	GUI Action
F1	Activates the online help.
F10	Activates the File menu.
Shift F10	Activates the pop-up menu if focus is on a folder or test in the test tree.

Hot Key Shortcuts

The harness provides shortcuts throughout the application for accessing menu titles, menu items, text fields, checkboxes, radio buttons, and command buttons.

Shortcut keys are identified in the following ways:

- **Text at the end of tool tips -** For example, Start Running Tests Alt-S
- **Underlined letters in menus and text buttons -** For example, <u>F</u>ile
- **Underlined letters in labeled fields -** For example, <u>R</u>esponse:

Keyboard Navigation

Keyboard navigation enables you to move keyboard focus from one GUI component to another by using the keyboard without activating the component. The following table lists the keys used for keyboard navigation.

TABLE 9   Keyboard Navigation Keys

Navigation Key	GUI Action
Tab	Navigates to the next focusable component in the GUI. The tab traversal order is generally left to right and top to bottom.
Shift-Tab	Navigates back to the next focusable component.
Control-Tab	Navigates to the next focusable component even if the current component accepts the Tab key as input (such as a text area).
Control-Shift Tab	Navigates back to the next focusable component even if the current component accepts the Tab key as input (such as a text area).
Left arrow	Moves keyboard focus left one character or component. If focus is in the test tree, focus moves up the tree and closes the node. If focus is on the splitter bar (F8 moves focus to the splitter bar), it moves the splitter bar left.
Right arrow	Moves keyboard focus right one character or component. If focus is in the test tree, focus moves sequentially down the tree, opening the node and traversing all tests in a folder. If focus is on the splitter bar (F8 moves focus to the splitter bar), it moves the splitter bar right.
Up arrow	Moves keyboard focus up one line or component. If focus is in the test tree, focus moves sequentially up the tree but does not open any closed folders. If focus is on the splitter bar (F8 moves focus to the splitter bar), it moves the splitter bar left.
Down arrow	Moves keyboard focus down one line or component. If focus is in the test tree, focus moves sequentially down the tree but does not open any closed folders. If focus is on the splitter bar (F8 moves focus to the splitter bar), it moves the splitter bar right.
Page Up	Navigates up one pane of information within a scroll pane.
Page Down	Navigates down one pane of information within a scroll pane.
Home	Moves to the beginning of the data. In a table, moves to the beginning of a row. If focus is in the test tree, moves to the top of the tree.
End	Moves to the end of the data. In a table, moves to the last cell in a row. If focus is in the test tree, moves to the bottom of the tree.
Control-F1	Displays the tool tip information for the GUI object that has focus. Can be used to determine which GUI object has focus.
F6	Shifts focus between left and right panes.

TABLE 9   Keyboard Navigation Keys

F8	Shifts focus to the splitter bar if focus is in the left or right pane.
Control-T	Shifts focus to the next link in a topic or in a report.
Control-Shift-T	Shifts focus to the previous link in a topic or in a report.

Navigation in Hyperlinked Text

Navigating hyperlinks in text areas such as the More Info pane and report viewer requires that you establish focus in the pane itself. After you have established focus inside the pane, use the keyboard navigation keys listed in Keyboard Navigation to navigate in the pane. Because some components in the GUI cannot display focus, you have to use the keyboard to traverse the GUI until you can determine that focus is established inside the pane.

After focus is established in the pane, use Control-T and Shift-Control-T to navigate to the next and previous link in the document. Use Control and Space bar to select (follow) the hyperlink. See Keyboard Activation for a list of keys used for keyboard activation.

Navigation in Folder Pane Status Tabs

Navigating in the folder pane test status tabs requires that you establish focus in the pane itself.

After focus is established in the pane, use the Arrow Up and Arrow Down keys listed in Keyboard Navigation to navigate in the pane. When only a single entry is present in the folder pane, you must use the Home or End key to select the item.

Because some components in the GUI cannot be displayed with focus indicators, you may have to use the keyboard to traverse the GUI until you can determine that focus is established inside the folder pane.

Use the Return or Enter key to navigate to the selected item. See Keyboard Activation for a list of keys used for keyboard activation.

Navigation in the Test Tree

Navigating in a test tree requires that you establish focus in the test tree itself.

After focus is established in the pane, use Arrow Up, Arrow Down, Arrow Left, and Arrow Right keys listed in Keyboard Navigation to navigate in the pane.

Keyboard Activation

After navigating to a component, you can then use the keyboard to activate the component. The following table lists the key that are used to activate GUI components.

TABLE 10 Keyboard Activation

Activation Keys	GUI Action
Enter or Return	Activates the default command button.
Escape	Dismisses a menu or dialog box without changes.
Space Bar	Activates the tool bar button that has keyboard focus.
Shift-Space Bar	Extends the selection of items in a list.
Control-Space Bar	If the item with focus is in a list, it toggles the selection state of the item without affecting any other list selections. If the item is a link, it follows the link.

Tools

The harness provides a set of tools that enable you to perform specific tasks related to running tests, troubleshooting, and reporting test results. The tools provided by the harness include the following:

- Agent Monitor tool - Enables you to monitor the activity of agents connected to the harness.
- Report Converter tool - Enables you to merge XML reports from multiple work directories into a single XML report.
- Test Manager tool - Enables you to perform the following tasks:
- Load, edit, or create a configuration.
- Run and monitor tests.
- Browse test results.
- Create and view reports.

Agent Monitor Tool

If your test suite uses the harness agent, use the Agent Monitor tool to view all agents in a test system that are running tests. Choose Tool > Agent Monitor from the menu bar to open the tool window.

The Agent Monitor window contains two sections: Agent Pool and Agents Currently In Use.
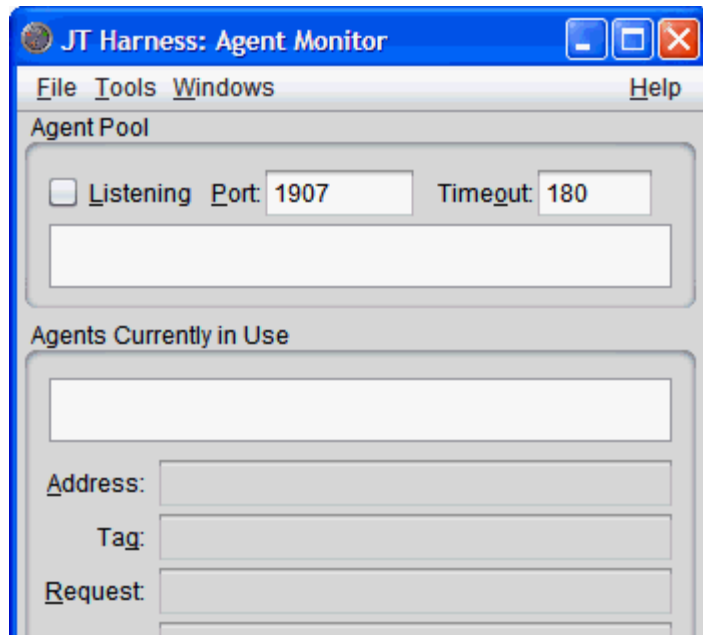
If your test suite uses the harness agent, see the *JavaTest Harness Agent User's Guide* for a detailed description of using the Agent Monitor tool.

Report Converter Tool

You can use the Report Converter tool to merge multiple XML reports created by the JT harness into a single report. To launch the Report Converter tool, choose Tools > Report Converter from the menu bar.
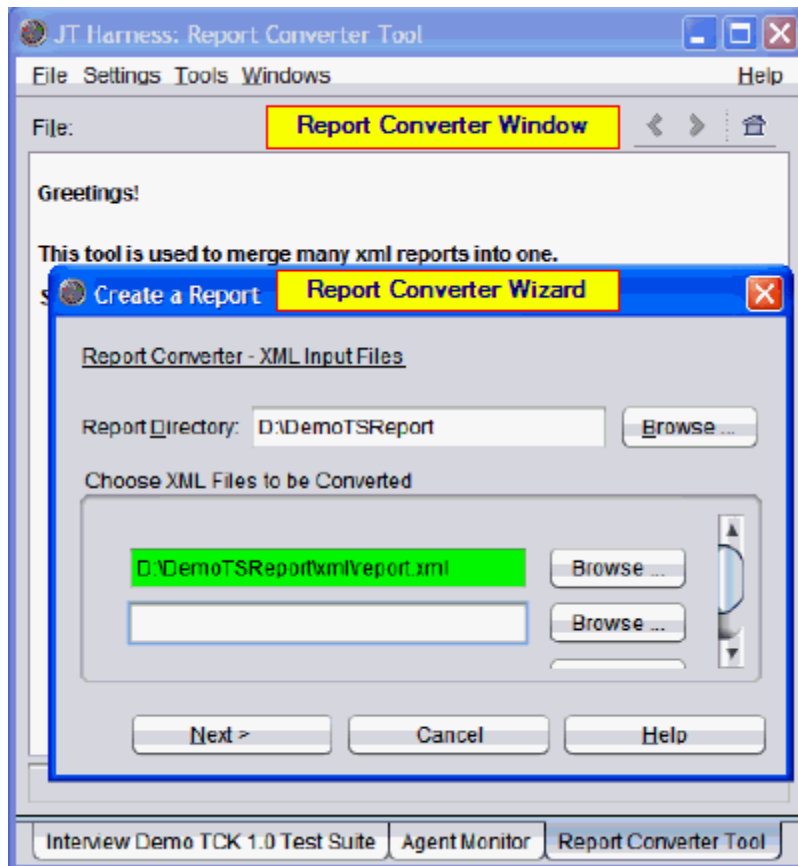
1. Report Converter window

2. Report Converter wizard

See Merging and Converting XML Test Reports for a detailed description of using the Report Converter tool .

Report Converter Window

You can use the Report Converter window to view reports.
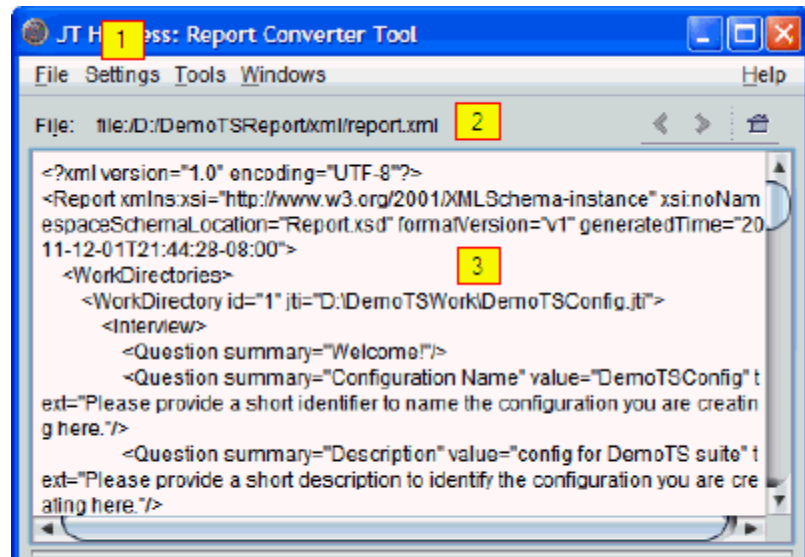
1. Report Converter Menu Bar

2. Navigation Bar

3. Report Viewer

Report Converter Menu Bar

The Report Converter menu bar contains a Settings menu. The menu bar also contains harness standard menus. See <u>Menus</u> for a description of the harness standard menus that are displayed on the menu bar.

The Settings menu contains the following menu items:

■ **New** - Launches the Report Converter wizard.

See <u>Report Converter Wizard</u> for a detailed description.

■ **Open** - Launches a file chooser for users to navigate to and open existing reports in the report viewer.

Navigation Bar

The navigation bar contains a report field and navigation buttons. The report field identifies the location and name of the report file displayed in the report viewer. When a report contains hyperlinks to other reports, users can use the links in the report and the navigation buttons to move from page to page.

Report Viewer

The harness displays the report in the viewer. The viewer is a noneditable text area.

Report Converter Wizard

The harness displays the Report Converter wizard when you launch the Report Converter Tool or when you choose Settings > New from the Report Converter Tool menu bar. The Report Converter wizard consists of two panels, an XML Input Files panel and a Report Options panel. See Merging and Converting XML Test Reports for instructions on the use of the Report Converter wizard.

XML Input Files Panel

Use the XML Input Files panel to specify the harness generated XML files that are to be merged and the directory where the harness is to write the final report.
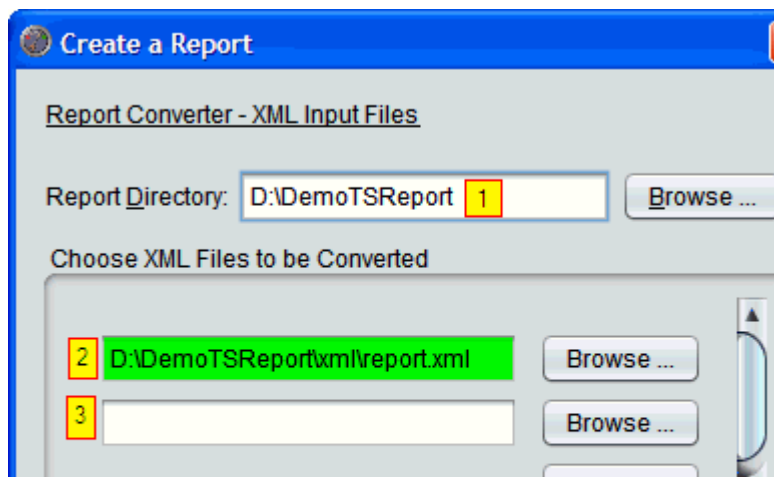
1. **Report Directory field and Browse button** - Set the directory where the report is saved

2. **Source file fields and Browse button**s - Specify the source XML report files

3. **Add More button** - Creates additional source file fields

4. **Next button** - Opens the Report Options panel

Report Options Panel

Use the Report Options panel to specify how conflicts between merged reports are handled, choose the report format, and initiate creating the merged report.
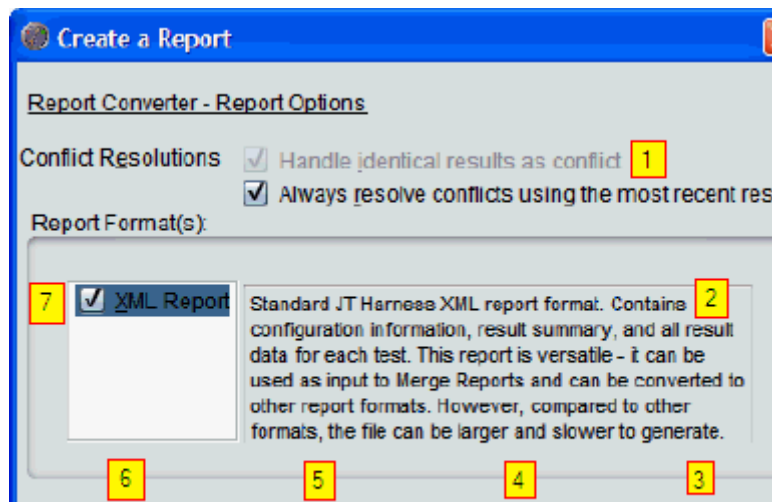
1. **Conflict Resolution options** - Determine how conflicts are handled

2. **Report Format Description** - Describes the selected report format

3. **Help button** - Launches a help viewer containing detailed information about the panel

4. **Cancel button** - Closes the Report Converter wizard without taking any action

5. **Create Report button** - Generates the report

6. **Back button** - Returns to the XML Input Files panel

7. **Report Format(s)** - List of available output types

Test Manager Tool

Use the Test Manager tool to load, edit, or create a configuration, to run and monitor tests, to browse test information, and to troubleshoot a test run.

You can launch the Test Manager at any time by choosing Tools > Test Manager or by choosing File > Open Quick Start from the menu bar.

The harness also automatically opens a new Test Manager when you perform one of the following actions:

- Open an existing work directory
- Create a work directory
- Open a test suite

Depending on the layout style that you use, the desktop menus (File, Tools, and Windows) might also be displayed in the menu bar. See Menus for a description of the desktop menus. See Appearance Preferences for a description of how to change the layout style.
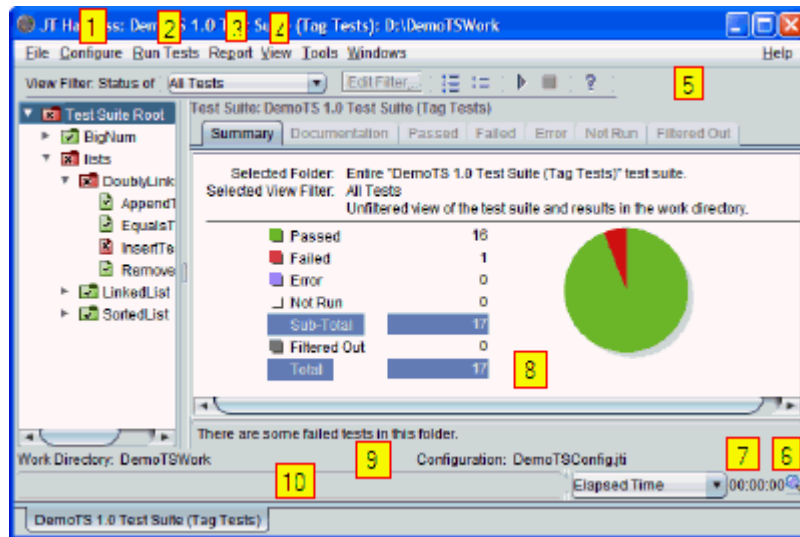
1. Configure Menu

2. Run Tests Menu

3. Report Menu

4. View Menu

5. Tool Bar

6. <u>Progress Monitor</u>

7. <u>Progress Indicator</u>

8. <u>Information Area</u>

9. <u>Work Directory Status Line</u>

10. <u>Test Manager Status Line</u>

11. <u>Test Tree</u>

The Test Manager also provides access to the following items:

- <u>Configuration Editor</u>
- <u>Template Editor</u>
- <u>Log Viewer</u>

Tasks Performed from the Test Manager

The following table describes the tasks that you can perform from the Test Manager.

TABLE 11 Test Manager Tasks

Task	Description
Load, edit, or create a configuration	Provide configuration information required to run your test suite. See <u>Configuring a Test Run</u> for a description of providing a configuration for running tests.
Run tests	Start test runs by choosing the Run Tests > Start menu item or click the ▶ button on the tool bar. See <u>Running Tests</u> for a detailed description of how to run tests.
Monitor test runs	Use the test tree with the folder and test views to monitor the status of the test run. For more information about monitoring test runs, see <u>Monitoring a Test Run</u>.
Browse test information	Use the test tree with the folder and test views to browse information about overall test status as well as what occurred during a test run. See <u>Browsing Test Information</u> for details about browsing test run information.
Create and view test reports	Use the Report menu to create and display reports of test run information. See <u>Creating and Displaying Reports</u> for details about generating and viewing test run information.
Troubleshoot a test run	Use the test tree with the folder and test views to troubleshoot a test run. See <u>Troubleshooting a Test Run</u> for help with troubleshooting test run problems.

Configure Menu

Use the Configure menu to load, create, modify, and view configuration data used for a test run. The following table describes the default menu items in the Configure menu. Because this menu can be customized you might see additional options defined in the current test suite.

The template menu options listed in the following table are only available when template usage is enabled.

TABLE 12   Configure Menu

Menu Item	Description
Edit Configuration	Opens the Configuration Editor and loads the current configuration. The harness uses the Configuration Editor to change configuration values required to run a test suite.
	See **Editing a Configuration** for detailed information.
Edit Template	Opens the Template Editor and loads the template on which the current configuration is based. If the test suite disables the template feature, this and all other template menu items are not displayed. This menu item is also disabled when the test suite only enables users to load but not edit templates. Use the Template Editor to change configuration values in the template and to propagate the changes to the current configuration.
	See **Editing a Template** for detailed information.
Edit Quick Set	Opens an additional menu containing Tests to Run, Exclude List (optional), Keywords (optional), Environment (optional), Prior Status, Concurrency, and Timeout Factor menu items that you can use to change specific configuration values.
	See **Editing Quick Set Values** for detailed information.
New Configuration	Opens the Configuration Editor and loads an empty configuration. The harness uses the Configuration Editor to create configuration data containing the test environment and standard values required to run a test suite.
	See **Creating a Configuration** for detailed information.
Load Configuration	Opens a Load Configuration File dialog box that you use to load an existing configuration into the Test Manager.
	The harness does not open the Configuration Editor when you load an existing, complete configuration interview.
	See **Loading a Configuration** for detailed information.
Load Recent Configuration	Displays a list of recently opened configuration. You can choose a configuration from the list to use as the current configuration.
New Template	Opens the Template Editor and loads an empty template. The harness uses the Template Editor to create a template containing known test environment and standard values required to run a test suite.
	See **Creating a Template** for detailed information about templates.

TABLE 12   Configure Menu

Load Template	Opens a Load Template dialog box that you use to load an existing template into the Test Manager. See <u>Loading a Template</u> for detailed information.
Load Recent Template	Displays a list of templates that have been opened. You can choose a template from the list to use as the template for creating a configuration.
Update Template	Enabled only when template propagation is enabled and the template on which current configuration is based has changed. Displays a dialog box that enables you to import changes from the template into your configuration.

Run Tests Menu

The Run Tests menu is used to start, stop, and monitor a test run. This menu can be customized to include additional actions, so you may see additional options. A custom menu is developed as part of a test suite, so it is visible only when the test suite is in use.

The following table describes the default items in the Run Tests menu.

TABLE 13   Run Tests Menu

Menu Item	Description
Start	The harness enables the Start menu item when it is not running tests. Choose the Start menu item to start a test run. Only one test run at a time can be active in the Test Manager. See <u>Starting a Test Run</u> for detailed information about starting a test run.
Stop	The harness enables the Stop menu item when it is running tests. Choose the Stop menu item to end a test run after the current test is completed. See <u>Stopping a Test Run</u> for detailed information about stopping a test run.
Monitor Progress	When the harness is running tests, you can use the Progress Monitor to display the progress of the test run and current resource information about the test system. Choose the Monitor Progress menu item to open the Progress Monitor. See <u>Using the Progress Monitor</u> for detailed information about using the Progress Monitor.

Report Menu

The Report menu contains menu items that create and view reports about test run information. The following table describes the items in the Report menu.

TABLE 14   Report Menu

Menu Item	Description
Create Report	Opens the Create a New Report dialog box for you to generate reports of test results.
	See **Creating Reports** for detailed information.
Open Report	Opens the Report dialog box used to specify a report.
	See **Displaying Reports** for detailed information.
Report History	Displays a list of available reports. Choose a report from the list to open a new instance of it in the Report browser.
	See **Displaying Reports** for detailed information about the Report browser.

View Menu

The View menu contains menu items that display information about a test run. The following table describes the items in the View menu.

TABLE 15   View Menu

Menu Item	Description
Configuration > Show Checklist (Optional)	Enabled by the harness when your test suite supports the configuration checklist. Displays a checklist of tasks to be performed before running tests. See **Displaying the Configuration Checklist** for detailed information.
Configuration > Show Exclude List (Optional)	Opens an Exclude List dialog box that contains the exclude list used to run the test suite. You can use the Exclude List dialog box to review but not edit the contents of the exclude list. If you use multiple exclude lists, the Exclude List dialog box displays the merged set of exclude lists.
	You must use the configuration editor window to add or remove exclude lists. See **Using Exclude Lists** for detailed information.
	See **Viewing Exclude List Contents** for detailed information about viewing the contents of the exclude list.

TABLE 15   View Menu

Configuration **>** Show Test Environment	Opens a Test Environment browser that displays the configuration values used when running the test suite. You can browse but not change values in the Test Environment browser. You must use the configuration editor to change the values. See <u>Editing a Configuration</u> for detailed information. See <u>Viewing Configuration Values</u> for detailed information about viewing the contents of the test environment.
Configuration **>** Show Question Log	Displays a log of the current configuration interview questions and answers. See <u>Displaying the Question Log</u> for detailed information.
Filters	Displays the available view filters, the active filter, and enables you to modify the custom view filter. See <u>Using View Filters</u> for detailed description.
Properties	Click the Properties menu item to display the Test Manager Properties dialog box containing the current settings of the Test Manager. See <u>Viewing Test Manager Properties</u> for detailed information about using the Test Manager Properties dialog box.
Logs	Is enabled if the harness detects any log output, otherwise it is disabled. Click the Logs menu item to display logged output generated by the harness for the test suite.
Test Suite Errors	Enables the Test Suite Errors menu item when it detects that the test suite (not the tests in the test suite) contains errors. Click the Test Suite Errors menu item to display a dialog box identifying the errors detected in the test suite. See <u>Viewing Test Suite Errors</u> for detailed information.
Toolbars	Is enabled if more than one toolbar is available to the user, otherwise it is disabled.

Tool Bar

The tool bar contains the following fields and buttons used to perform routine tasks that are also available as menu items from the menu bar:

- **View Filter: Status of** - Filters the test status (folder colors and counters) displayed in the Test Manager.

View filters do not determine the tests that are run. You must use the configuration editor to change the tests that are run. See <u>Using View Filters</u> for detailed information about each filter.

- **Edit Filter** - Enabled only when Custom is selected in the view filter list.

See [Using View Filters](#) for detailed information about creating a custom view filter.

- ▤ - Opens the Configuration Editor in Question Mode.

See [Question Mode](#) for a detailed description of this mode.

- ▤ - Opens the Configuration Editor in Quick Set Mode.

See [Quick Set Mode](#) for a detailed description of the mode.

- ▸ - Starts a test run.

See [Starting a Test Run](#) for a detailed description.

- ▪ - Stops a test run.

See [Stopping a Test Run](#) for a detailed description.

- ❓ - Displays User Guide information about the Test Manager.

Progress Monitor

The Progress Monitor is a dialog box that only displays information about the current configuration when the harness is running tests. The information displayed is equivalent to the Last Test Run view filter set in the Test Manager. Changing view filters in the Test Manager does not change the information displayed by the Progress Monitor.

Choose Run Tests > Monitor Progress from the Test Manager menu bar or click the 🔍 icon at the bottom of the Test Manager to open the Progress Monitor.
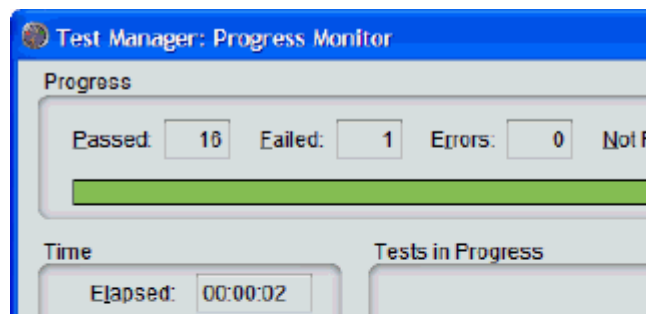
The following areas in the Progress Monitor display information about the test run:

- [Progress](#)
- [Tests in Progress](#)
- [Memory](#)

- [Time](#)

Progress

The following table describes the information displayed in the Progress area.

TABLE 16   Progress Area Contents

Name	Description
Passed	Displays the number of tests in the test suite that were run and had passing results.
Failed	Displays the number of tests in the test suite that were run and had failing results.
Errors	Displays the number of tests in the test suite that could not be run.
Not Run	Displays the number of tests in the test suite have not yet been run.
Test Results	A colored progress bar representing the results of the tests in the test suite.
	As the harness runs tests, the test results are displayed as colored segments in the progress bar. The colors used in the progress bar represent the current status of the test results.
	The colors below are the harness default settings. The harness supports using colors other than the default settings. See [Specifying Status Colors](#) in the *JavaTest Harness User's Guide: Command-Line Interface*.
	The progress bar is the same as that displayed in the test progress display at the bottom of the Test Manager.

The following table describes the colors used in the progress bar. The colors are displayed from left to right in the order in which they are presented in the table.

TABLE 17   Order of Colors Displayed in the Progress Bar

Color	Status	Description
Green	Passed	Tests in the test run having passing results when they were executed.
Red	Failed	Tests in the test run having failed results when they were executed.
Blue	Error	Tests in the test run that the harness could not execute.
		Errors usually occur because the test environment is not properly configured.
White	Not yet run	Tests in the test run that the harness has not yet executed. Tests excluded from the test run are not included.

You can display the progress bar in the Test Manager by clicking the ▼button and choosing Run Progress Meter from the selectable list. See <u>Using the Progress Meter</u> for detailed description.

Tests in Progress

The Tests in Progress text box displays either the names of the tests that the harness is currently running or the set of tests distributed for execution. It is empty when the harness is not running tests. Concurrency settings and agents in use determine the number of items displayed in the text box. See your test suite documentation for additional information.

Clicking on this list displays the appropriate test view in the Test Manager.

Memory

The Memory area contains two text fields and a bar graph. The following table describes the contents of the Memory area in the Progress Monitor.

TABLE 18   Memory Area Contents

Name	Description
Used:	The memory used to run the test.
Total:	The total memory available for use by the virtual machine.

Time

The Time area contains two fields that are continuously updated throughout a test run. The following table describes the contents of the Time area in the Progress Monitor.

TABLE 19   Time Area Contents

Name	Description
Elapsed:	The time elapsed since the test run was started.
Remaining:	The estimated time required to run the remaining tests.

Tests that timeout or have execution times significantly longer than the other tests being run can cause the harness to display inaccurate times.

Progress Indicator

The progress indicator located at the bottom of the Information Area displays the elapsed time of the previous test run when tests are not running. When tests are running, it automatically changes to the progress bar of the current test run. At the completion of the test run, the indicator changes to display the elapsed time.

You can use the ▼button at any time to select and display either the elapsed time or the progress bar.

Elapsed Time

When a test run starts, the elapsed time display resets to 00.00.00. At the end of the test run, the total elapsed time for the test run is displayed.

Progress Bar

The harness displays a progress bar and a percentage complete value. As the harness completes a test, it updates the progress bar and the percentage complete value.

Information Area

The Test Manager uses the information area to display information about the item selected in the test tree. The Test Manager uses two views, folder view and test view, to display the information.

Folder View

When you click a folder icon in the test tree, the Test Manager displays a Summary tab, a Documentation tab, five status tabs, and a status field containing information from the work directory about a folder and its descendants.
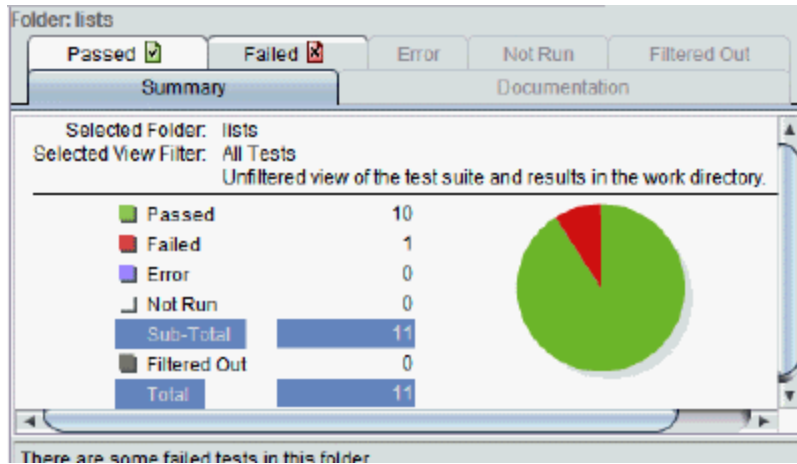
The folder view contains filtered summary and status information about the tests in a test folder. The Test Manager also displays a status message at the bottom of the area about the selected view. The messages can indicate that tests in the folder are loading or they can provide detailed status information about a selected test.

During a test run, you can use the folder view to monitor the status of a folder and its tests. You can also use the folder view during troubleshooting to quickly locate and open individual tests that had errors or failed during the test run. When a status pane is empty, the Test Manager disables its tab. This information is displayed in the Summary tabbed pane as values and as a pie chart. The folder view and test tree use the same view filter when displaying information.

See Displaying Folder Information for detailed information about this view.

Test View

When you click a test icon in the test tree or double click its name in the Folder view, the Test Manager displays tabbed panes that contain detailed information about the selected test.
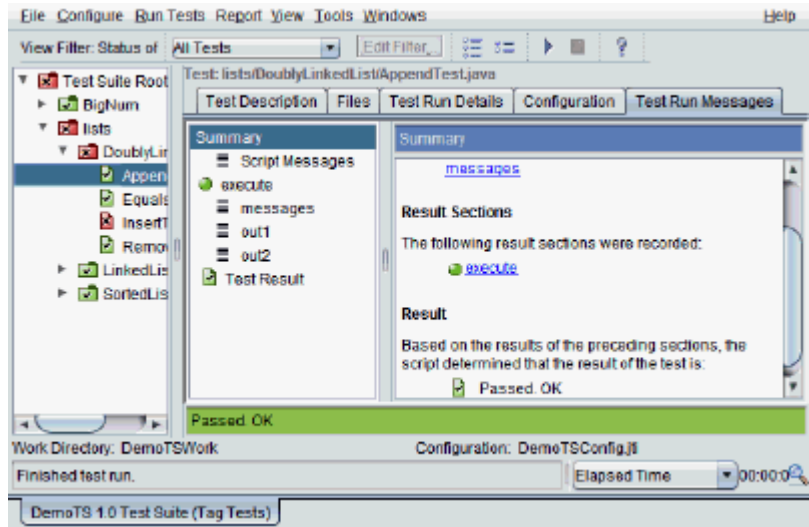
See <u>Displaying Test Information</u> for detailed information about this view.

Test Tree

The test tree is a multifunction panel that performs the following functions:

- Displays the current status and state of tests in the test suite
- Enables users to choose any combination of tests and folders in the test suite for a test run
- Enables users to refresh or to clear specific test results

The test tree supports keyboard navigation. See <u>Keyboard access</u> for a description of how the keyboard can be used to navigate the test tree pane.

Displaying Test Status and State

The test tree uses folder icons, test icons, and two independent types of filtering (run filtering and view filtering) to simultaneously display the progress of a test run and the current test results in the work directory.

The test tree indicates the progress of a test run by displaying an arrow to the left of each folder and test icon as it is loaded and run. See <u>Monitoring With a Test Tree</u> for detailed information about using the test tree to monitor a test run.

Based on the Test Manager view filter, the test tree updates the folder and test result icons to indicate the current test results in the work directory.

Example:

When you choose the Current Configuration view filter, the test tree only displays the results for the tests specified in your current configuration. If you choose the All Tests view filter, the test tree immediately redisplays all test results from the work directory regardless of the settings in your current configuration. To display filtered summary information about the test results in the work directory, click the test suite icon in the test tree.

See View Filters for a detailed description of how to specify which test results from the work directory are displayed in the test tree.

When you click a folder icon in the test tree, the harness displays its folder view in the Test Manager information area. The Test Manager uses the view filter to filter results displayed in the folder view. See Displaying Folder Information for additional information.

When you click a test icon in the test tree, the harness displays its test view in the Test Manager information area. Unlike the folder view, the Test Manager does not filter results in the test view. See Test View for additional information.

Run, Refresh, or Clear Results

You can select any combination of tests and test folders and right click in the test tree to open a pop-up menu for performing the following actions:

- Execute highlighted folders and tests by performing a quick pick execution.
- Refresh the list of folders and tests without restarting the harness by performing an on-demand refresh scan for new folders, new tests and updated test descriptions.
- Clear Results by performing an on-demand clearing of the contents of the highlighted folder, test, or entire work directory.

If a single test or folder is highlighted in the test tree, the Test Manager displays the appropriate Test View or Folder View. However, when multiple tests and folders are highlighted in the test tree, the harness displays a Multiple Tree Nodes Selected pane that lists the highlighted tests and folders. Highlight multiple tests and folders in the test tree by using standard keystrokes (such as Shift and Control) and clicking the test and folder icons in the test tree.

The selection list in the Multiple Tree Nodes Selected pane displays the full path of each node highlighted in the test tree. Click a test or folder in the test tree a second time to remove the highlighting and to remove it from the selection list. The GUI displays the updated Multiple Tree Nodes Selected pane.

Folders and tests do not have to be highlighted in the test tree for you to use the pop-up menu. See Test Tree Pop-up Menu for detailed information about using the pop-up menu.

Folder Icons

The test tree uses colored icons to indicate both run status and result status. The colors of the icons shown below are the default settings. The harness enables you to use colors other than the default settings. See Specifying Status Colors in the *Command-Line Interface User's Guide*.

Folder Run Status

When activity occurs in a folder, such as loading or running tests, the Test Manager displays an arrow to the left of the folder icon. The standard values of the current configuration, not the view filter, determine the run status displayed by the test tree.

Folder Result Status

The folder icon indicates the current test results in the work directory and does not change until its tests are completed. After the harness completes running the tests in a folder, it displays the appropriate result status icon.

The folder icon displayed in the test tree is determined by the result of all its tests (see Test Icons) and the current view filter (see View Filters). The folder icons displayed in the test tree indicate the highest priority result of any test hierarchically beneath it. The following table describes the folder icons in order of priority.

Note –

Test Icons

The test tree uses colored icons to indicate both run status and result status. The colors of the icons shown below are the default settings. The harness enables you to use colors other than the default settings. See Specifying Status Colors in the *Command-Line Interface User's Guide*.

Test Run Status

The Test Manager displays an arrow to the left of the test icon when running a test. The standard values of the current configuration, not the view filter, determine the run status displayed by the test tree.

Test Result Status

The test icon indicates the last known test result and does not change until the test is completed. After the harness completes the test, it displays the appropriate result status icon.

The test result icons displayed in the test tree are determined by the results in the work directory and by the view filter set in the Test Manager. See <u>View Filters</u> for a description of how to specify the test results displayed in the test tree. The following table describes the test icons used to indicate the result of each test.

Note – Users can change the color of the test icons displayed in the following table. However, the symbols displayed in the icons do not change.

TABLE 20   Test Icons

Icon	Result	Description
🔵	Error	A blue test containing an exclamation symbol (**!**) indicates that the test is not filtered out and that harness could not execute it. These errors usually occur because the test environment is not properly configured.
❌	Failed	A red test containing a **x** symbol indicates that the test is not filtered out and had a Failed result the last time it was executed.
➖	Not Run	A white test containing a **-** symbol indicates that the test is not filtered out but has not yet been executed.
✅	Passed	A green test containing a ✔symbol indicates that the test is not filtered out and had a Passed result the last time it was executed.
⬜	Filtered Out	A gray test containing no symbols indicates that the test is currently not selected to be run.

Test Tree Pop-Up Menu

The test tree provides a pop-up menu. Displaying the pop-up menu for a folder or test is a platform-specific operation (such as right clicking the folder or test icon in the test tree). You can also press Shift-F10 to display the pop-up menu for the test with focus.
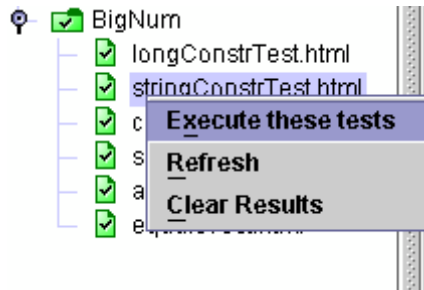
The pop-up menu contains the following menu items:

■ **Execute these tests** - See Quick Pick Test Execution for a detailed description.

■ **Refresh** - See Refresh Test Suite Contents for a detailed description.

■ **Clear Results** - See Clear Previous Test Results for a detailed description.

The harness does not allow you to perform operations using the pop-up menu when it is running tests. The harness displays an error dialog box if you attempt to perform an operation using the pop-up menu when tests are running.

See Keyboard access for a description of how you can use the keyboard to open and navigate the pop-up menu.

Quick Pick Test Execution

You can use the Execute these tests menu item to run the tests and folders highlighted in the test tree, subject to filters such as keywords.

The harness does not automatically perform a refresh operation before running the tests. If changes are made to a test suite, you must perform a refresh before running tests. See Refresh Test Suite Contents for a description of the refresh operation.

▼ Quick Pick Test Execution of Specific Folders or Tests

1. Select one or more folders and tests in the test tree.

- You must left click to select nodes. When more than one folder or test is highlighted in the test tree, the Test Manager displays a list of the selected tests and folders in the information area.

- Selecting a test executes only that test.

- Selecting a folder executes all tests currently known to the Test Manager in and below that folder.

2. Right click in the test tree pane to open the pop-up menu.

Opening the pop-up menu is a platform-specific operation (such as right clicking in the test tree pane). A modifier key or operation which doesn't utilize the clicking at all may be necessary depending on your platform details.

3. Choose Execute these tests from the test tree pop-up menu.

- If the Test Manager contains a completed configuration, the harness displays a message to confirm the execute operation.

- If the Test Manager does not contain a completed configuration, the harness displays a message asking you to complete the configuration or cancel the test run.

Note – The harness combines the <u>current configuration</u> and the nodes selected in the test tree to determine which tests will be executed. In the <u>Test Manager Preferences</u>, the option "Execution from tree popup includes configuration Tests to Run" how the harness uses the Tests to Run value in the configuration, as described below:

Box is unchecked (default). The harness ignores the tests specified in the <u>current configuration</u>'s Tests to Run question. It runs the tests selected in the test tree, subject to configuration Test Selection filters, such as keywords. This option is compatible with the behavior of test harnesses older than version 4.2.1.

Box is checked. The harness compares the tests you selected in the test tree and the tests specified in the configuration's Tests to Run question to find a set of common tests. The harness then applies the configuration's Test Selection filters (such as keywords) to the common test set, producing the set of eligible tests.

4. Tests will be executed. The above test selection process and exclusions from the criteria defined by the test suite or interview may eliminate all the eligible tests. If this happens you will see a message, and the test run is aborted.

Refresh Test Suite Contents

When developing tests, changes in a test suite are not automatically detected by the harness. The first time tests are run, the harness uses the test finder to read test descriptions. If the harness loads tests from an existing work directory, the test descriptions contained in those results are used by default.

The refresh operation enables test developers to load changes they might have made in a test suite without restarting the harness or reloading the test suite.

The harness does not require a work directory to perform a refresh of the test suite.

If you are viewing the test panel after refreshing a test or folder, update the test tree by choosing a different test or folder icon and repeating your test tree choice.

▼ Refresh Tests and Folders

1. Select one or more folders and tests in the test tree.

- You must left click to select a node. When more than one folder or test is highlighted in the test tree, the Test Manager displays a list of the selected tests and folders in the information area.

- Selecting a test chooses only that test.

- Selecting a folder chooses all tests currently known to the Test Manager in and below that folder.

2. Right click in the test tree to open the pop-up menu.

Opening the pop-up menu is a platform-specific operation (such as right clicking in the test tree).

3. Choose Refresh from the pop-up menu.

4. For files, the harness checks the time stamp of the file containing the test description.

5. If the time stamp has changed, the harness compares the test descriptions.

6. If the properties of the test descriptions are different, the harness removes the test result from the work directory and the Test Manager and loads a test containing the new test description into the Test Manager and displays it in the Not Run state.

7. For folders, the harness checks the time stamps of the files in a folder and scans for new folders and tests. This operation may take place on any folder including the root folder.

8. If a time stamp has changed, the harness compares the test descriptions.

9. If the properties of the test descriptions are different, the harness removes the test result from the work directory and the Test Manager and loads the test containing the new test description into the Test Manager and displays it in the Not Run state.

Clear Previous Test Results

You can use the Clear Results menu item to remove existing test results for any combination of tests and folders.

To clear test results, you must have an open work directory.

▼ Clear a Test Result

1. Select one or more tests in the test tree.

You must left click to select a node. When more than one test is highlighted in the test tree, the Test Manager displays a list of the selected tests and folders in the information area.

2. Right click in the test tree to open the pop-up menu.

Opening the pop-up menu is a platform-specific operation (such as right clicking in the test tree).

3. Choose Clear Results from the test tree pop-up menu.

The harness opens a confirmation dialog box that displays a list of the selected items and performs the following actions when the user clicks the Yes button:

- Removes the `.jtr` file from the work directory for that test.
- Refreshes the test description for that test.
- Displays the test in the Not Run state.

▼ Clear Test Results in Folders

1. Select one or more folders in the test tree.

You must left click to select a node. When more than one folder is highlighted in the test tree, the Test Manager displays a list of the selected tests and folders in the information area.

2. Right click in the test tree pane to open the pop-up menu.

Opening the pop-up menu is a platform-specific operation (such as right clicking in the test tree).

3. Choose Clear Results from the test tree pop-up menu.

The harness displays a confirmation dialog box that displays a list of selected items and performs the following actions when the user clicks the Yes button:

- Removes all `.jtr` files from the work directory for all tests in and below that folder.
- Deletes all other files in and below the folder in the work directory.
- Deletes all other directories corresponding to the folders in and below the folder in the work directory.
- Displays the folder and its tests as Not Run.

The harness does not display an error message if it is unable to delete a folder or file from the work directory.

View Filters

The harness provides a special view filtering facility that enables you to filter the status (the colors and counters) of the folders and tests displayed in the Test Manager. View filters function independently from the run filtering set in the configuration editor window.

See [Editing a Configuration](#) for detailed information about specifying the tests that are run.

Note – You also use view filters when generating reports. See [Creating Reports](#) for a description of using view filters when generating reports.

The harness provides four Test Manager view filters:

- [Last Test Run](#)
- [Current Configuration](#)
- [Current Template](#)
- [All Tests](#) (default)
- [Custom](#)

An additional view filter, such as a Certification filter, can be added by the test suite. Refer to your test suite documentation for detailed descriptions of any additional filters displayed in the list of view filters.

To use a view filter, either choose View > Filters from the menu bar or choose a view filter from the tool bar. See [Custom View Filter](#) for a description of how to create a custom view filter.

If you checked the Save Desktop State on Exit option in the Preferences dialog box, the current view filter information is saved when you exit.

Current view filter information is **not** saved if you:

- unchecked the Save Desktop State on Exit option in the Preferences dialog box
- provided test suite or work directory information on the command line when starting the harness
- used the -newDesktop on the command line when starting the harness

[See Shutdown Options.](#)

Last Test Run View Filter

When you select the Last Test Run view filter, the Test Manager displays the current totals in the Summary pane and status icons in the test tree for all folders and tests included in the last test run (even if you have exited the harness since the last test run) or the states from the test run in progress if the harness is running tests. The information displayed in the Test Manager is associated with the work directory.

The following examples provide descriptions of the use of the Last Test Run view filter:

- **Example 1 -** A test run had failed tests and you want to rerun only the failed tests. See [Specifying Prior Status](#) for a detailed description of setting the prior status value. The Test Tree and the Summary pane continue to show all tests in the last test run regardless of their status.

When you repeat the test run, the Test Manager clears the previous test results and only displays current status icons in the test tree and totals in the Summary pane for the tests and folders in the current test run.

- **Example 2 -** A test run had failed tests and you want to use an Exclude List in the next run that excludes the failed tests from the test run. See [Using Exclude Lists](#) for a detailed description of specifying an exclude list. The Test Tree and the Summary pane do not change when the exclude list is added to the configuration.

When you repeat the test run, the Test Manager clears the previous test results and only displays current status icons in the test tree and totals in the Summary pane for the tests and folders in the current test run.

Current Configuration View Filter

When you select the Current Configuration view filter, the Test Manager only displays the status of the folders and tests currently selected by the current configuration.

To select the Current Configuration view filter, choose View > Filter > Current Configuration from the menu bar or choose the Current Configuration view filter from the tool bar.

The following examples provide descriptions of the use of the Current Configuration view filter:

- **Example 1 -** You only want to run and view the results for tests in the `api` folder. See <u>Specifying Tests to Run</u> for detailed description of specifying the tests that are run.

Either choose View > Filter > Current Configuration from the menu bar or choose the Current Configuration view filter from the tool bar.

The Test Manager displays gray ▢folder and ▯test icons for all folders and tests except those under `api`. The Test Manager changes the Summary totals to indicate that these tests are filtered out.

The current result icons are only displayed in the test tree for those folder and test icons under `api` and the Summary pane is updated to display their totals. When you run the tests, the test tree and folder view only display the results for those folders and tests under `api`.

- **Example 2 -** You only want to run and view results for failed tests. See <u>Specifying Prior Status</u> for a detailed description of setting the prior status value.

Either choose View > Filters > Current Configuration from the menu bar or choose the Current Configuration view filter from the tool bar.

Immediately the Test Manager displays gray ▢folder and ▯test icons for all folders and tests except those with Failed test results. The Test Manager updates the Summary totals to indicate that these tests are filtered out and changes the Passed, Error, and Not Run totals to indicate 0.

The Test Manager displays those folders and tests in the test tree with Failed test results as red ❌folder and ❌test icons. The Test Manager updates the Summary information to indicate the number of tests that have Failed test results.

When you repeat the test run, as failed tests pass, the Test Manager updates their icons from red ❌folder and ❌test icons to gray ▢folder and ▯test icons (they no longer match the configuration criteria of the view filter). The Test Manager updates the totals in the Summary pane to match the results displayed in the test tree.

Current Template View Filter

When you select the Current Template view filter, the Test Manager only displays the status of the folders and tests selected by the template on which the current configuration is based.

Select the Current Template view filter by either choosing View > Filter > Current Template from the menu bar or choosing the Current Template view filter from the tool bar.

All Tests View Filter

When you select the All Tests view filter, the Test Manager immediately displays current totals in the Summary pane and status icons in the test tree for all folders and tests, regardless of the values set in the configuration. This is an unfiltered view of the complete work directory contents.

The harness only overwrites the previous results for tests when they are rerun. The All Tests filter displays the last run status for all tests in the work directory, including the tests that were excluded from the current test run.

If you are using the All Tests view filter and begin a test run, the harness displays an advisory dialog box. Using the All Test view filter will display the results of all tests in a test suite regardless of whether or not they are included in the test run.

You can choose to disable the dialog box by setting preferences, by choosing a different view filter, or by using the check box in the dialog to stop the dialog from being displayed in the future.

The following examples provide descriptions of the use of the All Tests view filter.

- **Example 1 -** A test run had failed tests and you want to rerun only the failed tests. See Specifying Prior Status for a detailed description of setting the prior status value. The Test Tree and the Summary pane do not change.

When you use the All Tests view filter, the Test Manager displays all totals in the Summary pane and status icons in the test tree regardless of the values set in the current configuration.

- **Example 2 -** A test run had failed tests and you want to use an Exclude List in the next run that excludes the failed tests from the test run. See Using Exclude Lists for a detailed description of specifying an exclude list. The Test Tree and the Summary pane do not change.

When you repeat the test run, the Test Manager displays current status icons in the test tree and totals in the Summary pane for every test in the work directory, including results for any tests excluded from the current test run.

Custom View Filter

If you want to monitor a specific set of test results in the Test Manager, you can create a custom view filter.

The custom view filter is unique to the test suite. When the harness opens the test suite in the Test Manager, it restores the custom filter, including any name that you assign it.

You can also use the custom filter to generate reports. See Creating Reports for a description of using the Custom filter when generating reports.
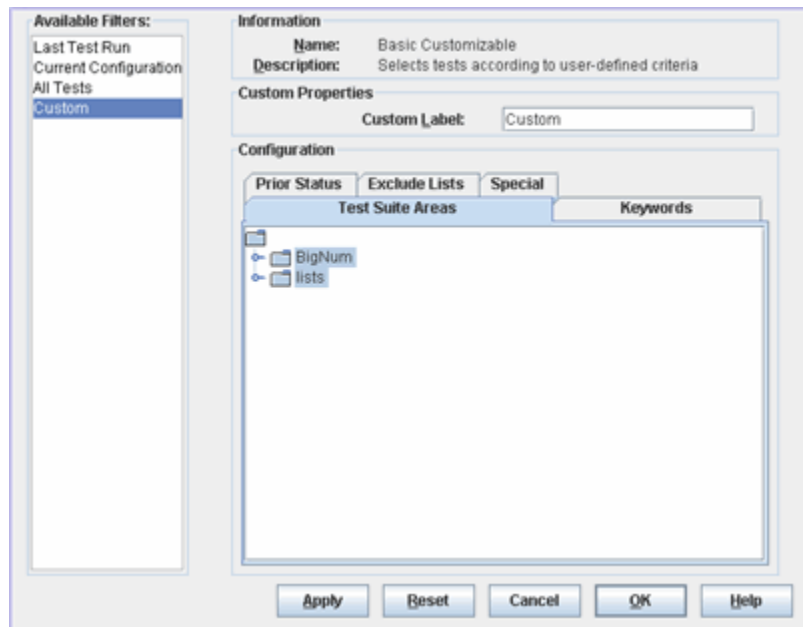
▼ Creating a Custom View Filter

1. Choose View > Filter > Configure Filters from the menu bar to open the Filter Editor. You can also choose Custom and then Edit Filter in the tool bar.

2. Choose Custom in the Available Filters panel. You can provide a name for the filter in the Custom Label field. The name is applied to the filter and restored each time the test suite is opened.

3. Use the tabbed panes to set the following view filter properties:

   - [Specify Tests to View](#)
   - [Use the Exclude List as a View Filter](#)
   - [Use Keywords as a View Filter](#)
   - [Use Prior Status as a View Filter](#)
   - [Use Special Settings as a View Filter](#)

4. Click one of the following buttons:
   - **Apply** - Saves but does not dismiss the dialog box. Updates the GUI if the filter is selected.
   - **Reset** - Discards all changes and restores the last saved Custom filter.
   - **Cancel** - Closes the dialog box without saving any changes.
   - **OK** - Saves the current changes, updates the Custom filter, and closes the dialog box.
   - **Help** - Displays online help for the Filter Editor.

Using Test Suite Areas, Keywords, Prior Status, Exclude Lists, and Special settings in the view filter does not stop the harness from running these tests. To filter the tests that are run, you must change the values in the configuration. See [Editing a Configuration](#) for a detailed description.

Specify Tests to View

Click the Test Suite Areas tab and use the tree to choose the results of test folders or individual tests that you want displayed in the Test Tree. The harness walks the test tree starting with the sub-branches and tests you specify and displays the results of all tests that it finds.

Use Keywords as a View Filter

If your test suite provides keywords, you can use the Keywords pane to restrict the set of test results displayed in the test tree and in the Summary pane.

To specify the keywords:

1. Click Match.

The harness enables the Expression button.

2. Click Expression to display a list of expressions that can be constructed.

3. From the list, choose the type of expression that you are building.

4. In the text field, enter the keywords and operators used in the expression.

The following table provides descriptions and examples of keyword expressions that can be constructed.

TABLE 21 Keyword Expressions

Expression	Description
Any Of	Displays all tests in the test suite having *any* of the keywords entered in the text field.
	Example:
	A test suite uses the keyword `interactive` to identify tests that require human interaction, and `color` to identify tests that require a color display.
	To display only the results for tests containing the `interactive` keyword, choose `Any Of` and then use the `interactive` keyword.
All Of	Displays results for all tests in the test suite having *all* of the keywords entered in the text field.
	Example:
	To display results for only the tests containing both the `interactive` and `color` keywords, choose `All Of` and then use the `interactive` and `color` keywords.
Expression	Displays results for all tests in the test suite having the *expression* entered in the text field.
	Construct a Boolean expression in the text field. Keywords stand as Boolean predicates that are true if, and only if, the keyword is present in the test being considered. A test is accepted if the overall value of the expression is true. All other tests are rejected by the restriction.
	Example:
	A test suite uses the keyword `interactive` to identify tests that require human interaction, and `color` to identify tests that require a color display.
	To display results for only the tests with the `color` keyword that do not also contain the `interactive` keyword, choose `Expression` and then use the `color` keyword, the `!` operator, and the `interactive` keyword.

Use Prior Status as a View Filter

Click the Prior Status tab and choose the test results from the previous test run that you want displayed in the Test Tree.

TABLE 22   Using Prior Status as a Filter

Prior Status Filter	Action
Passed	Displays tests with passed results the last time the test was executed.
Failed	Displays tests with failed results the last time the test was executed.
Error	Displays tests that the harness could not execute the last time it was included in a test run.
Not Run	Displays tests without results in the current work directory.

Prior status is evaluated on a test-by-test basis using information stored in the result files (`.jtr`) that are written in the work directory. Unless overridden by a test suite, a result file is written in the work directory for every test that is executed. When you change work directories between test runs, the result files in the previous work directory are no longer used and, if the new work directory is empty, the harness behaves as though the test suite was not run.

Use the Exclude List as a View Filter

To use the exclude list specified in the configuration interview as a filter, click the Exclude Lists tab and the Use settings in interview check box. The Exclude Lists pane displays the name of the exclude list file used by the current configuration.

Any test in the exclude list is filtered out and displayed as a ▯icon in the test tree.

Use Special Settings as a Filter

If the test suite architect provides a default filter for the test suite, click the Special tab and the Enable test suite filter check box to select this filter.

You must select this setting to correctly simulate the Current Configuration settings.

Use a Custom View Filter

To use a custom filter, you must choose it from the list of view filters below the test tree or from the View > Filters menu. The Test Manager updates the status of the folders and tests in the test tree that match the filter settings of the custom filter.

Work Directory Status Line

The work directory status line displays the current work directory or file name (for the full path, choose View > Properties), the configuration associated with the work directory, and the name of the template (if any) used to create the configuration.

Work Directory Status Line

Test Manager Status Line

The status message line is a resizable text area at the bottom of the Test Manager that displays information about current Test Manager activity, such as the state of a test run and the name of the test being run.
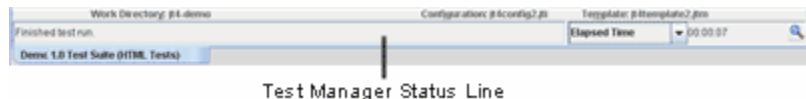
Test Manager Status Line

Note – If more than one test is running at a time, the Test Manager only displays the name of the last test started in the status message.

Configuration Editor

The Configuration Editor provides you with the following edit modes:

- Question Mode
- Quick Set Mode

In Question Mode, the Configuration Editor displays all of the questions and values in a configuration interview.
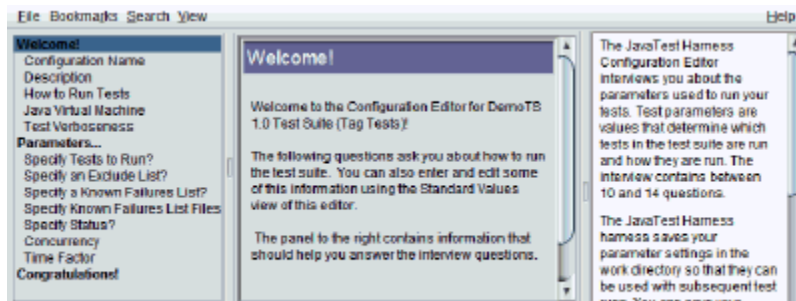
You can choosing Configure > Edit Configuration from the menu bar. You can also open the Configuration Editor in Question edit mode by clicking the ≡ button in the tool bar. After the Configuration Editor opens, you can use the View menu at any time to change edit modes without losing your changes.

See Question Mode for detailed information about this editing mode.

In Quick Set Mode, the Configuration Editor displays (in tab pane form) the runtime questions from a configuration interview. These are the values that specify which tests are run and how they are run. These values frequently change from test run to test run.
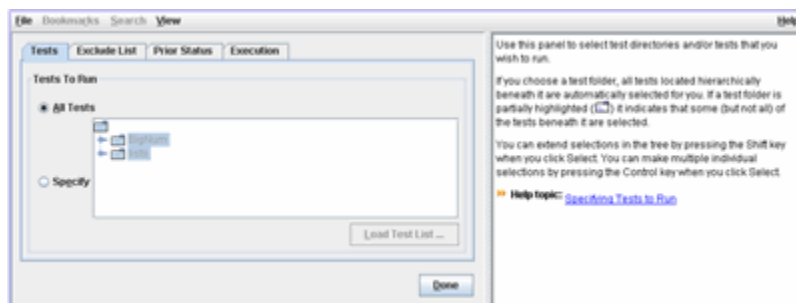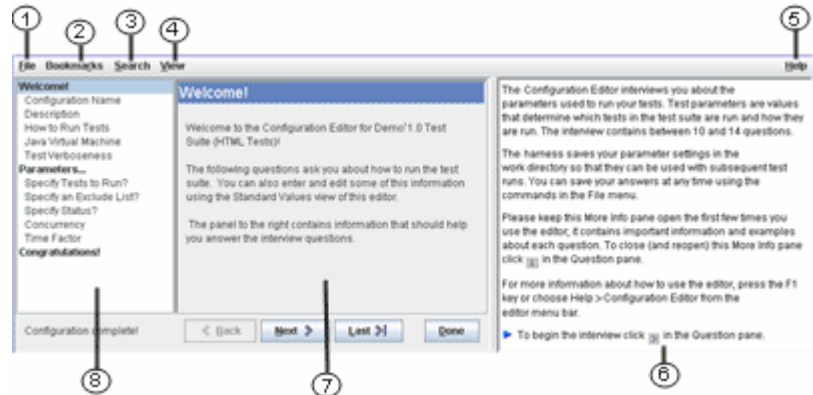
You can open the Configuration Editor in Quick Set edit mode by choosing Configure > Edit Quick Set from the menu bar. You can also open the Configuration Editor in Quick Set edit mode by clicking the ≡ button in the tool bar.

See Quick Set Mode for detailed information about this editing mode.

Question Mode

The Question Mode displays the complete configuration, allowing you to create a new configuration, change the values in a configuration, or search for character and value strings in a configuration.

1. File Menu

2. Bookmarks Menu

3. Search Menu

4. View Menu

5. Help Menu

6. More Info Pane (can be hidden)

7. Question Pane

8. Index Pane

Menus

In Question Mode, the Configuration Editor contains menus used to load, create, display, and change a configuration.

File Menu

The File menu contains items to open, save, and restore configuration files. The following table describes the items in the Configuration Editor File menu.

TABLE 23   Configuration Editor File Menu

Menu Item	Description
Save	Saves the current configuration. Choose File > Save at any time to save your answers and position in the configuration file. If the configuration is new, the editor opens the file chooser for you to use in naming and saving the current configuration. If you do not provide the .jti extension when you name the file, the editor adds the extension when it saves the file.
Save As	Opens a dialog box that you can use to save a configuration with a new name. The Configuration Editor makes the saved configuration the current configuration. If you do not provide the .jti extension when you name the configuration file, the editor adds the extension when it saves the file.
Revert	Discards any changes to the current configuration and restores the last saved version of configuration file.
New Configuration	Clears the current configuration and starts a new configuration. See Creating a Configuration> for a detailed description.
Load Configuration	Opens an existing configuration file and makes it the current configuration. See Loading a Configuration for a detailed description.
Load Recent Configuration	Displays a list of configuration files that have been opened in the Configuration Editor window. Choose a configuration file from the list to open it in the Configuration Editor window.
New Template	Provides the option to save or clear the current template before starting a new template. See Creating a Template for a detailed description.
Load Template	Opens an existing template. See Loading a Template for a detailed description.
Load Recent Template	Displays a list of templates that have been opened in the Template Editor. Choose a template from the list to open it in the Template Editor.
Close	Closes the Configuration Editor window.

Bookmarks Menu

The Bookmarks menu contains items to use bookmarks in the configuration. The following table describes the items in the Bookmarks menu.

TABLE 24 Configuration Editor Bookmarks Menu

Menu Item	Description
Enable Bookmarks	Enables and disables bookmarking in the configuration.
Show Only Bookmarked Questions	Display only the marked questions or all questions.
Mark Current Question	Clears the bookmark from a selected question in the Configuration Editor window.
Unmark Current Question	Bookmarks the selected question in the Configuration Editor window. Enabled only if the selected question is not bookmarked.
Clear Answer For Current Question	Clears the answer for a selected question in the Configuration Editor window. Enabled only if the selected question is bookmarked.
Open Group	Expands a selected set of questions in the Configuration Editor window.
Clear Answers to Bookmarked Questions	Clear the values in all marked questions.
Remove Bookmarks	Remove all bookmarks from the configuration.

Using bookmarked questions enables the user to display only those configuration questions that must be answered. See <u>Using Bookmarks in Configurations</u> for a description of how the Bookmarks menu can be used to specify the questions that the Configuration Editor window displays.

Search Menu

Use the Search menu items to find the occurrence in a configuration of a specific character or value string. When troubleshooting a test run, you can use the Search menu to quickly locate an answer that needs to be changed. The following table describes the items in the Search menu.

TABLE 25   Configuration Editor Search Menu

Menu Item	Description
Find	Opens a dialog box used to search the configuration for a specific character or value string.
Find Next	Searches the configuration for the next occurrence of a specific character or value string.

See [Searching a Configuration](#) for a detailed description of how to search for character and value strings in a configuration.

View Menu

Use the View menu to display the Configuration Editor window in Question Mode or in Quick Set Mode, to hide or display the More Info pane, and to display question tag field at the bottom of each question panel. The following table describes the items in the View menu.

TABLE 26   Contents of the View Menu

Menu Item	Description
Question Mode	Displays the Configuration Editor window in Question Mode.
Quick Set Mode	Displays the Configuration Editor window in Quick Set Mode.
More Info	Displays and hides the More Info pane in the Configuration Editor window.
Question Tag	Displays and hides the Question Tag field in the Configuration Editor Question Pane.
Refresh	Updates the values and questions displayed in the Configuration Editor window.

Help Menu

Use the Help menu to display the online help for the Configuration Editor window and both editor modes. The following table describes the items in the Help menu.

TABLE 27 Configuration Editor Help Menu

Menu Item	Description
Configuration Editor	Displays online help for the Configuration Editor window.
Template Editor	Displays online help for the Template Editor window.
Question Mode	Displays online help for the Question Mode.
Quick Set Mode	Displays online help for the Quick Set Mode.

Index Pane

When completing a configuration, the index pane lists the titles of the questions you have answered, are currently answering, or that the editor determines might need to be answered. The current question is highlighted.

In completed configurations, the questions that are displayed in the index can be controlled by setting bookmarks in the configuration. See Bookmarks Menu.

Note – The title is also displayed at the top of the question pane when you are answering a question.

Click on any question in the index list to make it the current question. Clicking on a question does not cause the list to change. If you change an answer that alters the configuration options, the Configuration Editor window updates the questions in the list to reflect the change in options. If a previously answered question is no longer displayed in the index list, the Configuration Editor saves its answer until you either save the configuration or change its value.

You can also use the buttons at the bottom of the Question pane to navigate through the configuration file. See Question Pane for a description of the navigation buttons.

Question Pane

The Configuration Editor window displays configuration questions in the main text area of the editor. You answer the questions using controls such as text boxes, radio buttons, or combo boxes located beneath the question. After you answer each question, click Next > at the bottom of the panel to proceed to the next question.

The buttons at the bottom of the Question pane control the following functions:

- **Back** - Returns to the previous question.
- **Last** - Advances as far as possible through the configuration.
- **Next** - Proceeds to the next question.
- **Done** - Saves your answers as a configuration and closes the Configuration Editor window.

See [Creating a Configuration](#) for information about using the Question pane to create a configuration.

See [Editing a Configuration](#) for information about using the Question pane to edit the current configuration.

More Info Pane

To open and close the More Info pane, choose View > More Info from the menu bar.

The More Info pane provides additional information about each question, including the following:

- Background information about the question
- Information about choosing an answer
- Examples of answers

See [Keyboard access](#) for a description of how the keyboard can be used to navigate the More Info pane.

Quick Set Mode

Quick Set Mode uses tabbed panes to display the values of a configuration that the harness uses to determine which tests are run and how they are run. Because they might change from test run to test run, the Quick Set Mode enables the user to quickly locate and edit these values.
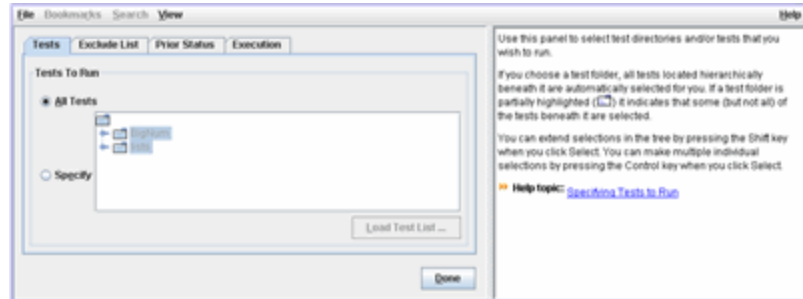
Quick Set Mode displays only the values that apply to your test suite.Tabs containing values that do not apply to your test suite are not displayed in Quick Set Mode. Depending on your test suite, Quick Set Mode can display four, five, or six tabbed panes and a menu bar.

- [Tests](#)
- [Exclude List](#)
- [Keywords](#) (Optional)
- [Prior Status](#)
- [Execution](#)

To use Quick Set Mode, a configuration must be loaded in the Configuration Editor. See [Editing Quick Set Values](#) for a description of how to change the Quick Set values of a configuration.

Template Editor

You can open the Template Editor by choosing Configure > New Template, Configure > Load Template, or Configure > Load Recent Template in the Test Manager menu bar. Like the Configuration Editor, the Template Editor provides users with the following two modes of displaying and working with the questions and values in a template.

- Question Mode
- Quick Set Mode

The harness always opens the Template Editor in Question Mode.

In Question Mode, the Template Editor displays all of the questions in the interview and enables the Bookmarks feature. You can use the Bookmarks feature to only display those questions in a template that a user must answer.

You can use the View menu at any time to change from one mode to the next. Because there is only one template file loaded in the Template Editor at a time, when you change viewing modes during an editing session, any changes that you have made in one viewing mode are automatically reflected in the other viewing mode.

See Question Mode for detailed information about using this mode to create or edit templates.

In Quick Set Mode, the Template Editor filters out the test suite specific questions and displays the remaining questions in logical groupings of the values. The Quick Set Mode displays them in tab form for easy user navigation and access. These values are the same as those values displayed in Question Mode. Only the form of their presentation to the user is different.
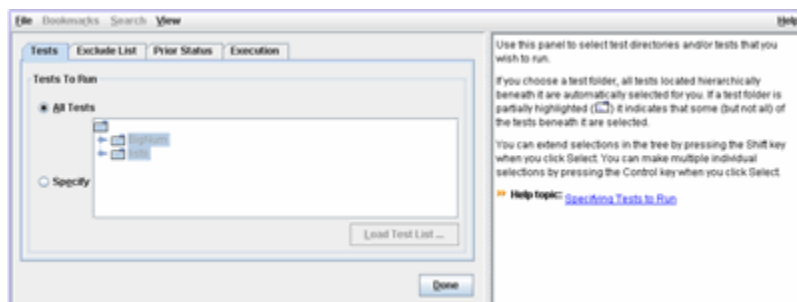
In Quick Set Mode, the Bookmarks and Search functions are not enabled.

See Quick Set Mode for additional information about using this mode to edit templates.

Question Mode

The Question Mode displays all questions in a test suite interview, allowing you to create a new template, edit an existing template, or search a template for specific characters or text.

1. [File Menu](#)

2. [Bookmarks Menu](#)

3. [Search Menu](#)

4. [View Menu](#)

5. [Help Menu](#)

6. [More Info Pane](#) (can be hidden)

7. [Question Pane](#)

8. [Index Pane](#)

Menus

In Question Mode, the Template Editor contains menus used to load, view, search, create, and edit a template.

File Menu

The File menu contains items to open, save, and restore template files. The following table describes the items in the File menu.

TABLE 28   Template Editor File Menu

Menu Item	Description
Save	Saves the current template.
	Choose File > Save at any time to save your answers and position in the template. If the template is new, the editor opens the file chooser for you to name and save the current values. If you do not provide the .jtm extension when you name the file, the editor adds the extension when it saves the template.
Save As	Opens a dialog box that you can use to save a template with a new name or change the location of the template. If you do not provide the .jtm extension when you name the template, the editor adds the extension when it saves the file.
Revert	Discards any changes to the current template and restores the last saved version of the template.
New Configuration	Clears the current configuration and starts a new configuration. See Creating a Configuration for a detailed description.
Load Configuration	Opens an existing configuration file and makes it the current configuration. See Loading a Configuration for a detailed description.
Load Recent Configuration	Displays a list of configuration files that have been opened in the Configuration Editor window. Choose a configuration file from the list to open it in the Configuration Editor window.
New Template	Provides the option to save or clear the current template before starting a new template. See Creating a Template for a detailed description.
Load Template	Opens an existing template. See Loading a Template for a detailed description.
Load Recent Template	Displays a list of templates that have been opened in the Template Editor. Choose a template from the list to open it in the Template Editor.
Close	Closes the Template Editor.

Bookmarks Menu

The Bookmarks menu contains items used to manage bookmarks in the template. Using bookmarked questions enables the user to display only those template questions that must be answered. See Setting Markers for a description of how the Bookmarks menu can be used to specify the questions that the Template Editor displays.

The following table describes the items in the Bookmarks menu.

TABLE 29   Template Editor Bookmarks Menu

Menu Item	Description
Enable Bookmarks	Enables or disables bookmarking in the template. When bookmarking is disabled, the remaining bookmark menu items are grayed out.
Show Only Bookmarked Questions	Displays only the bookmarked questions.
Mark Current Question	Sets the bookmark for a question selected in the index pane.
Unmark Current Question	Clears the bookmark from a question selected in the index pane. Enabled only if the selected question is bookmarked.
Clear Answer For Current Question	Clears the answer for a selected question in the index pane.Enabled only if the selected question is bookmarked.
Open Group	Expands a selected set of questions in the index pane.
Clear Answers to Bookmarked Questions	Clears the values in all bookmarked questions.
Remove Bookmarks	Removes all bookmarks from the Template.

Search Menu

Use the Search menu items to find the occurrence in a Template of a specific character or value string. When troubleshooting a test run, you can use the Search menu to quickly locate an answer that needs to be changed. See Searching the Template for a detailed description of how to search for character and value strings in a Template.

The following table describes the items in the Search menu.

TABLE 30   Template Editor Search Menu

Menu Item	Description
Find	Opens a dialog box used to search the template for a specific character or value string.
Find Next	Searches the template for the next occurrence of a specific character or value string.

View Menu

Use the View menu to display the Template Editor in Question Mode or in Quick Set Mode, to hide or display the More Info pane, and to display question tag field at the bottom of each question panel. The following table describes the items in the View menu.

TABLE 31   Contents of the Search Menu

Menu Item	Description
Question Mode	Displays the Template Editor in Question Mode.
Quick Set Mode	Displays the Template Editor in Quick Set Mode.
More Info	Displays and hides the More Info pane in the Template Editor.
Question Tag	Displays and hides the Question Tag field in the Template Editor Question pane.
Refresh	Updates the values and questions displayed in the Template Editor.

Help Menu

Use the Help menu to display the online help for the Template Editor and both editor modes. The following table describes the items in the Help menu.

TABLE 32   Template Editor Help Menu

| Menu Item	Description
Configuration Editor	Displays online help for the Configuration Editor.
Template Editor	Displays online help for the Template Editor.
Question Mode	Displays online documentation for the Question Mode.
Quick Set Mode	Displays online documentation for the Quick Set Mode.

Index Pane

When completing a template, the index pane lists the titles of the questions you have answered, are currently answering, and any questions that the editor can determine might need to be answered. The current question is highlighted and its title is displayed at the top of the question pane.

The number of questions displayed in the configuration interview is controlled by setting bookmarks in the template. See Bookmarks Menu for procedures used to reduce the number of questions displayed in the configuration interview.

Click on any question in the index list to make it the current question. Clicking on a question does not cause the list to change. If you change an answer that alters the Template options, the Template Editor updates the questions in the list to reflect the change in options. If a previously answered question is no longer displayed in the index list, the Template Editor saves its answer until you either save the Template or change its value.

You can also use the buttons at the bottom of the Question pane to navigate through the Template file. See Question Pane for a description of the navigation buttons.

Question Pane

The Template Editor displays template questions in the main text area of the editor. You answer the questions using controls such as text boxes, radio buttons, or combo boxes located beneath the question. After you answer each question, click **Next >** at the bottom of the panel to proceed to the next question.

The buttons at the bottom of the Question pane control the following functions:

- **Back** - Returns to the previous question.
- **Last** - Advances as far as possible through the template.
- **Next** - Proceeds to the next question.
- **Done** - Saves your answers as a template and closes the Template Editor.

See Creating a Template for information about using the Question pane to create a template.

See Editing a Template for information about using the Question pane to edit the current template.

More Info Pane

To open and close the More Info pane, choose View > More Info from the menu bar.

The More Info pane provides additional information about each question, including the following:

■ Background information about the question

■ Information about choosing an answer

■ Examples of answers

See Keyboard access for a description of how the keyboard can be used to navigate the More Info pane.

Quick Set Mode

Quick Set Mode uses tabbed panes to display only the runtime values of a template.
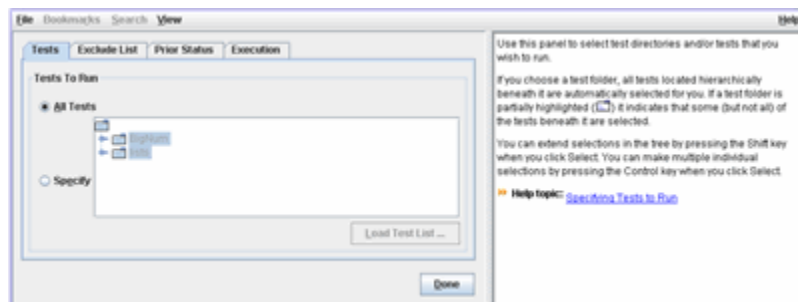
Runtime values are used to choose the tests that are run and specify how tests are run. These values can change from test run to test run. Quick Set Mode displays only the runtime values that apply to your test suite.The tabs containing runtime values that do not apply to your test suite are not displayed in Quick Set Mode.

■ Tests

■ Exclude List (optional)

■ Keywords (optional)

■ Prior Status

■ Environment (optional)

■ Execution (optional)

Quick Set Mode is not enabled until a template is either loaded or started in the Template Editor. See Editing a Template for detailed information about editing a template.

Log Viewer

Use the Log Viewer to view the log contents during or after a test run. The test suite generates the logs that are displayed in the viewer. The following illustration is an example of the content that a test suite might generate. See your test suite documentation for detailed information about the log files that it generates.
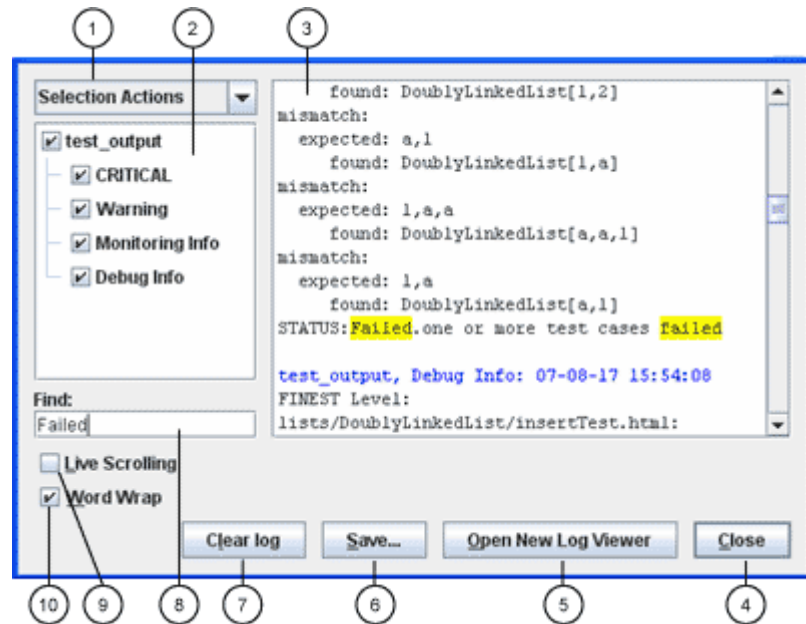
1. Selection Action list

2. Logs and levels pane

3. Log Message pane

4. Close button

5. Open New Log Viewer button

6. Save button

7. Clear Log button

8. Find text field.

9. Live Scrolling check box

10. Word Wrap check box

Selection Action List

The Selection Action list consists of three sections of message selection actions (log levels) that a user might choose for displaying the message of a log. A log is a series of notifications concerning a component of the system. Log levels are descriptive criteria applied to each message within a log.

The first section of the list enables the user to either Select All log levels for display in the log viewer or to Unselect All. The second section of the list enables the user to select one or more general message log levels for display. In the example illustration, these messages are represented by the General Logger messages. The third section of the list enables the user to select one or more test suite log levels for display. In the example illustration, these message types are represented by the TSM Logger message types.

The choices in this list only change the current state of selections. These selections would not apply to the new log which is displayed after the user chooses Select All.

Logs and Levels Pane

The log viewer displays a list of log levels available from the test suite that has been selected for display. You can select one or more of the log levels in any combination. These log levels are also listed in the Selection Action list. As in the Selection Action list, the log levels are grouped into general types. See Selection Action List for a description of logs and log levels. See your test suite documentation for detailed information about the log messages that it generates.

Log Message Pane

The contents of the output log specified by the selections in the logs and levels pane are displayed as text. The test suite specifies the output log contents displayed in the viewer. See your test suite documentation for detailed information about the log file contents that it generates.

Close Button

The Close button closes the log viewer. If multiple log viewers are open during a test run to monitor specific sets of log messages, the Close button only closes the log viewer to which it is bound.

Open New Log Viewer Button

Use the Open New Log Viewer button to open an empty log viewer. During a test run, multiple log viewers can be opened and used to monitor specific sets of log messages.

Save Button

Click the Save button to save the contents of the Log Viewer as an XML file.

Clear Log Button

Click the Clear Log button to erase the contents displayed in the log viewer and usually stored on a disk. These contents are automatically loaded the next time that the work directory is opened.

Find Text Field

Use the Find Text field to search the contents of the log message pane for specific text strings.

Live Scrolling Check Box

Check to monitor for log output while the harness runs. Log entries can be generated at any time, not just while tests are running.

Word Wrap Check Box

Check the Work Wrap check box to cause the log viewer to wrap long line of content in the log message pane.

5

Configuring a Test Run

Before the harness can execute the tests in a test suite, it requires information about how your computing environment is configured. You provide the harness with this information by loading a template (`.jtm`), loading an existing configuration file (`.jti`), creating a new configuration, or changing the values in a configuration.

Note – You can also specify configuration values from the command line when starting the harness GUI.

The quantity and scope of information required to run tests depends on the test suite. Some test suites run in diverse environments (different platforms and networks), while others run in very specific, well-defined environments. The test suite may provide a configuration interview or a template for you to use in creating a configuration for your test run. If your test suite does not provide any of these, consult the test suite documentation for directions about how you can supply the required configuration information.

This chapter contains the following topics:

- <u>Editing a Configuration</u>- Describes how to use the Question Mode edit a configuration.
- <u>Editing Quick Set Values</u>- Describes how to use the Quick Set Mode to edit the runtime values in a configuration file.
- <u>Creating a Configuration</u>- Describes how to create a configuration for use in running tests.
- <u>Saving a Configuration</u> - Describes how to save the current configuration.
- <u>Loading a Configuration</u> - Describes how to load an existing configuration or a template for use in running tests.
- <u>Using Bookmarks in Configurations</u> - Describes how to use bookmarks to simplify a configuration interview.
- <u>Searching a Configuration</u> - Describes how to search a configuration for characters or values.

■ <u>Working with Multiple Configurations</u> - Describes how to use multiple configuration files to switch configurations between test runs.

Creating a Configuration

You must create a configuration for the test run if an existing configuration is not provided by the test suite or your test group.

Note – If you have an existing configuration, load it in the Test Manager (see <u>Loading a Configuration</u>) and then edit it (see <u>Editing a Configuration</u>) as required for your test run.

Before creating a configuration, if you have a template provided by your test group or by the test suite, you can load it in the Test Manager (see <u>Loading a Template</u>) and use it as the basis for creating a configuration. You can also create a template (see <u>Creating a Template</u>), if an existing template is not available, and use it as the basis for creating a configuration.

▼ Create a New Configuration

To create a configuration for the test run, perform the following steps:

1. If you have an appropriate template provided by your test group or by the test suite, load it in the Test Manager (see <u>Loading a Template</u>).

   You can create a template (see <u>Creating a Template</u>), if an appropriate, existing template is not available.

   If you choose not to use a template, go to Step 2.

2. Choose Configure > New Configuration from the Test Manager menu bar or File > New Configuration from the Configuration Editor menu bar.

3. Answer the questions displayed in the Configuration Editor window.

   If you are creating a configuration from a template, the values set in the template are applied to the questions.

Note – The Configuration Editor window displays questions in the center pane (see the Question Pane). Use the text box, radio button, or combo box controls located beneath the question to provide the required configuration information. After you answer each question, click the Next button to proceed to the next question. Some questions provide information and do not require an answer. In these cases, click the Next button to proceed to the next question.

You can go backward and forward to any question to review or change your answer by doing one of the following:

- Choosing a question directly from the Index pane
- Clicking the Back button, the Next button, or the Last button
  As you move backward and forward, the Configuration Editor window saves all answers until you either save the configuration or change the answers. Choose File **>** Save at any time to save your answers and position in a configuration file. See Saving a Configuration.

4. After you complete the configuration, click the Done button to save the configuration and close the Configuration Editor window.

▼ Loading a Configuration

To load an existing configuration, perform the following steps:

1. Choose Configure **>** Load Configuration from the Test Manager menu bar, or, if you have an open Configuration Editor window, choose File **>** Load Configuration from the menu bar.

The harness opens the Load Configuration dialog box.
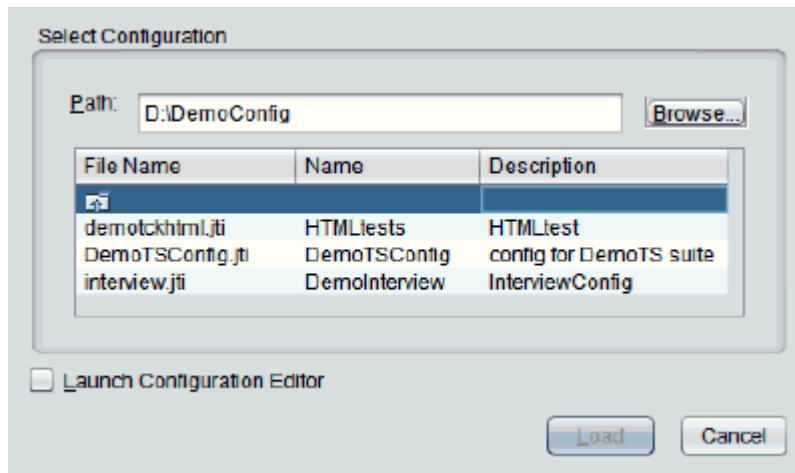
If your test group provides an existing configuration, you can use it to run tests.

2. Locate and select the configuration file (`.jti`).

3. Click the Load button.

The harness loads the file and closes the Load Configuration File chooser.

If you loaded a configuration file that you must edit before running tests, use the Configuration Editor to set appropriate values for your test environment. See Editing a Configuration.

▼ Editing a Configuration

To edit a configuration, perform the following steps:

1. Choose Configure > Edit Configuration in the Test Manager menu bar.

The harness opens the Configuration Editor in Question Mode and loads the current configuration.

In Question Mode, the Configuration Editor enables the Bookmarks feature. Use the Bookmarks feature to simplify an interview by displaying only questions whose answers might change. See Using Bookmarks in Configurations for the steps required to set, clear, and manage Bookmarks in a configuration. See Question Mode for detailed information about using the features of this mode to edit a configuration.

Quick Set Mode enables rapid editing of frequently changed values. If you are editing only these values, you can change to the Quick Set Mode. You can use the View menu at any time to change from one mode to the other. Because only one configuration file is loaded in the Configuration Editor at a time, any changes that you made in one mode are automatically reflected in the other mode. See Editing Quick Set Values for detailed information about using the features of this mode to edit a configuration.

2. In Question Mode, navigate to the question by one of the following means:

■ Use the Search menu to locate specific characters or character strings in the titles, questions, and answers. See Searching a Configuration.

■ Choose the question directly from the Index pane.

■ Click the Back button or Next button as required to locate the question.

3. Review and change configuration values as necessary.

When you move backward and forward through the list of questions, the Configuration Editor preserves your previous answers until you change them.

4. After making all required changes to the configuration, save the changed configuration by one of the following means:

■ If you are using a configuration template or an existing configuration file to create a new configuration file, choose File > Save As from the Configuration Editor menu bar. The Configuration Editor opens a dialog box for you to use in setting the location and name of the new configuration file.

■ If you are modifying an existing configuration file used to run tests, click the Done button. The Configuration Editor saves the changes to the configuration file and closes. When the Done button is enabled, you can click it regardless of your location in the configuration.

Editing Quick Set Values

The following topics describe how values displayed in the Quick Set Mode can be changed for a test run.

■ Specifying Tests to Run - Describes how to specify the individual tests or groups of tests in a test run.

■ Using Exclude Lists - Describes how to specify the exclude list used for a test run.

■ Using Known Failure Lists - Describes how to specify a known failures list used for a test run (analysis).

- Specifying Keywords (Only enabled if used by the test suite) - Describes how to specify keywords to filter the tests that are run.
- Specifying Prior Status - Describes how to filter the tests that are run based on their prior results.
- Setting Concurrency and Timeout Factor - Describes how to set the concurrency and timeout values of a test run.

▼ Specifying Tests to Run

To directly specify the tests that are run, perform the following steps:

1. Click the ☰Quick Set Mode button on the Test Manager tool bar or choose Configure > Edit Quick Set > Tests To Run in the menu bar.

The Configuration Editor opens in Quick Set Mode.
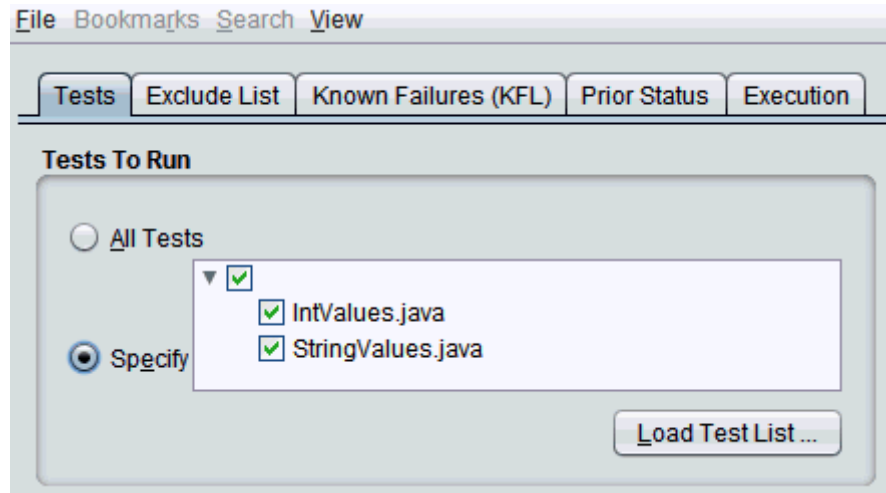
Note – In Question Mode, use the How to Specify Tests question to specify tests that are run.

2. Click the Tests tab if it does not have focus.

3. Choose the Specify radio button.

The Configuration Editor enables the tree and the Load Test List button.

4. Use the run tree or the Load Test List button to specify the tests that are run.

See the following topics for detailed information about using the run tree or a load test list:

- Run Tree
- Load Test List

5. Click the Done button to save the configuration change.

Run Tree

In the Tests to Run pane, you can use the run tree to specify individual tests and folders for the harness to run. The Tests pane highlights all folder and test icons selected for the test run. You can make individual selections in the tree by pressing the Control key when you click an icon or name. To select a series or sequence of tests or folders, press the Shift key and then click the first and the last icon or name in the sequence.

When you select some of the tests in and under a folder, the Test pane partially highlights the folder icon. If you choose a test folder, the harness selects all tests in the test suite under that location for the test run. If you choose one or more tests, the harness selects those individual tests for the test run.

The harness walks the tree starting with the sub-branches or tests you specify and executes all tests not filtered out by the exclude list, keyword, or prior status.

Note – Restrictions are applied cumulatively. For example, you can specify the tests in a test suite, then restrict the set of tests using an exclude list, and then further restrict the set to only those tests that passed on a prior run.

Load Test List

When you click the Load Test List button, the harness opens a dialog box for you to use in locating and selecting the test list file. The test list file is a .txt file that contains a list of tests. See Test List File Format and Creating a Test List File for a detailed description of the test list file.

When the file is loaded, the harness updates the run tree to indicate the tests specified in the test list file.

Test List File Format

The format requirements of the test list file are as follows:

- Line-oriented

- Relative test paths
- Blank lines and lines beginning with the pound symbol (#) are ignored
- On all other lines, the first item (up to the first whitespace) is taken as a test name and the remainder of the line is ignored

The format of the load list file is compatible with the format of the summary.txt generated as part of a report.

▼ Creating a Test List File

To create a test list file, perform the following steps:

1. Select the lines from summary.txt that define the tests you want to run.

Note – You can use other sources for obtaining the path names of the tests that you want to include in the test list file. On a Solaris platform, you might use awk, grep, or similar utilities to identify lines that specify the tests with failed or error results.

Example lines from summary.txt:

```
BigNum/compareTest.html Failed. exit code 1
 BigNum/equalsTest.html Failed. exit code 1
 BigNum/longConstrTest.html Failed. exit code 1
 BigNum/subtractTest.html Failed. exit code 1
 lists/DoublyLinkedList/appendTest.html Failed. exit code 1
 lists/DoublyLinkedList/equalsTest.html Failed. exit code 1
 lists/DoublyLinkedList/insertTest.html Failed. exit code 1
 lists/DoublyLinkedList/removeTest.html Failed. exit code 1
 lists/LinkedList/appendTest.html Failed. exit code 1
 lists/LinkedList/equalsTest.html Failed. exit code 1
 lists/LinkedList/insertTest.html Failed. exit code 1
 lists/LinkedList/removeTest.html Failed. exit code 1
 lists/SortedList/equalsTest.html Failed. exit code 1
 lists/SortedList/insertTest.html Failed. exit code 1
 lists/SortedList/removeTest.html Failed. exit code 1
```

2. Copy and paste the relative paths into a .txt file.

3. Remove all text that is not part of the test path.

Example lines in the .txt file:

```
 BigNum/compareTest.html
 BigNum/equalsTest.html
 BigNum/longConstrTest.html
 BigNum/subtractTest.html
 lists/DoublyLinkedList/appendTest.html
```

```
lists/DoublyLinkedList/equalsTest.html
lists/DoublyLinkedList/insertTest.html
lists/DoublyLinkedList/removeTest.html
lists/LinkedList/appendTest.html
lists/LinkedList/equalsTest.html
lists/LinkedList/insertTest.html
lists/LinkedList/removeTest.html
lists/SortedList/equalsTest.html
lists/SortedList/insertTest.html
lists/SortedList/removeTest.html
```
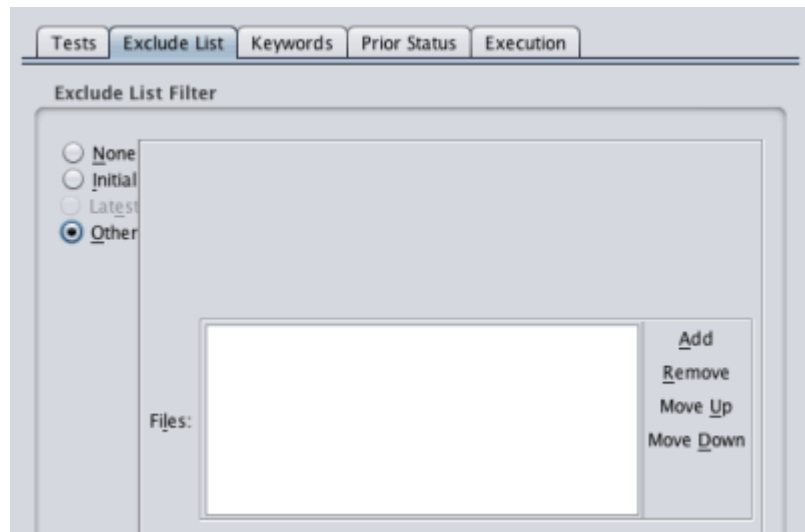
4. Save the `.txt` file using a descriptive name.

Example `.txt` file name:

```
BigNum_ListsfailedTests.txt
```

▼ Using Exclude Lists

1. Click the ☰ button in the Test Manager toolbar or choose Configure > Edit Quick
   Set > Exclude List from the Test Manager menu bar.

The Configuration Editor opens in Quick Set Mode.

> **Note –** Exclude list files contain a list of tests in a test suite that are not run by the harness. You can also use Question Mode view to specify exclude lists.

2. Click the Exclude List tab if it does not have focus.

3. Use the buttons in the tabbed pane to set exclude list selection options and to add or remove exclude lists used to run tests in your computing environment.

See <u>Exclude List Selection Options</u>, <u>Latest Exclude List</u>, and <u>Other Exclude List</u> for a detailed description of these values and how they are used by the harness when running tests.

4. Click the Done button to save the configuration change.

Exclude List Selection Options

Use the following selections to specify the exclude list option used to run tests:

- **None** - An Exclude List is not used.
- **Initial** - Only enabled if the test suite provides an exclude list. If you choose Initial, the tests are run using the exclude list provided by the test suite.
- **Latest** - Only enabled if the test suite provides a location for updated exclude lists. If you choose Latest, additional options are displayed. See <u>Latest Exclude List</u> below for detailed information.
- **Other** - A custom exclude list can be used. See <u>Other Exclude List</u> below for detailed information.

If a complete test is added to the exclude list, the harness updates the test result status without requiring the test be rerun. However, if only a test case from a test is added to the exclude list, the harness requires that you rerun the test using the updated exclude list before updating the test result status.

Latest Exclude List

If your test suite provides a URL for the latest test suite, the harness enables the Latest exclude list option. The following table describes the text and controls displayed when you choose the Latest exclude list option.

TABLE 33   Latest Exclude List Dialog Box Contents

Text and Controls	Description
Location:	Displays the location of the exclude list specified by the test suite. This is a non-editable field.
Last updated:	Displays the date that the exclude list was last updated. This is a non-editable field.
Check For Updates Automatically	Causes the harness to automatically check the location of the exclude list and compare the date-time stamps of the remote and local exclude lists. The harness then displays a dialog box advising you of the results. If a new exclude list is available, you can choose to download it.
Every _ Days	Sets an interval for the harness to automatically check the remote location of the exclude list for updates.
Every Test Run	Causes the harness to check the remote location of the exclude list for updates before each test run.
Check Now	Causes the harness to check the remote location of the exclude list for an update.

Other Exclude List

The following table describes the buttons displayed when you choose the Other exclude list option.

TABLE 34   Other Exclude List Dialog Box Contents

Button	Description
Add	Selects an exclude list file for your test suite. As you make selections with the file chooser dialog box, they are added to the list. After you add an exclude list, you can modify the list.

TABLE 34   Other Exclude List Dialog Box Contents

Remove	Clears an item from the list. Select an item in the list and click Remove.
Move Up	Moves an item one position higher in the list. Select an item in the list and click Move Up.
Move Down	Moves an item one position lower in the list. Select an item in the list and click Move Down.

▼ Using Known Failures Lists

A *known failures list* (KFL) is a user-authored text file with the extension `.kfl`. The KFL file lists tests in a test suite that are known to fail. The failing tests are identified by path names that uniquely identify test names and test cases. Known failures list files are maintained by users (they are not distributed as part of a test suite, unlike an exclude list). Currently the known failures information is used to generate and filter information in test reports. See the help topic <u>KFL Options in HTML Reports</u> for more information on creating and using a known failures list.

1. Click the ☰ button in the Test Manager toolbar or choose Configure > Edit Quick Set > Known Failures List from the Test Manager menu bar.

The Configuration Editor opens in Quick Set Mode.
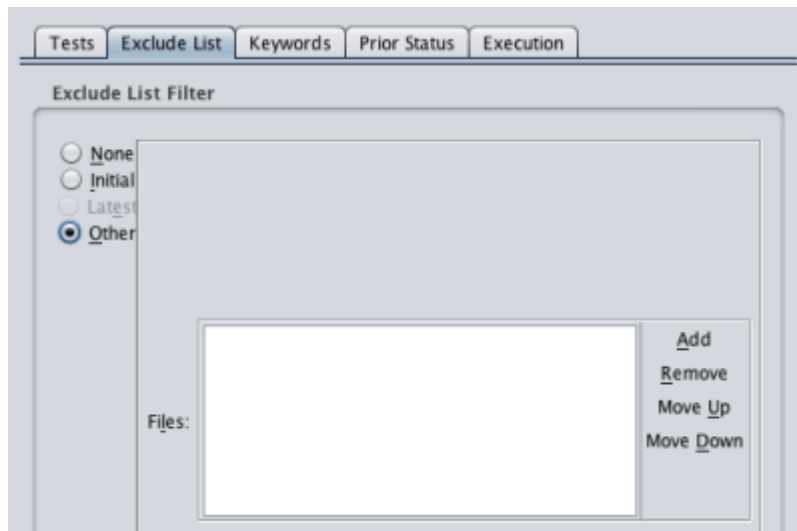
2. Click the Known Failures (KFL) tab if it does not have focus.

3. To specify one or more KFL filters, choose the Select Files radio button.
   Click Add to open a file chooser dialog. Locate your KFL list and click Select. You can continue to add filters, using the Move buttons to organize them as you see fit. Note, the order of the files displayed does not have any effect on the report outcome.
   Click Done when you are finished.

4. Click the Done button to save the configuration change.

For more information, see <u>KFL Options in HTML Reports</u>, <u>Known Failures List Reports</u>, and <u>Known Failure Analysis</u>.

▼ Specifying Keywords

To directly specify the tests that are run, perform the following steps:

1. Click the ☰ Quick Set Mode button on the Test Manager tool bar or choose Configure > Edit Quick Set > Keywords in the menu bar.

The Configuration Editor opens in Quick Set Mode.
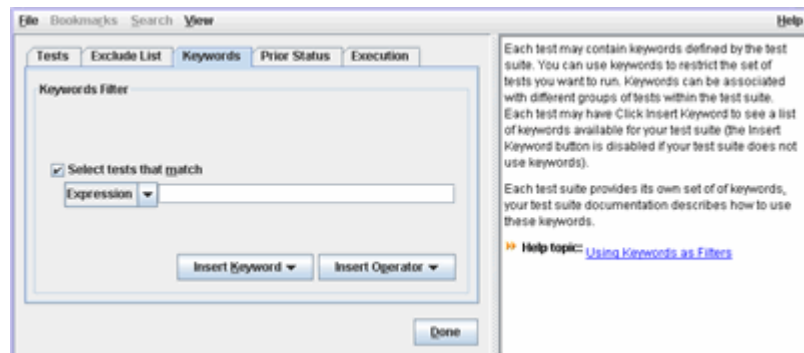
2. Click the Keywords tab if it does not have focus.

3. Click the Select tests that match checkbox.

The harness enables the Expression, Insert Operator, and Insert Keyword buttons.

4. Build an expression in the text field by using any logical combination of the following actions:

■ Click the Expression button to display a list of expressions that can be constructed. From the list, choose the type of expression that you are building. See List of Expressions for a description of the available types of expressions.

■ Click the Insert Keyword button to display the list of keywords provided by the test suite for use in filtering tests (this is only available if the test suite has information). From the list, choose one or more keywords used in the expression.

■ Click the Insert Operator button to display a list of logical operators that you can include to construct boolean expressions in the text field. From the list, choose an operator to include it in the expression. See List of Logical Operators for a description of the available operators.

5. Click the Done button to save the configuration change.

List of Expressions

The following table provides descriptions and examples of keyword expressions that can be constructed.

TABLE 35   Keyword Expressions

Expression	Description

TABLE 35   Keyword Expressions

Any Of	Runs all tests in the test suite having any of the keywords entered in the text field.
	Example:
	A test suite uses the keyword `interactive` to identify tests that require human interaction, and `color` to identify tests that require a color display.
	To execute only the tests containing the `interactive` keyword, choose Any Of and then use the Insert Keyword button to choose the `interactive` keyword.
All Of	Runs all tests in the test suite having all of the keywords entered in the text field.
	Example:
	To execute only the tests containing both the `interactive` and `color` keywords, choose All Of and then use the Insert Keyword button to choose the `interactive` and `color` keyword.
Expression	Runs all tests in the test suite having the expression entered in the text field.
	Use the Insert Keyword and the Insert Operator buttons to construct a Boolean expression in the text field. Keywords stand as Boolean predicates that are true if, and only if, the keyword is present in the test being considered. A test is accepted if the overall value of the expression is true. All other tests are rejected by the restriction.
	Example:
	A test suite uses the keyword `interactive` to identify tests that require human interaction, and `color` to identify tests that require a color display.
	To execute only the tests with the `color` keyword that do not also contain the `interactive` keyword, choose Expression and then use the Insert Keyword button to choose the `color` keyword, the Insert Operator button to choose the ! operator, and the Insert Keyword button to choose the `interactive` keyword.

List of Logical Operators

Logical operators are only available when Expression is selected in the list of expressions. The following table provides descriptions and examples of logical operators that can be used to build keyword expressions. The precedence column

indicates the order in which the expression is resolved. Expressions in parentheses are evaluated first, with nested parentheses being evaluated from the innermost parentheses outward.

TABLE 36 Keyword Operators

| Logical Operator	Precedence	Description
( ) group	1	Used to create groups of expressions.
		Example:
		A test suite uses the keyword `interactive` to identify tests that require human interaction and `color` to identify tests that require a color display.
		`!(interactive&color)`
		The harness will exclude tests that include both keywords.
! not	2	Logical not. Used to exclude tests containing the expression.
		Example:
		A test suite uses the keyword `interactive` to identify tests that require human interaction and `color` to identify tests that require a color display.
		`!interactive&!color`
		The harness will exclude tests that include either keyword.
& and	3	Logical and. Used to combine expressions.
		Example:
		A test suite uses the keyword `interactive` to identify tests that require human interaction and `color` to identify tests that require a color display.
		`interactive&`color
		The harness will only choose tests that include both keywords.
| or	4	Logical or. Used to specify either of two expressions.
		Example:
		A test suite uses the keyword `interactive` to identify tests that require human interaction and `color` to identify tests that require a color display.
		`interactive|`color
		The harness will only choose tests that include either keyword.

▼ Specifying Prior Status

To specify prior status used to restrict the set of tests that are run, perform the following steps:

>

1. Click the ▬Quick Set Mode button on the Test Manager tool bar or choose
   Configure > Edit Quick Set > Prior Status in the menu bar.

The Configuration Editor opens in Quick Set Mode and displays the Prior Status tab.
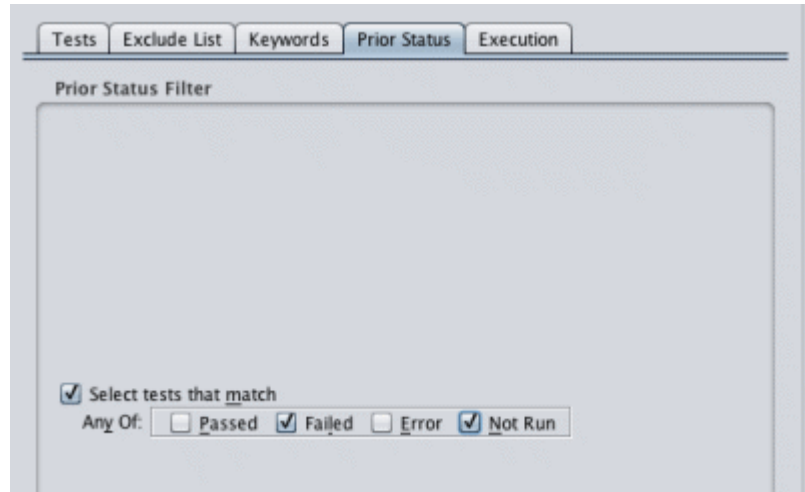
Note – In Question Mode, use the Status question to specify prior status used to
restrict the set of tests that are run.

---

2. Check the Select tests that match option.

The Configuration Editor enables the prior status check boxes.

3. Check the one or more of the prior status conditions.

See Prior Status Selections for detailed descriptions of the available choices and how
the harness uses them to filter the tests that are run.

4. Click the Done button to save the configuration change.

Prior Status Filter Selections

By choosing Select tests that match, you can run tests with restrictions based on their result from a prior test run. The following table describes the available Prior Status filter selections.

TABLE 37 Using Prior Status as a Filter

Prior Status Filter	Action
Passed	Selects tests with passing results the last time the test was executed.
Failed	Selects tests with failed results the last time the test was executed.
Error	Selects tests that the harness could not execute the last time it was included in a test run.
Not Run	Selects tests without results in the current work directory.

Prior status is evaluated on a test-by-test basis using information stored in the result files (`.jtr`) that are written in the work directory. Unless overridden by a test suite, a result file is written in the work directory for every test that is executed. When you change work directories between test runs, the result files in the previous work directory are no longer used and, if the new work directory is empty, the harness behaves as though each test in the test suite was not run.

You can also use the Prior Status setting in combination with the Current Configuration view filter to display only those test and folder status icons that match the specified prior status. The test tree displays all other tests and folders as gray, filtered out ▢folder and ▢test icons. During a test run, when a test result no longer matches the prior status filter, the test tree changes the test and folder icons to gray, filtered out ▢folder and ▢test icons.

For example, if you only want to monitor tests in a test suite that had failed results you would set the Prior Status filter to Any Of Failed and repeat the test run. As tests pass, the test tree changes their icons from failed to filtered out, indicating that they no longer match the Prior Status filter Any Of Failed.

Note – It is often useful to choose all of the status values except Passed for the first few test runs, then refine the filtering to reduce the number of tests in subsequent runs.

▼ Setting Concurrency and Timeout Factor

To set the concurrency and the timeout factor, perform the following steps:

1. Click the ☰Quick Set Mode button on the Test Manager tool bar, choose Configure > Edit Quick Set > Concurrency, or choose Configure > Edit Quick Set > Timeout Factor in the menu bar.

The Configuration Editor opens in Quick Set Mode.
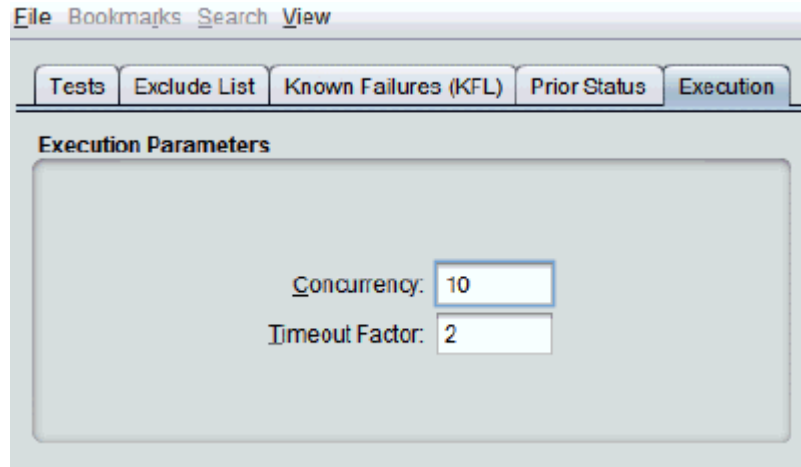
Note – You can also use the Configuration Editor window in Question Mode to specify the concurrency and the timeout factor.

2. Click the Execution tab if it does not have focus.

3. Use the text fields in the tabbed pane to set the concurrency and the timeout factor values used to run tests in your computing environment.

See Concurrency and Time Factor for a detailed description of these values and how they are used by the harness when running tests.

4. Click the Done button to save the configuration change.

Concurrency

The harness can run tests concurrently. If you are running the tests on a multiprocessor computer or are using multiple agents on a test system, concurrency can reduce the time required to run tests. For detailed information about using agents to run tests, refer to your test suite documentation and to the *JavaTest Harness Agent User's Guide* if it is provided by the test suite.

When using multiple agents to run tests, the concurrency value must not exceed the number of agents. If the concurrency value exceeds the total number of available agents, an error will occur in the test run.

If you have unexpected test failures, run the tests again, one at a time. Some test suites may not work correctly if you run tests concurrently. The default range of values used by the harness is from 1 to 50.

For your first test run, leave this field set to 1. After the tests run properly, you can increase this value. Unless your test suite restricts concurrency, the maximum *number* of threads specified by the `concurrency` command is 50. See your test suite documentation for additional information about using concurrency values greater than 1.

This field is disabled for some test suites.

Time Factor

To prevent a stalled test from stopping a test run, most test suites set a timeout limit for each test. The timeout limit is the amount of time that the harness waits for a test to complete before moving on to the next test.

If you are running the tests on a particularly slow CPU or slow network, you can change the time limit by specifying a floating point value in the time factor field. Each test's timeout limit is multiplied by the time factor value. The default range of values used by the harness is from 0.1 to 100.0.

Note – In the Time Factor field, the harness uses the form of floating point values that is specific to the locale in which it is run. For example, if your locale uses floating point values in the form of x,x, harness uses that form of floating point value in the Time Factor field. In setting the timeout factor in the following example, specify values of 2,0 and 0,5 if your locale uses floating point values in the form of x,x.

Example:

If you specify a value of `2.0`, the timeout limit for tests with a basic 10-minute time limit becomes 20 minutes. Specifying a value of `0.5` for tests with a 10-minute limit produces a 5-minute timeout limit.

At first, use the default value of `1.0` to run tests and then, if necessary, increase the value. The actual timeout calculation for any particular test suite might vary.

▼ Saving a Configuration

To save your current answers and position while editing or creating a configuration, choose either File > Save or File > Save As from the Configuration Editor menu bar.

If you are creating a new configuration, the harness opens a dialog box for you to use in setting the location and name of the new configuration file.

If you use different configurations to run a test suite, see <u>Working with Multiple Configurations</u>.

Using Bookmarks in Configurations

In Question Mode, you can use Bookmarks to restrict the set of questions displayed in the Configuration Editor (Bookmarks are not used in Quick Set Mode). Bookmarks are persistent and are saved in the `.jti` file, and reloaded when the configuration is used again. When bookmarks are disabled, they are preserved in program memory but cannot be manipulated.

▼ Set Bookmarks for Specific Questions

To set bookmarks for specific questions in the configuration, perform the following steps:

1. Click the Bookmarks > Enable Bookmarks check box in the menu bar.

2. Click a question in the Configuration Editor.

3. Set a bookmark for the highlighted question by performing one of the following actions:

■ Click to the left of the highlighted question text.

■ Choose Bookmarks > Mark Current Question in the menu bar.

- Right click the highlighted question and choose the Mark Current Question menu item in the pop-up menu.

▼ Display Questions With Bookmarks

To only display the questions with bookmarks, perform the following step:

- Click the Bookmarks > Show Only Bookmarked Questions checkbox in the menu bar.

The Configuration Editor displays bookmarked questions in the following manner:

- The first question in the interview is displayed.
- Questions with bookmarks are displayed.
- If the interview is complete (all questions have valid answers) the final question is displayed.
- If the interview is incomplete (one or more questions have invalid or incomplete answers), the questions from the last marked question to the first question with an invalid or incomplete answer are displayed.
- Sequences of questions not described by a previous category are grouped and represented by three dashes (---) in the list. These groups can be opened to display the complete sequence of questions.

▼ Open Groups of Questions

To open a group of hidden questions, perform the following steps :

1. Click the --- section in the list of questions.

Groups of hidden questions without bookmarks are represented by three dashes (---) in the list of questions.

2. Click the Bookmarks > Open --- Group menu item from the menu bar or right click the highlighted --- section and choose Open --- Group menu item from the pop-up menu.

▼ Close Groups of Questions

To clear the values of all questions with bookmarks, perform the following steps:

1. Click a question in the list of questions.

2. Click the Bookmarks > Clear Answers to Bookmarked Questions menu item from the menu bar.

▼ Clear the Value of a Specific Question

To clear the value of a specific question, perform the following steps:

1. Click the question in the list of questions.

2. Click the Bookmarks > Clear Answer For Current Question menu item or right click the highlighted question and choose Clear Answer For Current Question from the pop-up menu.

▼ Remove All Bookmarks

To remove the bookmarks from the configuration, perform the following step:

■ Choose the Bookmarks > Remove Bookmarks menu item.

The configuration list reverts to the full configuration list.

▼ Remove a Bookmark From a Question

To remove the bookmarks from a question in the configuration, perform the following steps:

1. Click the question in the list of questions.

2. Click the Bookmarks > Unmark Current Question menu item or right click the highlighted question and choose Unmark Current Question menu item from the pop-up menu.

Searching a Configuration

When using the Configuration Editor in Question Mode, you can locate and display the panes containing a specific character string. The harness can search titles, questions, and answers for matching characters. It does not search the More Info.

▼ Search for Characters in Configurations

To search for character strings in the configuration titles, questions, and answers, perform the following steps:

1. Click the ▤ Question Mode button on the Test Manager tool bar or choose Configure > Edit Configuration in the menu bar.

The Configuration Editor opens in Question mode.

2. Choose Search > Find.

The harness displays the Find Question dialog box.
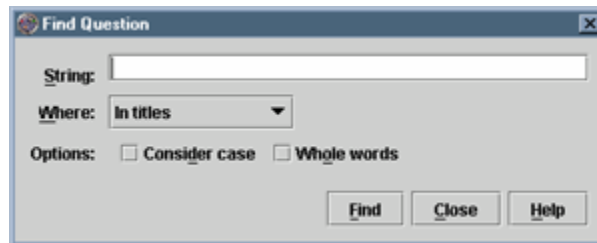
3. Enter the search character string in the String field.

See the String field description in [Search Criteria](#).

4. Choose the search location from the Where drop down list.

See the Where description in [Search Criteria](#).

5. Choose any required search options.

See the Options descriptions in [Search Criteria](#).

1. Click the Find button to search for the character string.

To repeat the search, either click the Find button or use the Search > Find Next menu item from the Configuration Editor menu bar.

2. Click the Close button to dismiss the dialog box.

Search Criteria

The following table describes the search criteria used in the Find Question dialog box.

TABLE 38   Find Question Dialog Box Search Criteria

Item	Description
String	Enter the character string that you are trying to find.
Where	Choose where you want to search:
	• In titles
	• In text of questions
	• In answers
	• Anywhere
Options: Consider case	Specifies that the search pattern match the case of the characters in the Find text field.
Options: Whole words	Specifies that the search pattern only match whole words from the Find text field.

Working with Multiple Configurations

In some testing situations it is useful to use separate configuration files to switch between different configurations for different test runs. For example, one configuration could select automated tests and another could select interactive tests.

Prior to running tests, use the Configure menu to load the required configuration file. See Loading a Configuration for detailed information. These configuration files can be loaded from a central resource provided by your test group. See your test group for the name and location of the configuration files.

If your group does not provide an existing set of configuration files, you can create them by using the Configuration Editor to edit an existing configuration and then save each variation to a file name of your choosing. See Editing a Configuration for detailed information.

You can save these configuration files anywhere in your file system. Generally, however, the work directory should not be used to save configuration files. Clearing the work directory would delete the configuration file.

After these configuration files are created, they can be used by your test group to run tests.

6

Creating and Editing Templates

The harness provides users with the optional ability to create and edit templates that can be used to run tests on multiple test systems or test platform.

Note – Your test suite might not enable creating or editing templates. The menu items and features described here will not appear in your user interface if your test suite does not support templates.

For example, if your test suite enables creating and editing templates and your test group uses a central location to provide and manage the resources required to run tests (the test suite, report directories, configuration files, or the harness), you can create a template that contains all known configuration values required by the test group to run tests. Each user can load the template from the central site, provide only those values unique to their test environment or test run, and run their tests using their completed configuration.

See Creating a Template for a detailed description of the process used to create a template.

See Editing a Template for a detailed description of the process used to change the contents of a template.

In addition to providing users with partially completed configurations, templates can further simplify the process of creating and maintaining configurations through the use of the following features:

■ Bookmarks

■ Propagation

An optional feature of templates is the ability to propagate (inherit) template updates to all configurations based on that template. When the test suite developer enables template propagation, the harness checks for template updates when a user does any of the following:

■ Starts a test run.

- Opens a work directory.
- Loads a configuration.

For a configuration to receive template updates, the test suite developer must enable propagation in both the configuration and its template.

▼ Creating a Template

To create a template, perform the following steps:

1. Choose Configure **>** New Template from the Test Manager menu bar.

2. Answer the questions displayed in the Template Editor.

   The Template Editor displays questions in the center pane (the Question Pane). After answering a question, click the Next button to proceed to the next question.

3. Choose File **>** Save As from the menu bar and save the template with a relevant name.

   Template file names are automatically appended with a `.jtm` extension.

Note – Changes to a template propagate to configurations based on the template *only* if propagation is enabled in both the template and the derived configurations.

Editing a Template

The Template Editor provides the following two modes of displaying and editing the questions and values in a template.

See <u>Question Mode</u> for detailed information about using the features of this editing mode.

- Question Mode - Enables editing of all template questions.
- Quick Set Mode - Enables rapid editing of a limited set of template questions.

   Quick Set Mode filters out all test suite specific questions and displays a limited

set of general questions whose values are often changed from test run to test run. In this editing mode, the Template Editor creates logical groupings of the values and displays them in tab form for easy user navigation and access.

▼ Editing in Question Mode

To edit all template questions and bookmarks in a template, perform the following steps:

1. If you are editing the current template (named at the bottom of the Test Manager) choose Configure > Edit Template in the Test Manager menu bar, otherwise choose either Configure > Load Template or Configure > Load Recent Template.

   If you are editing the current template (named at the bottom of the Test Manager), the harness opens the Template Editor and loads the template. Go to step 3.

   If you are editing a different template, the harness displays the Load Template dialog box.

2. Use the dialog box to locate and load the template.

   The harness opens the Template Editor and displays the template in Question Mode.

3. Use Question Mode to change the values of questions in the template.

4. Click Done to save the changes and close the Template Editor.

You can use the View menu at any time to change from one mode to the next. Because there is only one template file loaded in the Template Editor at a time, when you change modes during an editing session, any changes made in one mode are automatically reflected in the other.

If a configuration is attached to the edited template, the harness opens a Template updated dialog box that enables you to review the changes and to accept, reject, or postpone having the changes propagated to the attached configuration. See Resolving Configuration-Template Conflicts for a description of the dialog box and the choices available to the user.

▼ Editing in Quick Set Mode

To edit one or more of the quick set values in a template, perform the following steps:

1. If you are editing the current template (named at the bottom of the Test Manager) choose Configure > Edit Template in the Test Manager menu bar, otherwise choose either Configure > Load Template or Configure > Load Recent Template.

   If you are editing the current template (named at the bottom of the Test Manager), the harness opens the Template Editor and loads the template. Go to step 3.

   If you are editing a different template, the harness displays the Load Template dialog box.

2. Use the dialog box to locate and load the template.

   The harness opens the Template Editor and displays the template in Question Mode.

3. Choose View > Quick Set Mode in the Template Editor menu bar.

   The harness displays the Template Editor in Quick Set Mode.

4. Set the quick set values by using the following tabs:

   - <u>Tests</u> - Use this tab to edit the default set of tests that a user can run in this configuration.

   - <u>Exclude List</u> - (Optional) Use this tab to edit the default list of tests that are excluded from a test run using this configuration.

   - <u>Keywords</u> - (Optional) Use this tab to edit the default keywords and expressions used in this configuration to run tests.

   - <u>Prior Status</u> - Use this tab to edit the default restrictions on tests that are run by using their status from the previous test run.

   - <u>Execution</u> - (Optional) Use this tab to edit the default concurrency and timeout factor values used in the configuration for running tests.

5. Click Done to save the changes and close the Template Editor.

You can use the View menu at any time to change from one mode to the next. Because there is only one template file loaded in the Template Editor at a time, when you change modes during an editing session, any changes made in one mode are automatically reflected in the other.

If a configuration is attached to the edited template, the harness opens a Template updated dialog box that enables you to review the changes and to accept, reject, or postpone having the changes propagated to the attached configuration. See <u>Resolving Configuration-Template Conflicts</u> for a description of the dialog box and the choices available to the user.

▼ Changing Tests to Run

To change the set of tests that are specified to run, perform the following steps:

1. Choose View > Quick Set Mode from the Template Editor menu bar.

2. Click the Tests tab in the Template Editor.
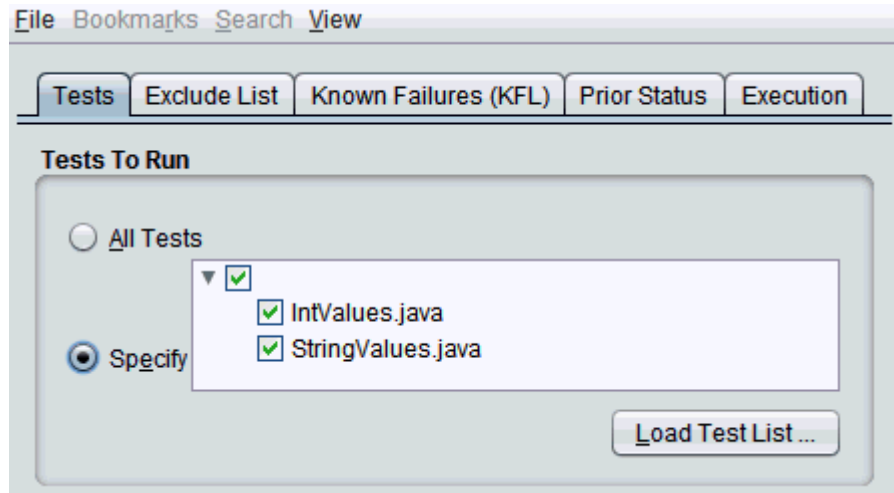
3.

4. In Quick Set Mode, choose Specify to enable the Template Editor test tree and the Load Test List button.

   In Question Mode, use the How to Specify Tests question to specify whether to use the test tree or a test list.

5. Use the test tree or a test list to specify the tests that are run.

   See the following topics for detailed information about using the test tree or load list to specify tests:

   [Specifying Tests in the Test Tree](#)
   [Loading a Test List](#)

6. Change additional template settings or click the Done button to save the changes in the template.

   If you are creating a new template for these changes, instead of clicking the Done button, choose File > Save As from the Template Editor menu bar and save the template with a relevant name.

Specifying Tests in the Test Tree

In the Template Editor test tree (not the Test Manager test tree) you can choose individual tests and folders of tests for the harness to run. The harness walks the test tree starting with the sub-branches or tests you specify and executes all tests not filtered out by the exclude list, keyword, or prior status.

Restrictions are applied cumulatively. For example, you can specify the tests in a test suite, then restrict the set of tests using an exclude list, and then further restrict the set to only those tests that passed on a prior run.

If you choose a test folder, the harness selects all tests in the test suite under that location for the test run.

If you choose one or more tests, the harness selects those individual tests for the test run.

The Tests pane highlights all folder and test icons selected for the test run. You can make individual selections in the test tree by pressing the Control key when you click an icon or name in the test tree.

To select a series or sequence of tests or folders, press the Shift key and then click the first and the last icon or name in the sequence.

When you select some (but not all) of the tests in and under a folder, the Test pane partially highlights the folder icon.

▼ Loading a Test List

To load a test list file in the template, perform the following steps:

1. Click the Load Test List button.

2. Use the dialog box to locate and select a test list file.

   See Test List File Format for a detailed description of a test list file.

3. When the file is loaded, the test tree is updated to indicate the tests that have been specified.

Test List File Format

A test list is a `.txt` file containing a list of tests to run. See [Creating a Test List File](#) for a detailed description of the test list file. The format requirements of the test list file are as follows:

- Line-oriented

- Relative test paths
- Blank lines and lines beginning with the pound symbol (#) are ignored
- On all other lines, the first item (up to the first whitespace) is taken as a test name and the remainder of the line is ignored

The format of the load list file is the same as the format of the `summary.txt` generated as part of a report.

▼ Creating a Test List File

To create a test list file, perform the following steps:

1. Select the lines from `summary.txt` that define the tests you want to run.

Note – You can use other sources for obtaining the path names of the tests that you want to include in the test list file. On a Solaris platform, you might use `awk`, `grep`, or similar utilities to identify lines that specify the tests with failed or error results.

Example lines from `summary.txt`:

1. ```
   BigNum/compareTest.html Failed. exit code 1
   BigNum/equalsTest.html Failed. exit code 1
   BigNum/longConstrTest.html Failed. exit code 1
   BigNum/subtractTest.html Failed. exit code 1
   lists/DoublyLinkedList/appendTest.html Failed. exit code 1
   lists/DoublyLinkedList/equalsTest.html Failed. exit code 1
   lists/DoublyLinkedList/insertTest.html Failed. exit code 1
   lists/DoublyLinkedList/removeTest.html Failed. exit code 1
   lists/LinkedList/appendTest.html Failed. exit code 1
   lists/LinkedList/equalsTest.html Failed. exit code 1
   lists/LinkedList/insertTest.html Failed. exit code 1
   lists/LinkedList/removeTest.html Failed. exit code 1
   lists/SortedList/equalsTest.html Failed. exit code 1
   lists/SortedList/insertTest.html Failed. exit code 1
   lists/SortedList/removeTest.html Failed. exit code 1
   ```
   Copy and paste the relative paths into a `.txt` file.

   Remove all text from the path names that is not part of the test path.

   Example lines in the `.txt` file:

   ```
   BigNum/compareTest.html
   BigNum/equalsTest.html
   BigNum/longConstrTest.html
   BigNum/subtractTest.html
   lists/DoublyLinkedList/appendTest.html
   ```

```
lists/DoublyLinkedList/equalsTest.html
lists/DoublyLinkedList/insertTest.html
lists/DoublyLinkedList/removeTest.html
lists/LinkedList/appendTest.html
lists/LinkedList/equalsTest.html
lists/LinkedList/insertTest.html
lists/LinkedList/removeTest.html
lists/SortedList/equalsTest.html
lists/SortedList/insertTest.html
lists/SortedList/removeTest.html
```

2. Save the `.txt` file using a descriptive name.

   Example `.txt` file name:

   `BigNum_ListsfailedTests.txt`

▼ Adding or Removing Exclude Lists

To add or remove exclude list files in the template, perform the following steps:

1. Choose View > Quick Set Mode from the Template Editor menu bar.

2. Click the Exclude List tab in the Template Editor.

3.
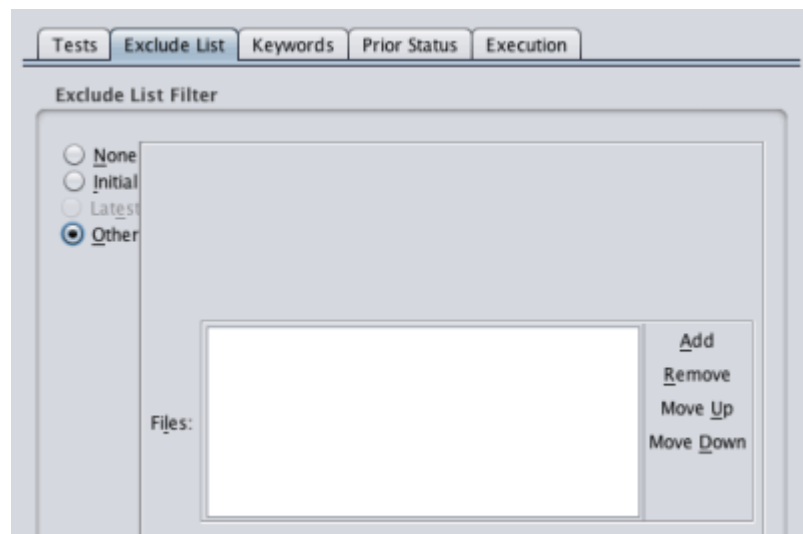
4. Add or remove exclude lists in the template as required.

Note – Exclude list files contain a list of tests in a test suite that are not run by the harness. See Exclude List Pane for a description of this pane.

5. Change additional template settings or click the Done button to save the changes in the template.

   If you are creating a new template for these changes, instead of clicking the Done button, choose File > Save As from the Template Editor menu bar and save the template with a relevant name.

Exclude List Pane

Use the following to change the exclude list used to run tests:

- **None** - An Exclude List is not used.
- **Initial** - Only enabled if the test suite provides an exclude list. If you choose Initial, the tests are run using the exclude list provided by the test suite.
- **Latest** - Only enabled if the test suite provides a location for updated exclude lists. If you choose Latest, additional options are displayed. See Latest Exclude List below for detailed information.
- **Other** - A custom exclude list can be used. See Other Exclude List below for detailed information.

If a complete test is added to the exclude list, the harness updates the test result status without requiring the test be rerun. However, if only a test case from a test is added to the exclude list, the harness requires that you rerun the test using the updated exclude list before updating the test result status.

Latest Exclude List

If your test suite provides a URL for the latest test suite, the harness enables the Latest exclude list option. The following table describes the text and controls displayed when you choose the Latest exclude list option.

TABLE 39   Latest Exclude List Dialog Box Contents

Text and Controls	Description
Location	Displays the location of the exclude list specified by the test suite. This is a non-editable field.
Last updated	Displays the date that the exclude list was last updated. This is a non-editable field.
Check For Updates Automatically	Causes the harness to automatically check the location of the exclude list and compare the date-time stamps of the remote and local exclude lists. The harness then displays a dialog box advising you of the results. If a new exclude list is available, you can choose to download it.
Every _ Days	Sets an interval for the harness to automatically check the remote location of the exclude list for updates.
Every Test Run	Causes the harness to check the remote location of the exclude list for updates before each test run.
Check Now	Causes the harness to check the remote location of the exclude list for an update.

Other Exclude List

The following table describes the buttons displayed when you choose the Other exclude list option.

TABLE 40   Other Exclude List Dialog Box Contents

Button	Description
Add	Selects an exclude list file for your test suite. As you make selections with the file chooser dialog box, they are added to the list. After you add an exclude list, you can modify the list.
Remove	Clears an item from the list. Select an item in the list and click Remove.
Move Up	Moves an item one position higher in the list. Select an item in the list and click Move Up.
Move Down	Moves an item one position lower in the list. Select an item in the list and click Move Down.

▼ Editing Keywords and Expressions

To edit keywords and keyword expressions in the template, perform the following steps:

1. Choose View > Quick Set Mode from the Template Editor menu bar.

2. Click the Keywords tab in the Template Editor.

   The JT harness displays this tab only if your test suite uses keywords.
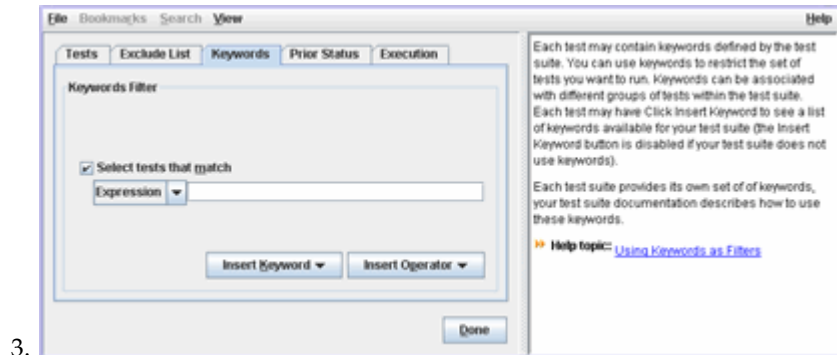
3. Use the Expression, Insert Operator, and Insert Keyword buttons to build an expression in the text field by using any logical combination of the following actions:

   Click the Expression button to display a list of expressions that can be constructed. From the list, choose the type of expression that you are building.

   Click the Insert Keyword button to display the list of keywords provided by the test suite for use in filtering tests (this is only available if the test suite has information). From the list, choose the keywords used in the expression.

   Click the Insert Operator button to display a list of operators that you can use to construct boolean expressions in the text field. From the list, choose the operator used in the expression.

   See Keyword Expressions for descriptions and examples of keyword expressions that can be constructed.

   Change additional template settings or click the Done button to save the changes in the template.

Keyword Expressions

The following table provides descriptions and examples of keyword expressions that can be constructed.

TABLE 41 Keyword Expressions

Expression	Description
Any Of	Runs all tests in the test suite having any of the keywords entered in the text field.
	Example:
	A test suite uses the keyword `interactive` to identify tests that require human interaction, and `color` to identify tests that require a color display.
	To execute only the tests containing the `interactive` keyword, choose Any Of and then use the Insert Keyword button to choose the `interactive` keyword.
All Of	Runs all tests in the test suite having all of the keywords entered in the text field.
	Example:
	To execute only the tests containing both the `interactive` and `color` keywords, choose All Of and then use the Insert Keyword button to choose the `interactive` and `color` keyword.
Expression	Runs all tests in the test suite having the expression entered in the text field.
	Use the Insert Keyword and the Insert Operator buttons to construct a Boolean expression in the text field. Keywords stand as Boolean predicates that are true if, and only if, the keyword is present in the test being considered. A test is accepted if the overall value of the expression is true. All other tests are rejected by the restriction.
	Example:
	A test suite uses the keyword `interactive` to identify tests that require human interaction, and `color` to identify tests that require a color display.
	To execute only the tests with the `color` keyword that do not also contain the `interactive` keyword, choose Expression and then use the Insert Keyword button to choose the `color` keyword, the Insert Operator button to choose the ! operator, and the Insert Keyword button to choose the `interactive` keyword.

List of Logical Operators

Logical operators are only available when Expression is selected in the list of expressions. The following table provides descriptions and examples of logical operators that can be used to build keyword expressions. The precedence column

indicates the order in which the expression is resolved. Expressions in parentheses are evaluated first, with nested parentheses being evaluated from the innermost parentheses outward.

TABLE 42 Keyword Operators

Logical Operator	Precedence	Description
( ) group	1	Used to create groups of expressions.
		Example:
		A test suite uses the keyword `interactive` to identify tests that require human interaction and `color` to identify tests that require a color display.
		`!(interactive&color)`
		The harness will exclude tests that include both keywords.
! not	2	Logical not. Used to exclude tests containing the expression.
		Example:
		A test suite uses the keyword `interactive` to identify tests that require human interaction and `color` to identify tests that require a color display.
		`!interactive&!color`
		The harness will exclude tests that include either keyword.
& and	3	Logical and. Used to combine expressions.
		Example:
		A test suite uses the keyword `interactive` to identify tests that require human interaction and `color` to identify tests that require a color display.
		`interactive&`color
		The harness will only choose tests that include both keywords.
| or	4	Logical or. Used to specify either of two expressions. Example:
		A test suite uses the keyword `interactive` to identify tests that require human interaction and `color` to identify tests that require a color display.
		`interactive|`color
		The harness will only choose tests that include either keyword.

▼ Editing Prior Status Settings

To change the prior status settings in the template, perform the following steps:

1. Choose View > Quick Set Mode from the Template Editor menu bar.

2. Click the Prior Status tab in the Template Editor.

Tests | Exclude List | Keywords | Prior Status | Execution

Prior Status Filter

☑ Select tests that match
Any Of: ☐ Passed ☑ Failed ☐ Error ☑ Not Run

3.

4. Change the prior status settings in the template as required.

   See [Prior Status Pane](#) for a description of this pane.

5. Change additional template settings or click the Done button to save the changes in the template.

   If you are creating a new template for these changes, instead of clicking the Done button, choose File > Save As from the Template Editor menu bar and save the template with a relevant name.

Prior Status Pane

By choosing Select tests that match, you can run tests with restrictions based on their result from a prior test run. The following table describes the available Prior Status filter selections.

TABLE 43   Using Prior Status as a Filter

Prior Status	Action
Passed	Selects tests that passed the last time the test was executed.
Failed	Selects tests that failed the last time the test was executed.
Error	Selects tests that the harness could not execute the last time it was included in a test run.
Not Run	Selects tests without results in the current work directory.

Prior status is evaluated on a test-by-test basis using information stored in the result files (`.jtr`) that are written in the work directory. Unless overridden by a test suite, a result file is written in the work directory for every test that is executed. When users change work directories between test runs, the result files in the previous work directory are no longer used and, if the new work directory is empty, the harness behaves as though each test in the test suite was not run.

▼ Editing Concurrency and Timeout Factor

To change the concurrency and the timeout factor in the template, perform the following steps:

Choose View > Quick Set Mode from the Template Editor menu bar.

Click the Execution tab in the Template Editor.
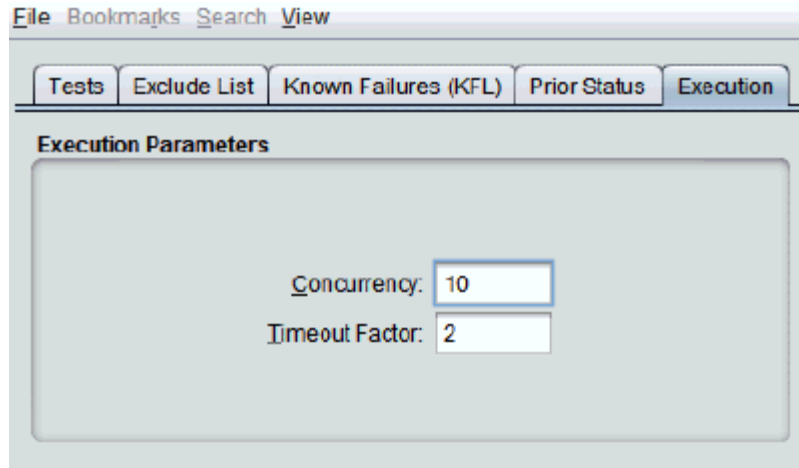
Change the values in the Concurrency and Time Factor fields as required.

See [Concurrency](#) and [Time Factor](#) for detailed descriptions of these settings.

Change additional template settings or click the Done button to save the changes in the template.

If you are creating a new template for these changes, instead of clicking the Done button, choose File > Save As from the Template Editor menu bar and save the template with a relevant name.

Concurrency

The harness can run tests concurrently. If users run tests on a multiprocessor computer or are using multiple agents on a test system, concurrency can reduce the time required to run tests. For detailed information about using agents to run tests, refer to the test suite documentation and to the *JavaTest Harness Agent User's Guide* if it is provided by the test suite.

When using multiple agents to run tests, the concurrency value must not exceed the number of agents. If the concurrency value exceeds the total number of available agents, an error will occur in the test run.

The default range of values used by the harness is from `1` to `50`. If unexpected test failures occur, run the tests again, one at a time. Some test suites may not work correctly if tests run concurrently.

For the first test run, set the value in this field to `1`. After the tests run properly, incrementally increase this value. Unless the test suite restricts concurrency, the maximum *number* of threads specified by the `concurrency` command is `50`. See the test suite documentation for additional information about using concurrency values greater than `1`.

This field is disabled for some test suites.

Time Factor

To prevent a stalled test from stopping a test run, most test suites set a timeout limit for each test. The timeout limit is the amount of time that the harness waits for a test to complete before moving on to the next test.

If users run tests on a slow CPU or slow network, change the time limit by specifying a floating point value in the time factor field. Each test's timeout limit is multiplied by the time factor value. The default range of values used by the harness is from `0.1` to `100.0`.

Note – In the Time Factor field, the harness uses the form of floating point values that is specific to the locale in which it is run. For example, if your locale uses floating point values in the form of `x,x`, harness uses that form of floating point value in the Time Factor field. In setting the timeout factor in the following example, specify values of `2,0` and `0,5` if your locale uses floating point values in the form of `x,x`.

Example:

If you specify a value of `2.0`, the timeout limit for tests with a basic 10-minute time limit becomes 20 minutes. Specifying a value of `0.5` for tests with a 10-minute limit produces a 5-minute timeout limit.

At first, use the default value of `1.0` to run tests and then, if necessary, increase the value. The actual timeout calculation for any particular test suite might vary.

Resolving Configuration-Template Conflicts

When the harness detects a conflict between a configuration and its template, it displays a Template update conflict dialog box.
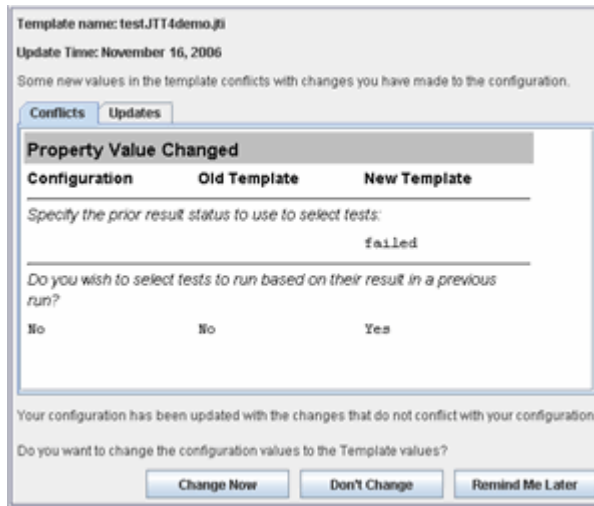
By default, any change to a template is considered a Conflict and is listed in the Conflicts tab. Configuration users get to choose whether to accept the new template value, reject the value, or postpone deciding.

A test suite can also declare a template question to be auto-updating. If a user changes an auto-updating value in the template, the harness replaces the corresponding property in the attached configuration. The updated value is listed on the Updates tab.
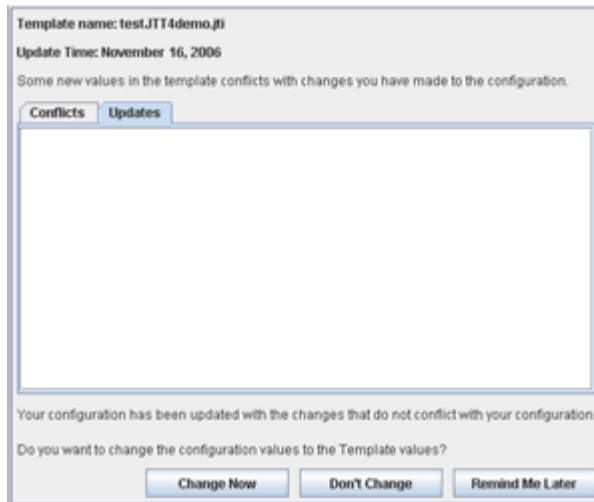
A user cannot refuse to accept an auto-updating value.

Choose a dialog button based on the following guidelines:

- **Change Now** - Click this button if you want the new template values to overwrite the corresponding configuration values.
- **Don't Change** - Click this button if you want the new template values to be ignored.
- **Remind Me Later** - Click this button if you want to defer the conflict resolution until the next time the harness checks for updates.

Setting Bookmarks

You can use Bookmarks to restrict the set of questions displayed when a user loads the template in a Configuration Editor. This feature enables you to reduce the visible length and complexity of a configuration interview by only displaying the questions that a user must answer before running tests.

Bookmarks are persistent and are saved in the `.jtm` file. If bookmarks are disabled by a user, they remain in program memory but cannot be manipulated.

▼ Set Bookmarks for Specific Questions

To set bookmarks for specific questions in the template, perform the following steps:

1. Click the Bookmarks **>** Enable Bookmarks check box in the menu bar.

2. Click a question in the Template Editor.

3. Set a bookmark for the highlighted question by performing one of the following actions:
   - Click to the left of the highlighted question text.
   - Choose Bookmarks **>** Mark Current Question in the menu bar.
   - Right click the highlighted question and choose the Mark Current Question menu item in the pop-up menu.

▼ Display Questions With Bookmarks

To only display the questions with bookmarks, perform the following step:

- Click the Bookmarks > Show Only Bookmarked Questions checkbox in the menu bar.

  The Template Editor displays bookmarked questions in the following manner:

- The first question in the interview is displayed.
- Questions with bookmarks are displayed.
- If the interview is complete (all questions have valid answers) the final question is displayed.
- If the interview is incomplete (one or more questions have invalid or incomplete answers), the questions from the last marked question to the first question with an invalid or incomplete answer are displayed.
- Sequences of questions not described by a previous category are grouped and represented by three dashes (---) in the list of questions. These groups can be opened to display the complete sequence of questions.

▼ Open Groups of Questions

Groups of hidden questions without bookmarks are represented by three dashes (---) in the list of questions.

To open a group of hidden questions, perform the following steps:

1. Click the --- section in the list.

2. Click the Bookmarks > Open --- Group menu item from the menu bar or right click the highlighted --- section and choose Open --- Group menu item from the pop-up menu.

▼ Close Groups of Questions

To close groups of questions without bookmarks, perform the following steps:

1. Click the question in the list.

2. Click the Bookmarks > Close --- Group menu item from the menu bar or right click the highlighted --- section and choose Close --- Group menu item from the pop-up menu.

▼ Clear the Values of All Questions With Bookmarks

To clear the values of all questions with bookmarks, perform the following steps:

1. Click a question in the question list.

2. Click the Bookmarks > Clear Answers to Bookmarked Questions menu item from the menu bar.

▼ Clear the Value of a Specific Question

To clear the value of a specific question, perform the following steps:

1. Click the question in the list.

2. Click the Bookmarks > Clear Answer For Current Question menu item or right click the highlighted question and choose Clear Answer For Current Question from the pop-up menu.

▼ Remove All Bookmarks

To remove the bookmarks from the template, perform the following step:

- Choose Bookmarks > Remove Bookmarks.

  The list of questions reverts to the full list.

▼ Remove a Bookmark From a Question

To remove the bookmarks from a question in the template, perform the following steps:

1. Click the question in the question list.

2. Click the Bookmarks > Unmark Current Question menu item or right click the highlighted question and choose Unmark Current Question menu item from the pop-up menu.

Using Template Search

When using the Template Editor in Question Mode, you can locate and display the panes containing a specific character string by choosing Search > Find. The Find Question dialog box is used in locating and displaying a specified character string.

The editor can search titles, questions, and answers for matching characters. It does not search the More Info.
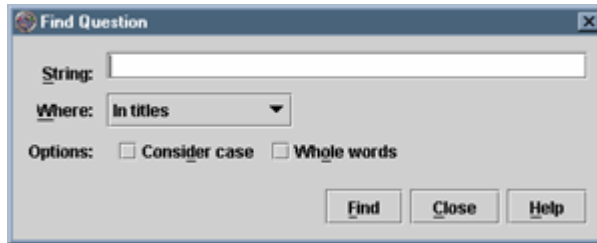
The following table describes the search criteria used in the Find Question dialog box.

TABLE 44   Find Question Dialog Box Search Criteria

Item	Description
String	Enter the character string that you are trying to find.
Where	Choose where you want to search:
	In titles
	In text of questions
	In answers
	Anywhere
Options: Consider case	
	Specifies that the search pattern match the case of the characters in the Find text field.
Options: Whole words	
	Specifies that the search pattern only match whole words from the Find text field.

Click the Find button to search for the character string.

To repeat the search, either click the Find button again or use the Search > Find Next menu item from the Template Editor menu bar.

Click the Help button to display context-sensitive help.

▼ Loading a Template

To load a template, perform the following steps:

1. Choose Configure > Load Template from the Test Manager menu bar.

   The harness opens a Load Template dialog box containing a path field with Browse button and a navigation area that displays the file name, the template name (contained in the template), and the template description (contained in the template).

   The Path field sets the starting location for the directories and files displayed in the navigation area.

   The navigation area displays the directories and template files (if any) in the location set in the Path field. Only template files are displayed in the navigator area. Other files are hidden.
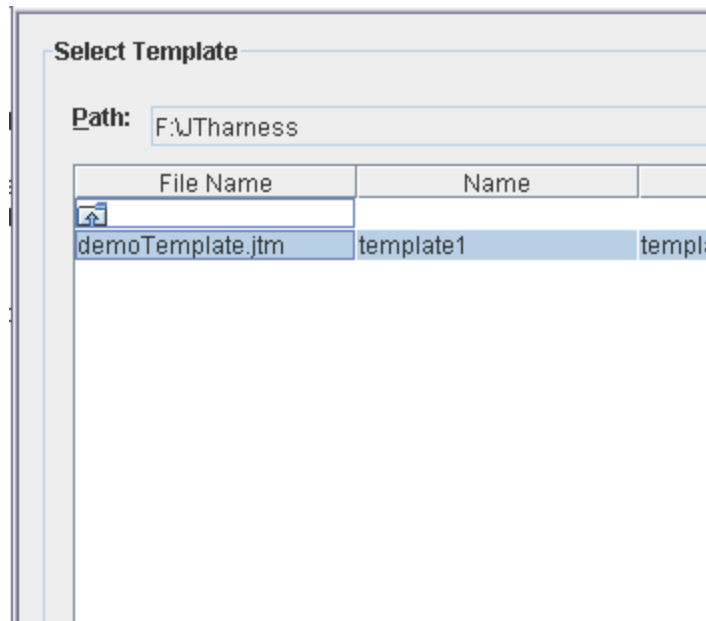
2. Use the icons in the dialog box to navigate to the location of the template.

   The Path text field in the dialog box displays the location for the list of files displayed in the dialog box.

3. Click the template icon and the Load button.

   The harness loads the template. If the template is not correct for the test suite, the harness displays an error dialog box and closes without loading the template.

7

Running Tests

In the Test Manager, you can use the Run Tests menu or the toolbar button to start a test run of the tests specified in your current configuration. See <u>Test Manager Tool</u> for a description of the Run Tests menu and the tool bar buttons. See <u>Configuring a Test Run</u> for a description of how to use the Configuration Editor to specify configuration information used to run tests.

You can also use the test tree pop-up menu to start a test run of one or more tests selected from the test tree. See <u>Test Tree Pop-up Menu</u> for a description of how to use the test tree pop-up menu to run individual tests in a test tree.

Note – If you do not want to use the Test Manager to run tests, you can use the command line. See <u>Command-Line Summary</u> in the *Command-Line Interface User's Guide* for information about running tests from the command line.

The harness saves all test results after running tests but does not automatically generate reports of test results. You must generate test reports from the Test Manager or from the command line. See <u>Generating and Viewing Reports</u> for detailed information about reports.

If you use the All Tests view filter and begin a test run, the harness displays an advisory dialog box. Using the All Test view filter displays the results of all tests in a test suite regardless of whether or not they are included in the test run.

You can choose to disable the dialog box by setting harness Preferences, by choosing a different view filter, or by using the check box in the dialog to stop the dialog from being displayed in the future.

This chapter contains the following topics, presented in a sequence that you can use when running tests:

- **Starting a Test Run** - Describes how to use the Test Manager to start a test run using the current configuration.

- **Monitoring a Test Run** - Describes how to use the Test Manager (including the use of the Log Viewer) to monitor a test run.

- **Monitoring Agents** - Describes how to use the Agent Monitor tool to monitor agent activity during a test run.
- **Stopping a Test Run** - Describes how to use the Test Manager to stop a test run.
- **Troubleshooting a Test Run** - Describes how to use the Test Manager to troubleshoot a test run.

Starting a Test Run

When the harness is not running tests, it enables both the ⏵ button on the tool bar and the Run Tests > Start menu item.

Note – Only one test run at a time can be active in each Test Manager.

To start a test run using the current configuration, either click the ⏵ button or choose Run Tests > Start. You can also use the test tree pop-up menu to run a specific test or group of tests in a folder. See <u>Test Tree Pop-Up Menu</u>.

Before the harness attempts to run the test suite, it verifies that the required configuration information is complete. You can view the configuration state in the Test Manager Properties browser. See <u>Viewing Test Manager Properties</u> for a description of the browser.

If the configuration information is not provided or is incomplete, the harness opens a dialog box advising you that the configuration must be completed before it can begin running tests. You can choose to open the configuration editor window or cancel the test run. If you choose to open the configuration, the harness opens the configuration at the incomplete section.

To change the test suite or work directory before running tests, refer to the following topics:

- <u>Opening a Test Suite</u>
- <u>Opening a Work Directory</u>

- [Creating a Work Directory](#)

If a JavaTest Harness agent is used to run the tests for your product, you must start the agent before you begin the test run. See *JavaTest Harness Agent User's Guide* for detailed information about the JavaTest Harness agent.

If the harness starts the test run and issues a request before the active agent starts running, the harness waits for an available agent until the timeout period ends. If the timeout period ends before an agent is available, the harness reports an error for the test.

Monitoring a Test Run

After the test run begins, the Test Manager displays information about the test run in the areas described in the following table.

TABLE 45   Monitor a Test Run

Area	Description
Test Tree	The test tree uses colored icons to display the current run and test results status of the folders and tests in the work directory. As the harness completes running individual tests, it updates each test tree icon to indicate the test result status. See Monitoring With a Test Tree for detailed information about using the test tree to monitor the progress of the test run.
Status Messages	Below the test tree area is a resizable text area that displays information about current Test Manager activity. In this area the harness displays status messages about the state of the test run and the name of the test being run.
Progress Indicator	The Test Manager Status Line contains a progress indicator. The indicator displays the elapsed time of the previous test run when tests are not running. When tests are running, it automatically changes to the progress bar of the current test run. At the completion of the test run, the indicator changes to display the elapsed time. See Progress Indicator for information about using the test progress meter to monitor the progress of the test run.
Progress Monitor	A separate Progress Monitor is available that displays current, detailed information about the progress of the test run. See Progress Monitor for information about using the Progress Monitor.
Information Area	As tests run, the harness displays information about the run in the information area to the right of the test tree. The information area provides two views: Folder view - When you click a folder icon in the test tree, the harness displays a Summary tab, a Documentation tab, five status tabs, and a status field containing information from the work directory about a folder and its descendants. See Displaying Folder Information for detailed information about browsing folder information. Test view - When you click a test icon in the test tree or double click its name in the Folder view, the harness displays six tabbed panes that contain detailed information about the test. See Displaying Test Information for detailed information about browsing test information.

While monitoring the test run in the Test manager, you can also display the contents of output logs being generated during the test run. See Monitoring

Output Logs and Monitoring Services for detailed information.

The harness also provides a web server that you can use to remotely monitor and control batch mode test runs. The HTTP Server provides two types of output:

HTML Formatted Output allowing users to remotely monitor batch mode test runs in a web browser

Plain Text Output intended for use by automated testing frameworks

See the *Command-Line Interface User's Guide* for details about running tests from the command line and using the web server.

Monitoring With a Test Tree

The test tree uses folder icons, test icons, and two independent types of filtering (run filtering and view filtering) to simultaneously display the following conditions:

- Progress of a test run
- Current test results in the work directory

When a test run begins, you can track its progress in the test tree by observing the folder and test icons. The test tree displays an arrow at the left of each folder and test icon as it is loaded and run.

After the harness completes running a test, it writes the test results to the work directory and updates the folder and test icons in the test tree. The test tree displays folder and test icons based on the view filter specified in the Test Manager window.

The harness supports using status colors specified by the user instead of the default color settings. See Specifying Status Colors in the *Command-Line Interface User's Guide*.

Regardless of whether or not a test was run, the test tree displays filtered out

folder and test icons for those tests and folders filtered out by the view filter. All other icons are updated to reflect their current result status from the work directory.

Changing either the run or the view filter settings causes the harness to immediately update the folder and test icons displayed in the test tree.

See View Filters for a description of how to specify which test results from the work directory are displayed in the test tree.

See Test Tree for a detailed description of the icons, filters, and other features used in the test tree.

The goal of a test run, when using the appropriate view filter, is for the root test

suite folder to display the passed folder icon. The passed root test suite folder icon signifies that all tests in the test suite not filtered out of the test run (by

specifying tests that are run, exclude lists, keywords, and prior status) with passed test results. See Changing QuickSet Values for a description how to quickly set run filters that include or remove tests from a test run.

Click the test suite icon in the test tree to display status information for the test suite in the Test Manager information area. The view filter used in the test tree is also used to display folder status information. See

Using View Filters

In the toolbar at the top of the test tree is a list of view filters that you can use to select the status of the folders and tests displayed by the test tree. The harness includes a Current Configuration filter and an All Tests filter. Additional filters, such as certification filters, can also be added by the test suite. Refer to your test suite documentation for detailed descriptions of any additional filters in the list of view filters.

Selecting a view filter only filters the status (folder colors and counters) displayed in the Test Manager window, not the tests that are run. You must use the configuration editor to filter the tests that are run. See Specifying Tests to Run.

See View Filters for detailed information about the view filters available in the Test Manager.

Monitoring Output Logs

You can use the Log Viewer to monitor logs produced during a test run. To use the Log Viewer, your test suite must generate the logs that the viewer can display. See your test suite documentation for detailed information about the log files that it generates. See Log Viewer for a detailed description of the Log Viewer.

The Log Viewer enables you to perform the following actions when monitoring output logs:

- Display specific types of information from a log by selecting one or more notification levels.

  For example, you can choose to display only the Critical and Warning messages.

- Monitor real-time content of output logs.

  You can track log data in real time by selecting the Live Scrolling check box.

- Search log messages for text strings.

  You can use the Find text field to search for literal strings within the currently viewed log data.

- Display multiple views of real-time content.

  You can display data side-by-side by clicking the Open New Log Viewer button and using different criteria for displaying the output data in each viewer. Only the first instance of the Log Viewer is used to restore log viewing preferences next time the work directory is opened.

- Selection of log levels across all available logs in a single action.

  Use the Selection Actions list to turn on a log level in all available logs. For example, you can turn on Error reporting for all logs. Because the operation only affects the current logs, logs created after you make the selection are not affected. To include logs created after the initial selection, reselect the action.

- Save log contents displayed in the log viewer. To save the log contents, click the Save button at the bottom of the viewer.

- Clear the contents displayed in the log viewer. To clear the log contents, click the Clear log button at the bottom of the viewer.

Agent Monitor Tool

If your test suite uses the harness agent, use the Agent Monitor tool to view all agents in a test system that are running tests. Choose Tool > Agent Monitor from the menu bar to open the tool window.

The Agent Monitor window contains two sections: Agent Pool and Agents Currently In Use.
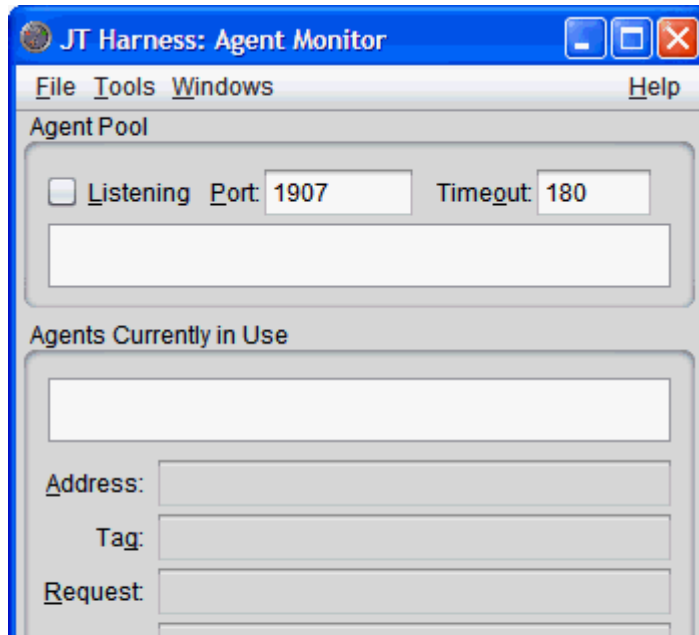
If your test suite uses the harness agent, see the *JavaTest Harness Agent User's Guide* for a detailed description of using the Agent Monitor tool.

Stopping a Test Run

When the harness is running tests, it enables both the ▣ button on the toolbar and the Stop menu item.

Either click the ▣ button or choose Run Tests > Stop to stop a test run.

As it completes each test, the harness writes the test results (`.jtr` files) in the work directory. Stopping a test run causes the tests in progress to indicate an error.

When you stop a test run, the harness does not generate reports of test results. You must generate the reports from the Test Manager window or from the command line. See Creating and Displaying Reports for detailed information about test reports.

Troubleshooting a Test Run

Normally, the goal of a test run is for all tests in the test suite that are not filtered out by the current configuration to have passing results. See Changing Configuration Values for a description of how the current configuration filters tests in a test run.

If the root test suite folder contains tests with errors or failing results, you must troubleshoot and correct the cause to satisfactorily complete the test run.

Tests with errors ▣ are tests that could not be executed by the JT harness. These errors usually occur because the test environment is not properly configured or the software under test is defective. See Tests with Errors for a detailed description of troubleshooting tests with errors.

Tests that failed ▣ are tests that were executed but had failing results. See Tests that Fail for a detailed description of troubleshooting tests that failed.

The Test Manager window provides you with the following facilities to effectively troubleshoot a test run:

- Test Tree
- Folder View
- Test View
- Log Viewer
- Service Monitor

Test Tree

Use the test tree and view filters to identify specific folders and tests with errors or failing results. Open the red and blue folders until the specific tests that failed or had errors are displayed.

Folder View

When you click a folder icon in the test tree pane, the JT harness displays a filtered summary of its test status in the Test Manager information area that matches the test tree.

Note – The View filter chosen in the Test Manager window might change the summary values displayed in the folder view, but does not change the test results written in the work directory.

Click the Error and the Failed tabs to display the lists of all tests in and under a folder that were not successfully run. You can double-click a test in the lists to view its detailed test information. Refer to Test View below for a description of the test information that the JT harness displays.

Test View

When you click a test icon in the test tree or double-click its name in the folder view, the JT harness displays unfiltered, detailed information about the test in the information area. The Test Manager displays the current information for that test from the work directory.

Note – Because the Test Manager does not use a view filter when displaying test information, the test status displayed in the information area may not match the filtered view in the test tree or the folder Summary view.

The test view contains detailed test information panes and a brief status message at the bottom identifying the type of result. This message may be sufficient for you to identify the cause of an error or failure.

If you need additional information to identify the cause of the error or failure, use the following panes listed in order of their importance:

- Test Run Messages contains a Message list and a Message pane that display the messages produced during the test run

- Test Run Details contains a two-column table of name-value pairs recorded when the test was run.
- Configuration contains a two-column table of the name-value pairs derived from the configuration data that were actually used to run the test.

Log Viewer

Use the Log Viewer (select View > Logs) to monitor log files generated during a test run or to inspect log files after a test run is completed. See Monitoring Output Logs for a detailed description.

Service Monitor

If a test suite uses services, use the Service Monitor (select View > Services) to see the status of services. See Monitoring Services for a detailed description.

8

Browsing Test Information

You can quickly browse test information in the Test Manager window by clicking folder and test icons in the test tree.
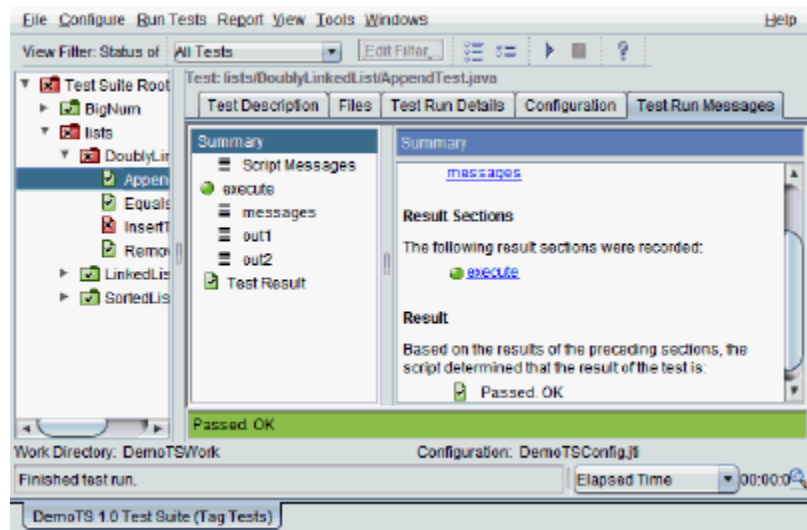
This chapter is divided into the following topics:

- Test Tree - Describes the test tree and filtering used to display the test suite, its folders, tests and status icons.
- Folder View - Describes how to display filtered folder information in the Test Manager window.
- Test View - Describes how to display unfiltered test information in the Test Manager window.
- Viewing Test Manager Properties - Describes how to display the Test Manager Properties browser.

- [Viewing Test Suite Errors](#) - Describes how to display the Test Suite Errors dialog box.

Displaying Folder Information

The folder view contains filtered summary and status information about the tests in a test folder. This information is displayed both as values and as a pie chart. The folder view and test tree use the same view filter when displaying information.

To display the folder view, click a folder icon in the test tree. The folder view displays a Summary tab, a Documentation tab, five status tabs, and a status display.
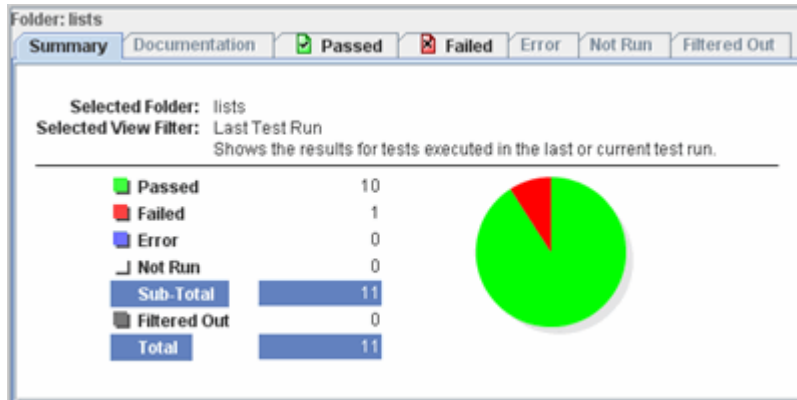
During a test run, you can use the folder view to monitor the status of a folder and its tests. You can also use the folder view during troubleshooting to quickly locate and open individual tests that had errors or failed during the test run. When a status pane is empty, the harness disables its tab.

See [Summary Information](#) for a description of the information displayed in the Summary pane.

The Documentation tab displays test suite documentation, assuming the test suite developer has provided it. In the test tree, select a test (not a folder) to display its description, if any, in the Documentation tab.

See [Status Information](#) for a description of the folder information displayed by clicking the following status tabs:

- [icon]Passed (green with a check mark)

- ■ Failed (red with an x)

- ■ Error (blue with an exclamation point)

- ■ Not Run (white with a dash)

- ■ Filtered Out (gray or shaded)

The status message displayed at the bottom of the pane provides information about the selected tab. The messages indicate that tests in the folder are loading or provide detailed status information about a selected test.

Summary Information

When you click a folder icon in the test tree, the harness uses the current view filter to display Summary information about the folder's test results.

The Summary pane contains header information that identifies the folder and the view filter presented in the Summary table and its associated pie chart. The pie chart displayed in the Summary pane is a graphical representation of the tabular data.

The following table describes the filtered work directory information that the Summary table and pie chart can display.

TABLE 46 Summary Pane Contents

Field	Description
Passed	The number of tests in a folder (including all of its subordinate folders) displayed in the test tree that were run and have passed test results.
Failed	The number of tests in a folder (including all of its subordinate folders) displayed in the test tree that were run and have failed test results.
Error	The number of tests in a folder (including all of its subordinate folders) displayed in the test tree that were run but had errors.
Not Run	The number of tests in a folder (including all of its subordinate folders) in the test tree that have not yet been run and were not filtered out.
Sub-Total	The total number of tests that were selected to run.
Filtered Out	The total number of tests in a folder (including all of its subordinate folders) displayed in the test tree that were filtered out. Tests that were filtered out include tests that you omitted from the test run by using keywords, prior status, or exclude lists.
Total	Total number of tests in a folder and its subordinate folders.

When using the All Tests view filter, new settings in the current configuration do not change the values displayed in the Summary pane.

When using the Current Configuration view filter, each time you make a change in the configuration, the values displayed in the Summary pane are recalculated and displayed based on the new settings in the current configuration. See the following example for a description of the use of the Current Configuration view filter and the All Tests view filter.

Example:

If you rerun a set of tests and use the Current Configuration view filter with Prior Status: Failed set in the configuration (see Specifying Prior Status), as actual test status in the work directory changes from Failed to either Passed or Error, the test status displayed in the Summary pane changes from Failed to Filtered Out.

If you change the view filter to All Tests, the test tree and Summary pane immediately display the actual results of all tests in the work directory, regardless of the Prior Status settings in the configuration.

See Problems Viewing Test Results for additional information about viewing test status in the test tree and Summary pane.

Click the appropriate status information tab to identify the individual tests in a category.

Status Information

In addition to Summary and Documentation information, the folder view contains five status tabs that use the view filter to group and list a folder's tests by their results in the work directory. See [Displaying Folder Information](#) for additional information.

In the tabbed panes, click a test in the list to display the summary message at the bottom of the pane.

By default the list has two columns, Name and Title. Click a heading to sort the list.

To display alternate information in the second column, right-click on the column heading and make a selection from the Description or Runtime context menus.
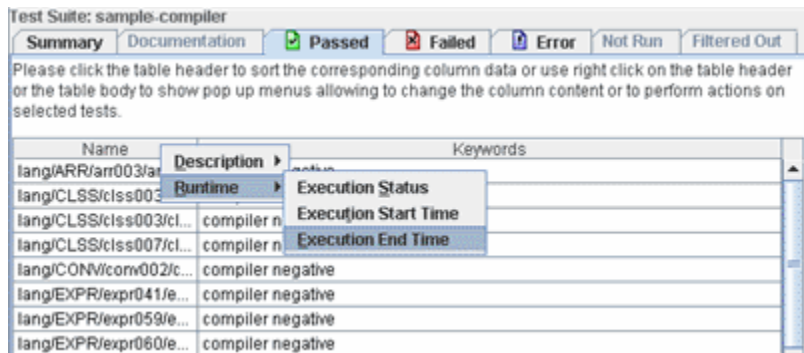
Double click the test name or press Enter to display the test in the test tree and to view its unfiltered test information. See [Displaying Test Information](#) for additional information.

The colors of the following icons are the harness default settings. The harness enables you to use colors other than these default settings. See [Specifying Status Colors](#) in the *Command-Line Interface User's Guide*.

Passed

Uses the view filter to display the test names of all tests in the folder (including all of its subordinate folders) displayed in the test tree that had passing results when they were run.

Failed

Uses the view filter to display the path names of all tests in the folder (including all of its subordinate folders) displayed in the test tree that were run and had failing results.

Error

Uses the view filter to display the path names of all tests in the folder (including all of its subordinate folders) displayed in the test tree with errors that prevented them from being executed.

Not Run

Uses the view filter to display the path names of all tests in the folder (including all of its subordinate folders) displayed in the test tree that are selected by the view filter but have not been run.

Filtered Out

Uses the view filter to display a two-column list. The first column contains the names of tests in the selected folder whose results are filtered out by the selected view filter.

The second column contains a specific reason why the test result is filtered out by the view filter. The specific reason depends on the view filter criteria. See View Filters for a detailed description of filtering criteria.

Multi-Selection Panel

When multiple tests are selected in the test tree, the harness displays a Multi-Selection panel in the Information Area.
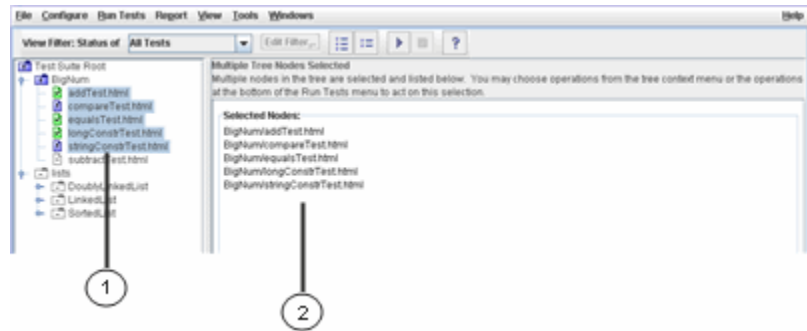
1. Test Tree Selection

2. Multi-Selection Panel

   This panel contains a text description area at the top of the panel that describes the operations that can be performed using tests and folders selected in the test tree and listed below.

   In the section below the text description area, the harness displays a list of the tests and folders selected in the test tree.

Displaying Test Information

The test view contains unfiltered, detailed information from the work directory about a specified test. To display the test view, click a test icon in the test tree or double-click a test name.

The test view contains six tabs and a colored status field (which indicates Pass or Fail).
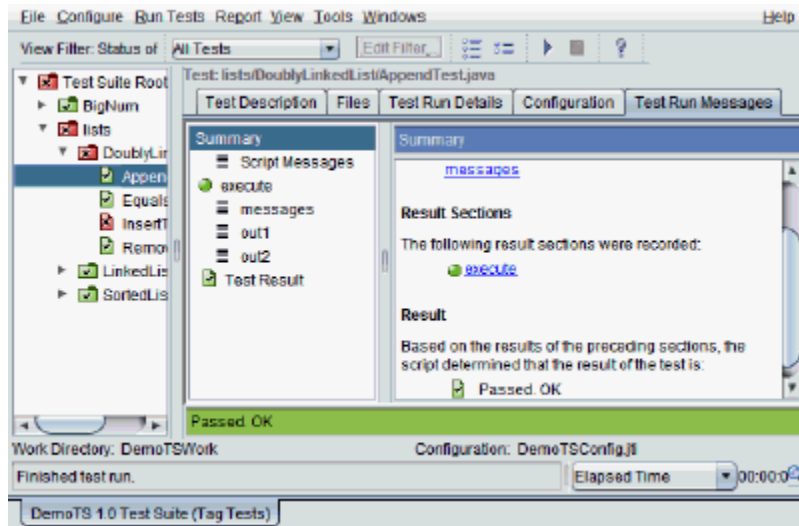
Note – The test view does not use view filtering to display information. If you are using a view filter other than All Tests, the status color displayed in this view might not match the test icon or the folder view tab.

The following table describes the contents of the test view panes. The status field at the bottom of the pane contains a description of the test result and is visible in all of the test view panes.
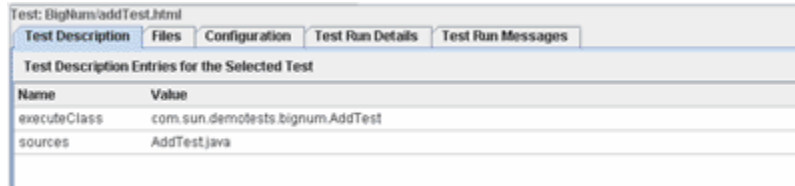
TABLE 47 Information Pane Contents

Tab	Description
Test Description Pane	Displays the name-value pairs contained in the test description. The contents are input data and always available.
Documentation Pane	Displays test documentation, if it exists.
Files Pane	Contains a drop-down list of source files from the test description. Click a file name from the drop-down list to display its contents. The contents are input data and always available.
Configuration Pane	Displays a table of configuration name-value pairs used to run a specific test. The contents are output data and only enabled if the test was run.
Test Run Details Pane	Displays the name-value pairs that were recorded when the test was run. The contents are output data and only enabled if the test was run.
Test Run Messages Pane	Contains a tree and message panel of output from sections of the test. Click a name to display its contents. The contents are output data and only enabled if the test was run.

When an information pane is empty, the harness disables it.

Test Description Pane

Click the Test Description tab to display a two-column table of name-value pairs derived from the test descriptions cached in the work directory. Each test in a test suite has a test description.

When you open the test suite in the harness, the test finder reads the test descriptions and caches them in the work directory. Test descriptions in the cache and the Test Description pane are not updated until you close and reopen the test suite. Refer to your test suite documentation for detailed descriptions of the names and values displayed in the Test Description pane.

Name

The names displayed in the table identify the attributes and properties contained in the test description.

Value

The values displayed in the table are the attribute and property values that the harness used to run the test. The values are read from the files in the test suite.

Files Pane

The Files tab displays those files nominated by a test suite on a per-test basis. The Files tab can render `.html`, `.rtf`, and `.txt` files as well as most `.au`, `.wav`, `.mid`, `.gif`, `.jpg`, and `.png` files. You can browse but not edit source files in this panel.
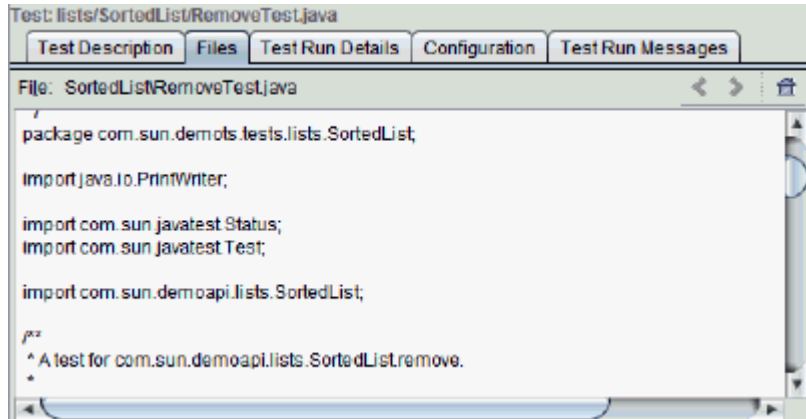
Choose a file from the drop-down list to display its contents.

Configuration Pane

The Configuration tab displays a two-column table of the name-value pairs that were derived from the configuration file and actually used to run the test. To view this information, select a test from the test tree and choose the Configuration tab.
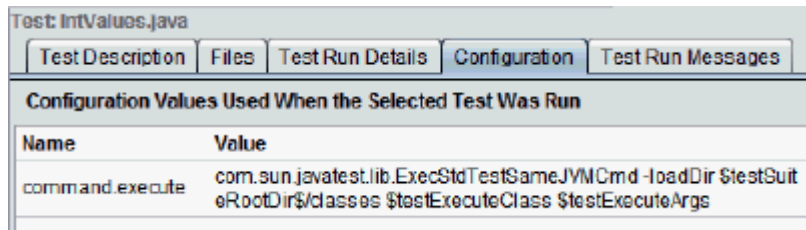
The contents of this pane vary for each test suite. Because the table contains values that were used when the test was run, it may provide valuable information when troubleshooting a test run. Refer to your test suite documentation for detailed descriptions of the name-value pairs for your test.

Name
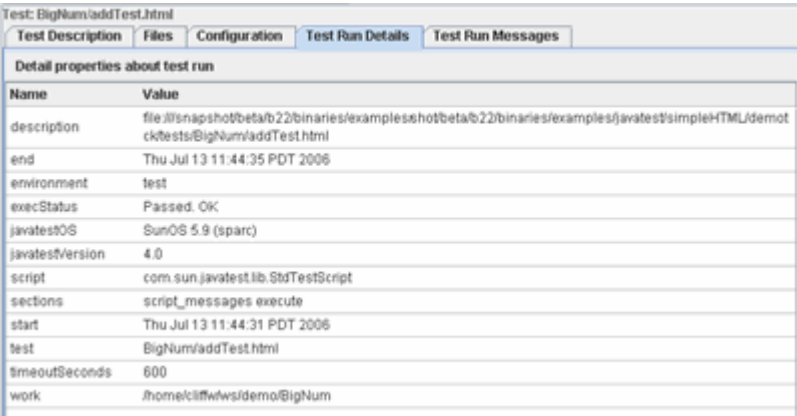
The names in the table identify test environment properties used by the harness to run the test.

Value

The values displayed in the table were used to run the test.

Test Run Details Pane

Click the Test Run Details tab to display a two-column table of name-value pairs that were recorded when the test was run and may provide valuable information when troubleshooting a test run. Refer to your test suite documentation for detailed descriptions of the result property name-value pairs for your test.

Test: BigNum/addTest.html

Test Description	Files	Configuration	**Test Run Details**	Test Run Messages

Detail properties about test run

Name	Value
description	file:///snapshot/beta/b22/binaries/examples/shot/beta/b22/binaries/examples/javatest/simpleHTML/demot ck/tests/BigNum/addTest.html
end	Thu Jul 13 11:44:35 PDT 2006
environment	test
execStatus	Passed. OK
javatestOS	SunOS 5.9 (sparc)
javatestVersion	4.0
script	com.sun.javatest.lib.StdTestScript
sections	script_messages execute
start	Thu Jul 13 11:44:31 PDT 2006
test	BigNum/addTest.html
timeoutSeconds	600
work	/home/cliffw/ws/demo/BigNum

Name

The harness derives property names from the test results file and displays them in the table with the following defaults:

- Information about the version of the harness used to run the test
- Information about the operating system used to run the test
- Date and time the test started
- Date and time the test ended

- Additional details recorded by the test script used to run the test

Because the properties listed in the table are a function of the test that you are running, the contents vary for each test suite.

Information written by commands, tests, and scripts as they are executing is displayed in the Test Run Messages pane.

Value

The values displayed in the table are from the test results file created by the harness after running the test.

Test Run Messages Pane

To display detailed messages describing what happened during the running of each section of the test, click the Test Run Messages tab. This information is useful when troubleshooting a test run.
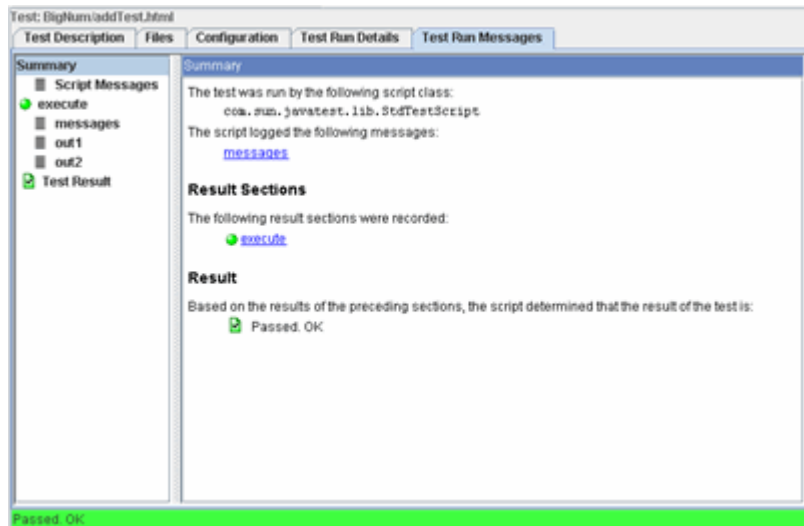
The Test Run Messages tab contains the following areas:

- **Message List** - The vertical area on the left side of the Test Run Messages tab.
- **Message Area** - The vertical area on the right side of the Test Run Messages tab.

- **Test Result Status Bar** - The horizontal area at the bottom of the Test Run Messages tab.

Message List

The message list provides a detailed list of messages issued during a test run. Click an item in the list to display its contents in the message area. The message list contains links to the following types of messages:

- [Summary Message](#)
- [Output Summary and Result Messages](#)
- [Test Result Message](#)

Summary Message

Only one per test, this message summarizes all of the messages generated during a test run and provides hypertext links to their detailed contents. The Summary Message contains the following information in the message area:

- Test script used to run the test
- Messages logged by the test script
- Individual test result sections
- Test result and its result icon

Script Messages

This message is passed up from the script that executed the test. There is only one script message per test. Script messages vary for each test script. Refer to your test suite documentation for detailed descriptions of its script messages when troubleshooting a test run.

Output Summary and Result Messages

Each test result section has an Output Summary and Result message that provides summary messages and hypertext links to its detailed messages. The name of the Output Summary message is a function of the test suite and varies for each test suite.

Some tests have only one result section, while others have multiple sections. Refer to your test suite documentation for detailed descriptions of the tests when troubleshooting a test run.

The following table lists and describes the message types.

TABLE 48 Output Summary and Result Messages

Message Type	Description
Output Summary	A two-column table listing the name and size of each output section. Each of the following output sections contains text generated while executing the test section:
	messages - Provides the command string used by the test script to run the test section. Unlike ref and log, the messages field always exists in a section.
	ref - The name of this message field is determined by the test and might be a name other than ref. A test can use this output stream to provide standard output information from the test section.
	log - The name of this message field is determined by the test and might be a name other than log. A test can use this output stream to provide standard error information from the test section. Many tests only use the log stream and include tracing as well as standard error information when writing to the log output.
	The contents of each output section varies from test suite to test suite. Refer to your test suite documentation for detailed descriptions of the test section messages when troubleshooting a test run.
	If no details exist in an output section, the harness does not create its hypertext link and indicates in the Size (chars) column that it is empty.
Result	Contains a colored status icon and a brief description of the results of the specific test section. The color of the circle indicates the result of the test section.

Test Result Message

The Test Result Message indicates the cumulative result of the test. There is only one Test Result Message per test.

Note – For negative tests, the Test Result correctly indicates Passed when all of its test sections have failed.

Message Area

The area displays the messages issued during a test run. The number, names, and content vary for each test suite and may also vary for different tests in the same test suite.

Test Result Status Bar

The area displays an abbreviated form of the Test Result Message. See Test Result Message for detailed information.

Displaying Configuration Information

The harness provides special browsers for displaying configuration checklists, test environment values, exclude list contents, and configuration log information derived from the current configuration.This section contains the following topics:

- Displaying the Configuration Checklist - Describes how to display the checklist that a test suite's interview may produce.

The checklist is optional and may not be produced by all test suites.

- Viewing Configuration Values - Describes how to view the configuration values used to run a test suite.

When troubleshooting you can use the Test Environment browser to view the configuration values and their sources that were derived from the configuration file and used by test suite specific plugin code to execute and run tests.

- Viewing Exclude List Contents - Describes how to view the list of tests excluded from a test run.

You can use the Exclude List browser to display the list of tests that were excluded from a test run. The browser also displays details about individual tests selected in the list.

- [Displaying the Question Log](#) - Describes how to browse (in HTML format) the text of all completed questions asked in the configuration interview and their answers.

Displaying the Configuration Checklist

Your test suite might produce a checklist of steps that must be performed before running tests. The checklist is dynamically generated by the harness from the current configuration. Changes to the values in the current configuration can produce different checklist items.

If the tests suite produces a checklist, the harness enables the Show Checklist menu item. To view the Configuration Checklist of the current configuration interview choose View > Configuration > Show Checklist.

To close both the Configuration Checklist and the viewer, click the Close button at the bottom of the window.

Viewing Configuration Values

The Test Environment browser displays the configuration values used to run a test suite.
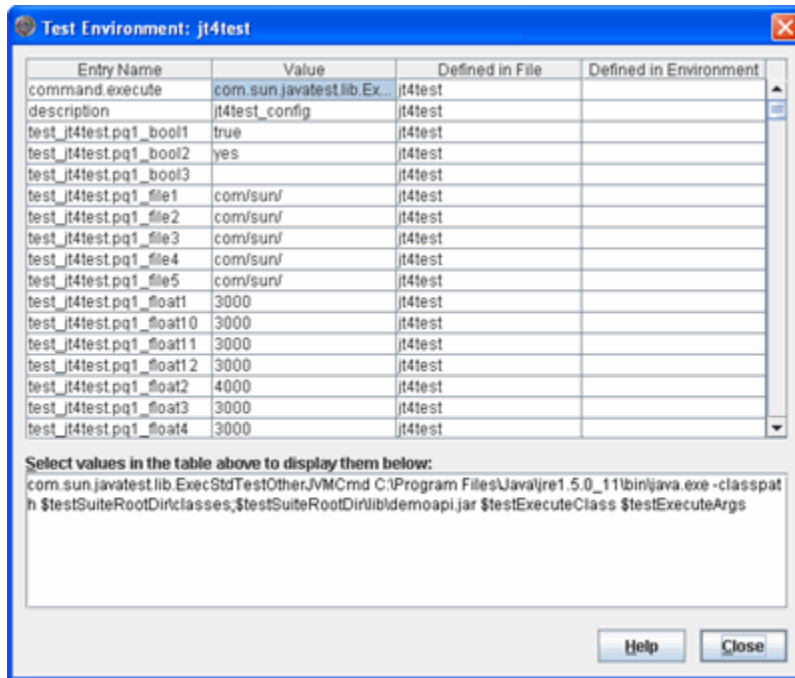
To open the Test Environment browser, choose View > Configure > Show Test Environment from the Test Manager menu bar.

The Test Environment browser contains a four-column table that displays information derived from the configuration file and used in configuring the test run. Useful features of the Test Environment browser include the following:

- Click and drag the column headers or their separators to rearrange the order and change the size of the columns.
- Click inside a table cell to display its contents in the text box below the table. This is a useful feature when the contents of a cell are too long for the table to effectively display them.

- Click a column header to alpha-numerically sort the contents of the table. This is a useful feature when searching for a specific name or value in a lengthy table.

The following table describes the contents of the Test Environment browser.

TABLE 49   Test Environment Dialog Box Contents

Name	Description
Entry Name	Identifies a name-value pair derived from the configuration file and used by test suite specific plug-in code to execute and run tests.
Value	A value specified by the user in the configuration file used to configure a test run.
Defined in file	Identifies the source of the configuration information used to run the tests. If your test suite uses a configuration (.jti) file to run tests, the source of the values is displayed.
Defined in environment	When appropriate, this field displays the environment name from the configuration.

Viewing Exclude List Contents

The Exclude List browser displays the list of tests excluded from the test run by the exclude list.
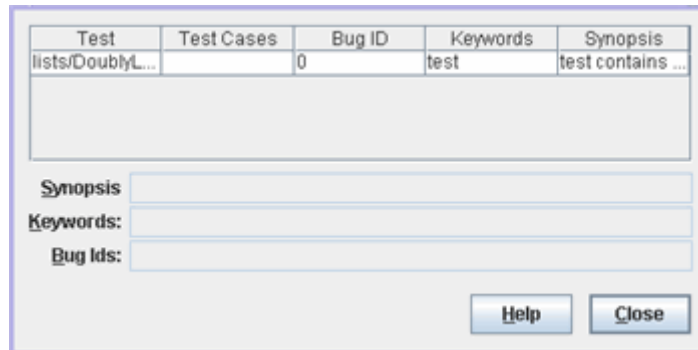
To open the Exclude List browser, choose View > Configure > Show Exclude List from the Test Manager menu bar.

The harness opens the Exclude List browser and displays the contents of the exclude list used to run tests and details about each test. You can use the browser to view but not edit the contents of the exclude list.

Use the Configuration Editor to add or remove exclude lists in a configuration. See Configuring a Test Run for detailed information.

Exclude List Contents

Information about individual tests in the exclude list is displayed in single, multi-column rows. Click a test in the list to display specific details in the text fields at the bottom of the panel.

Test Details

The following table describes the details displayed about individual tests highlighted in the Exclude List contents area.

TABLE 50   Exclude List Test Details

Field	Description

TABLE 50   Exclude List Test Details

Synopsis	Provides annotated information about the excluded test
Keywords	Provides a list of keywords that describe why the test was excluded
Bug Ids	Lists the bug tracking IDs for the excluded test

Displaying the Question Log

The harness creates a log file of all the completed questions in the current, saved configuration file and their answers. The harness does not update the question log until you save the configuration file or click the **Done** button in the Configuration Editor.

Open the Question Log of the current, saved configuration file by choosing View > Configure > Show Question Log from the Test Manager menu bar.

The log provides a list of all questions in the saved configuration with links to the following details about each question:

- Question
- Question tag
- Question description
- Response (if appropriate)

Viewing Test Manager Properties

To view the properties of a test manager, choose View > Properties. The harness opens the Test Manager Properties viewer.
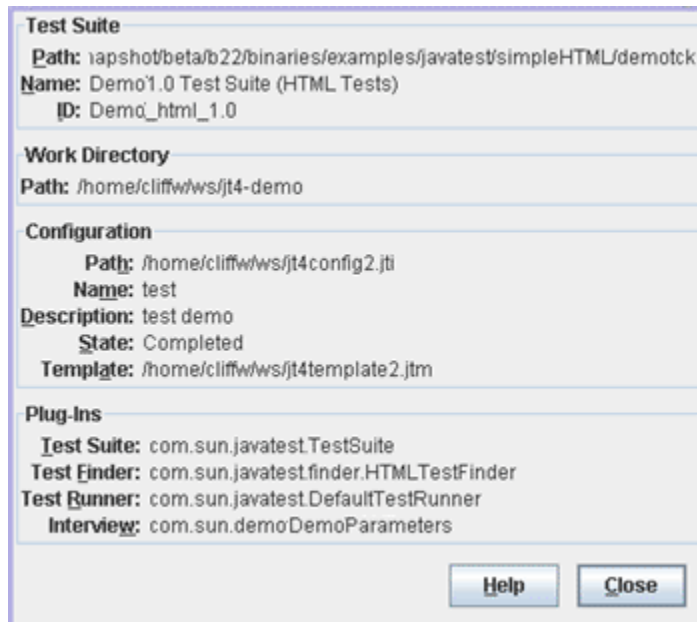
The Test Manager Properties viewer contains the following four areas:

- [Test Suite](#)
- [Work Directory](#)
- [Configuration](#)
- [Plug-Ins](#)

All value fields in the viewer can be highlighted and copied to the clipboard with keystroke Control-C.

Test Suite

The Test Suite properties area displays the Path, Name, and ID of the current test suite opened by the test manager.

Work Directory

The Work Directory properties area displays the path of the current work directory opened by the test manager.

Configuration

The Configuration properties area displays the Path, Name, Description, State, and Template of the current configuration interview opened by the test manager. The State field indicates whether the configuration is complete and tests can be run. It also identifies the availability of special filters.

Plug-Ins

The Plug-Ins properties area displays the name of the plug-ins used by the Test Manager. The plug-ins are provided by the test suite architect. The following table describes the properties that might be identified in the Plug-Ins properties area.

TABLE 51 Plug-Ins Area

Property	Description
Test Suite	The fully qualified name of the test suite class used by the test manager.
Test Finder	The fully qualified name of the test finder class used by the test manager.
Test Runner	The fully qualified name of the test runner class used by the test manager.
Interview	The fully qualified name of the interview class used by the test manager.

Viewing Test Suite Errors

When the harness detects errors in the test suite, it displays an error dialog box and enables the View > Test Suite Errors menu item.
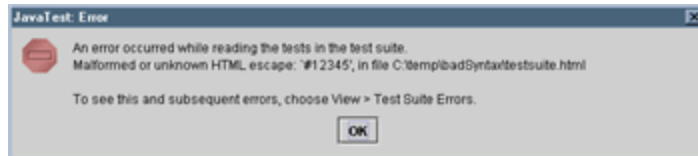
To display the Test Suite Errors viewer, you can choose View > Test Suite Errors from the Test Manager menu bar.
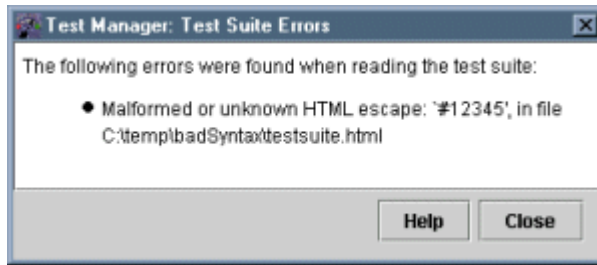
The dialog box displays a list of errors detected in the test suite. Unless instructed otherwise, report any test suite errors to the owner of the test suite.

9

Creating and Displaying Reports

The harness does not automatically create reports at the end of a test run. You must use the harness to manually create a new report after the test run. You can create and view reports containing the following test run information:

- Tests grouped by test status
- Configuration interview questions and answers
- Test environment used for the test run

See Creating Reports for a description of how to create test reports.

To view reports in the harness Report Browser, choose Report **>** Open Report from the menu bar. See Displaying Reports for a description of how to view reports.

Because harness reports contain relative and fixed links to other files, you must update these links when moving reports to other directories. The harness provides a command-line utility for you to use when moving reports to other directories. See Moving Test Reports in the *Command-Line Interface User's Guide* for a description of how to use the EditLinks utility to move reports.

Creating Reports

The harness does not automatically create reports of test results after a test run. You must create test reports either from the harness GUI or from the command line in batch mode (see Writing Reports in the *Command-Line Interface User's Guide*).

Creating a Test Report

HTML Options

HTML Files

Reports might contain configuration values as well as test results. To ensure that any configuration values printed in reports are those that were used to run the tests, observe the following precautions after running tests for which you intend to generate one or more reports:

Do not change any values in the current configuration.

Do not load a different configuration into the work directory.

Note – To minimize the chance of creating reports with configuration values that are inconsistent with the test results, either create reports after running the tests or use different work directories for different configurations.

▼ Creating a Test Report

To create a test report from the harness GUI, complete these steps:

Choose Report > Create Report from the Test Manager menu bar. The harness opens the Create a New Report dialog box.
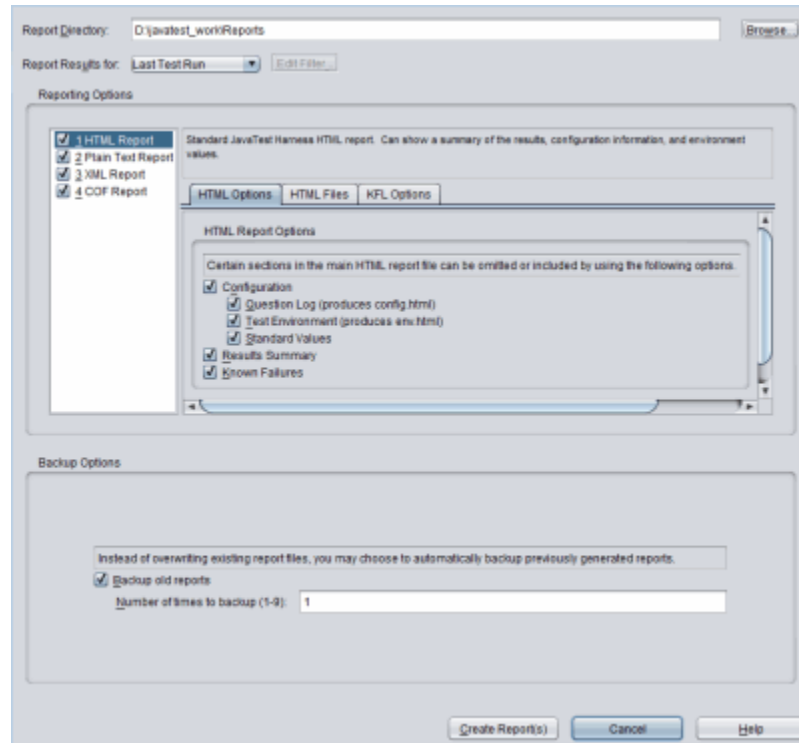
Type the name of a report directory in the Report Directory field or click the Browse button to specify where to put the new reports.

You can either specify a new directory or an existing directory. If you ran reports earlier, the Report Directory field displays the directory from the previous run. If you use an existing report directory, the harness can save the previous reports as

backups when it writes the new reports. Use the settings in the Backup Options pane to backup old reports and specify the number of backup reports to keep in the report directory.

The "Report Results for" drop down specifies the filter used to select the test results that are reported. You can choose Last Test Run, Current Configuration, All Tests, *Custom*, or Certification. If the *Custom* filter was previously renamed, the dialog box displays the current name of the custom filter. You can create or modify a custom filter for generating test reports for a specific set of test criteria.

See [Using View Filters](#) for a description of the filters. See [Custom View Filter](#) for a description of how to create a custom view filter.

Check "Backup Options settings" to enable the report backup option, and specify the number of backups preserved.

When "Backup old reports" is enabled, the harness saves the previous reports by appending a tilde and a sequential number to the existing directory names (such as `html~1~`). This effectively moves them out of the way, making way for the new report(s). The links inside the backup `index.html` file are rewritten to refer to the correct (renamed) subdirectory name. For example, a report directory with one backup would look like:

TABLE 1-1

```
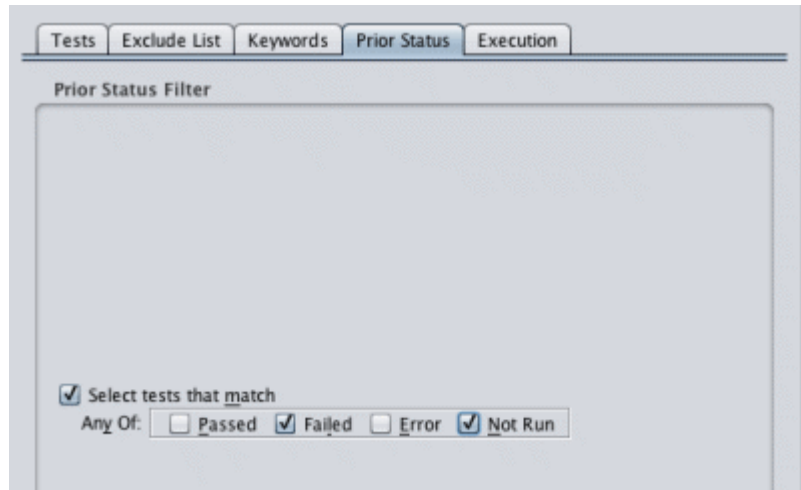html/
html~1~/
index.html
index.html~1~
reportdir.dat
text/
text~1~/
```

The harness maintains the specified number of backup copies by deleting the oldest copy when the limit of backups is reached for a specific report. Changing this or any other setting in the report dialog box does not affect any previously saved backup reports. Existing backups are not deleted if backups are turned off.

The harness backs up the entire set of reports. If the first set of reports included HTML, Text, XML, and COF formats, and the next set of reports are generated in XML format only, the harness creates backups of the previous reports (appending

the appropriate tilde and sequence number to the backup directory name). The harness generates the new top-level `index.html` file and updates its hyperlinks to maintain a self-consistent backup report set.

In the left column, choose the report format(s) to be generated by the harness. The options set in this dialog are persistent. Once set, they are used each time a report is generated. These settings also apply to reports printed from the command-line interface.

HTML Report: An HTML report showing configuration information, a result summary, and environment values. See HTML Options and KFL Options.

Plain Text Report: A text report showing only the test names and their result status. If the configuration names at least one known failures list, known failure analysis reports are also created.

XML Report: An XML report containing configuration information, a result summary, and all result data for each test. This report is versatile - it can be used as input to Merge Reports and can be converted to other report formats. However, the file can be larger and slower to generate than other formats.

COF Report: The report browser displays a `cof.xml` hyperlink to the COF report page, which contains a report in the COF format. COF is an extensive XML format containing environment, configuration and test result data.

Click the Create Report(s) button.

The harness writes the reports and displays a dialog box that gives you the option of either viewing the new reports in the report browser or returning to the Test Manager window.

HTML Options

If you selected HTML Report in the Report Formats list, use the HTML Options tab to select the sections of the main HTML report file that are generated.
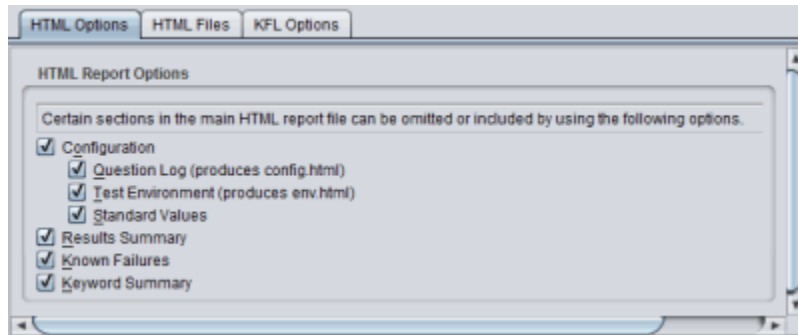
The following options are available for generating HTML reports:

Configuration - Selecting this option enables all of the following subordinate options:

> **Question Log** - Generates a report that is the equivalent of the Configuration Question Log.

> **Test Environment** - Generates a report that is the equivalent of the Show Environment dialog box.

> **Standard Values** - Generates a report that contains the values from the Quick Set Mode.

Results Summary - Generates a report of the pass, Fail, Error, Not Run, and Total values. The HTML report provides hyperlinks to content in the other HTML files. If it is not selected, the hyperlinks are not generated.

Known Failures - Generate a report of known failures, based on one or more Known Failures List (KFL) files. When the value for "Specify a Known Failures List?" is set to Yes in the configuration editor, and one or more KFL files have been specified, the settings are stored in your user preferences and used as the default when the KFL option is selected.

Keyword Summary - Generates a report that provides a count of the number of occurrences of keywords that appeared in the selected tests.

HTML Files

Use the options on this tab to specify the content and the location of HTML reports. The main report is the home page for browsing HTML reports. It can be named `report.html` or `index.html`. The main report contains hyperlinks to the extra files you choose to generate. The extra files you can choose are as follows:

Passed Tests (`passed.html`): Tests that were executed during the test run and passed.

Failed Tests (`failed.html`): Tests that were executed during the test run but failed.

Error Tests (`error.html`): Tests that had errors and could not be run.

Not Run Tests (`notrun.html`): Tests that were not excluded from the test run but were not run.

The main report and any extra files are written to the location you specified in the Report Directory field.

KFL Options in HTML Reports

Known failure list reports enrich the reporting output so you can monitor the status of certain tests across test runs. This section describes how to create and use a known failures list and discusses KFL reports and the [Known Failure Analysis](#).

Using a Known Failures List

A known failures list (KFL) is list of tests or test cases that are known to fail. Its purpose is to provide failure data for reporting, so that failure behavior can be tracked over time. Using a known failures list is optional and the feature is off by default. To activate the feature, answer yes to the "Specify a Known Failures List?" question in the configuration editor.

Once KFL files have been specified, (see [Creating a Known Failures List](#) and [Specifying a Known Failures List](#)) you can choose the Known Failures option in the HTML Options tab. This enables Known Failure Analysis options on the KFL

Options tab. Options you check on this tab are preserved as preferences for future reports. This is true if reports are launched from the user interface or the command line.

Reporting Options:

Check for test discrepancies. This feature is No by default. To activate this option, choose it in the user interface.

Optional files/sections to generate:

These options are set to Yes by default.

List tests which are expected to fail and are still failing. See <u>Unexpected Error</u> (`kfl_fail2error.html`).

List tests which were expect to fail but resulted in error instead. See <u>New Passed</u> (`kfl_fail2newpassed.html`).

Report on tests in KFL which are now missing.

Any generated data is added to the <u>Known Failure Analysis</u> which is found towards the end of the HTML report (`report.html` and/or `index.html`). It is a summary of various comparisons between the selected result set (Current Configuration, Last Test Run, etcetera) and the items listed in the KFLs provided in the configuration.

Creating a Known Failures List

The KFL is a text file with the extension `.kfl`.

Note – The KFL file name or its path must not begin with a dash "-" or contain spaces. As described in <u>Specifying Known Failures Lists With kfl</u> in the *Command-Line Interface User's Guide*, a space is a separator on the command line, therefore file path arguments such as `C:\Program Files\myconfig\foobar` do not work.

The `.kfl` file lists one test or test case per line. You have the option of specifying a bug ID following the test name (a space separator is required). Use # for comments. For example:

```
# Demo.kfl
lists/DoublyLinkedList/insertTest.html 0000123
BigNum/subtractTest.html 0000456
BigNum/compareTest.html 0000789
```

If you provide a bug prefix URL, that ID will be appended to that address, creating a convenient link in your report. To specify the bug prefix URL, select File > Preferences. Under Test Manager, select Reporting, and specify the URL.

This feature does not validate the URL or perform any processing.

Specifying a Known Failures List

Follow these steps to use one or more known failure lists to add failure analysis data to HTML reports.

From the configuration editor, select Configure > Edit Configuration.

Under Parameters, select "Specify a Known Failures List?" and choose Yes. Once Yes is chosen the parameter "Specify Known Failures List Files" is active.

Choose one or more KFL files, as described in the configuration editor's KnownFailuresListInterview.

The files you choose are stored in your user preferences and are considered the defaults.

If you have specified KFL files in the configuration editor it's preferable to modify the list using the configuration editor. If you want to modify the list of KFL files from the command line, See Specifying Known Failures Lists with kfl in the *Command Line Interface User's Guide*.

Known Failures List Reports

If you specified a known failures list in the configuration editor, the HTML report will include a section titled Known Failure Analysis. When you create a report you can check the HTML Option "Known Failures" to create the New Passed, Missing (not run), and New Failures reports. The options on the KFL Options tab generate reports for New Passed, Missing (test not found) and Old Failure.

Known Failure Analysis

The Known Failure Analysis table in the HTML report attempts to describe the differences found in the set of results being analyzed versus what you provided in the KFL. The KFL is the set of expected failures of tests and test cases; if they did not fail in the current set of results, that is considered a discrepancy. This is a sample table:

Known Failure Analysis

Known Failure File(s):

- D:\DemoTSWork\demo.kfl

KFL Discrepancies	Tests (4)	Test Cases (0)
New Passed	0	0
Unexpected Error	0	0
Unrelated Errors	0	(not calculated)
Missing (not run)	0	0
Missing (test not found)	3	0
New Failure	1	0
Old Failure	0	(not calculated)

The headers **Tests (#)** and **Test Cases (#)** contain a number representing the number of discrepancies found in that column. The numbers below that header should add up to that number (except the Old Failure does not count as a discrepancy).

The Known Failure Analysis table contains numbers linked to each of the report categories below. If the number is clicked, the user is taken to a file which has details - basically the information you'd find in the KFL file. The test name or test case name and the associated bug IDs. The test names are hyperlinked to the JTR result file in the work directory and the bug ids are hyperlinked if the bug URL preference is set in the Test Manager Preferences (as described in Reporting).

New Passed

A test or test case in the current set of results has a Passed status, when it was expected to fail.

Sample scenario:

A product under development has a defect described in bug 1234, which exists in builds 1-5. The quality department now expects an error in test `abc123` whenever the tests are run against the product, and has put test `abc123` on the known failures list (KFL).

Build 6 arrives and test `abc123` now passes because the defect has been fixed.

By looking at this Fail to Pass section of the report, the quality department will see test `abc123` listed and investigate the reason this test is suddenly passing.

After confirming the fix, a typical action would be to remove test `abc123` from the KFL.

Unexpected Error

A test or test case in the current set of results has an Error state, but it was expected to Fail. An Error usually indicates some sort of problem executing the test. Investigate items in this category to see why they resulted in an Error rather than a simple Pass or Fail.

Missing (Not run)

For some reason the test on the KFL was not run in the set of results being reported. It may not indicate a problem if a partial set of results is being analyzed, or if the KFL contains a wide set of tests, all of which would never appear in a single set of results.

Missing (test not found)

The results for a test listed in the KFL are missing. Because the test was listed in the KFL, it was expected to exist and Fail. A missing test may not indicate a problem, but should be investigated. This section can be disabled by deselecting the appropriate option in the HTML KFL report options in the Create Report dialog.

New Failure

Any test or test case which has a Failed status in the current results but does not appear in the KFL.

Old Failure

The lists of tests and test cases which failed and were expected to fail. This is not a discrepancy and is provided for informational purposes.

Unrelated Errors

Error results present in the results being reported upon, which do not correspond to any entries on the KFL list(s) supplied. This is a convenience listing so that errors can be examined at the same time as known failure list discrepancies.

Report Files

HTML Report Files

The possible contents of the report directory are as follows:

```
index.html
/html
    config.html
    env.html
    error.html
    error_gr.html
    excluded.html
    failed.html
    failed_gr.html
    kfl_fail2error.html
    kfl_fail2fail.html
    kfl_fail2missing.html
    kfl_fail2notrun.html
    kfl_fail2pass.html
    kfl_newfailures.html
    kfl_otherErrors.html
```

```
        kfl_tc_fail2error.html
        kfl_tc_fail2missing.html
        kfl_tc_fail2notrun.html
        kfl_tc_fail2pass.html
        kfl_tc_newfailures.html
        notRun.html
        passed.html
        passed_gr.html
        report.css
        report.html
   /text
        summary.txt
   /xml
        report.xml
```

In the `html` directory, the KFL report file names correspond to the KFL reports as follows:

TABLE 52 Known Failure List Reports

HTML Report	File Names
New Passed	`kfl_fail2pass.html`, `kfl_tc_fail2pass.html`
Unexpected Error	`kfl_fail2error.html`, `kfl_tc_fail2error.html`
Missing (Not run)	`kfl_fail2missing.html`, `kfl_tc_fail2missing.html`
Missing (test not found)	`kfl_fail2notrun.html`, `kfl_tc_fail2notrun.html`
New Failure	`kfl_newfailures.html`, `kfl_tc_newfailures.html`
Old Failure	`kfl_fail2fail.html`, `kfl_tc_fail2fail.html`

Plain Text Report Files

A plain text report directory contains a `text/` directory which contains the main text report, `summary.txt`.

If the configuration associated with the work directory names at least one [known failures list](#) (KFL) file, the `text/` directory contains the additional known failure analysis files listed here:

TABLE 53   Known Failure List Reports

Text Report	File Names
New Passed	`kfl_fail2pass.txt,` `kfl_tc_fail2pass.txt`
Unexpected Error	`kfl_fail2error.txt,` `kfl_tc_fail2error.txt`
Missing (Not run)	`kfl_fail2missing.txt,` `kfl_tc_fail2missing.txt`
Missing (test not found)	`kfl_fail2notrun.txt,` `kfl_tc_fail2notrun.txt`
New Failure	`kfl_fail2fail.txt,` `kfl_tc_fail2fail.txt`
Old Failure	`kfl_newfailures.txt,` `kfl_tc_newfailures.txt`

The files whose names begin with `tc_` are present only if the test suite supports test case identification.

Displaying Reports

You can use the Report browser or a web browser to display harness reports.

▼ Display Reports in the Report Browser

Choose Report > Open Report from the menu bar.

The harness opens a file chooser dialog box for you to specify a report directory. When you specify a report directory, the harness opens the Report Browser.
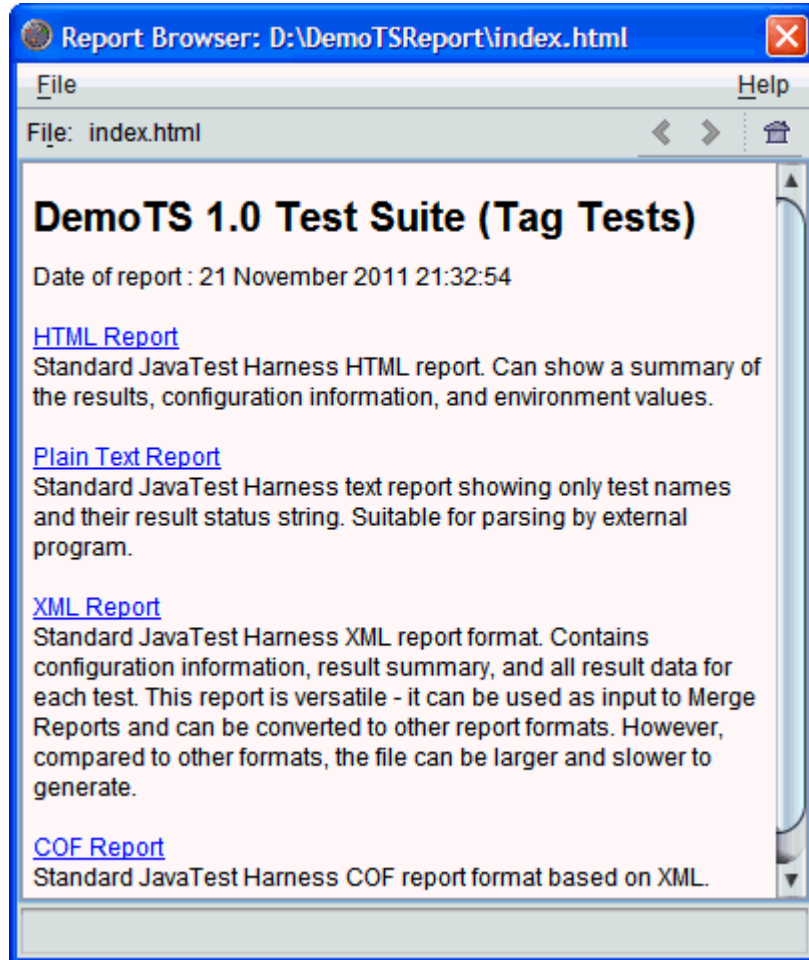
Choose the report format from the directory listing.

If you choose the HTML format version of the reports, a second directory listing is displayed. On the toolbar, you can use the report browser's back button (<) to return to the initial report page. You can also click the file name showing on the toolbar to open a dropdown of recently visited files and directories.
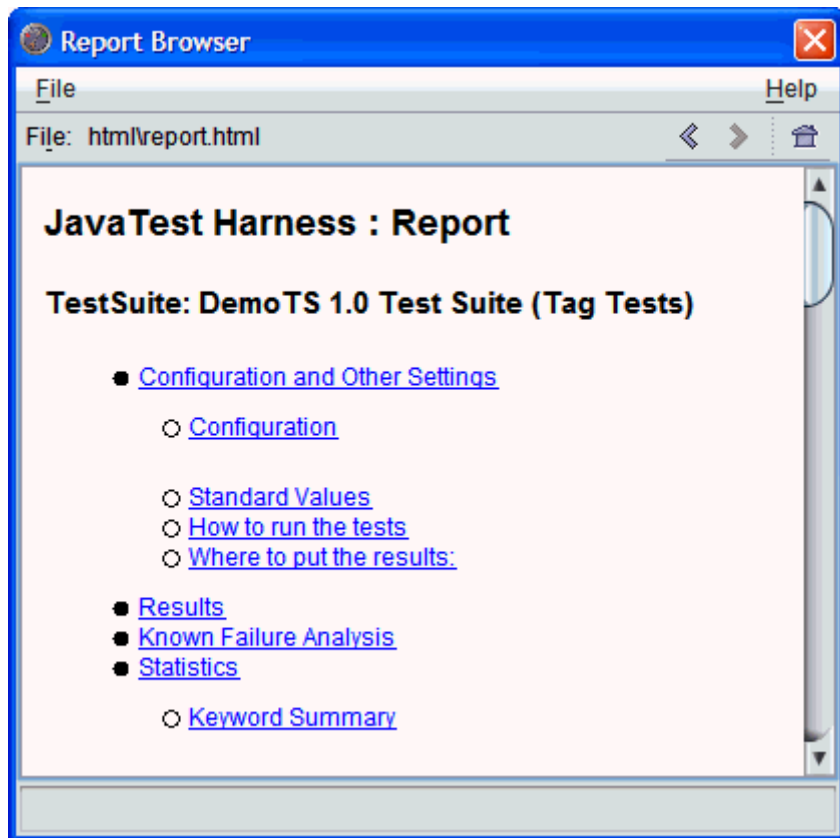
If you choose the text format version of the report, the browser displays a `summary.txt` hyperlink to the summary page. The summary page provides a text list of tests that were run and their test result status.

```
Report Browser
File                                                    Help
File: text\summary.txt                              ≪  ≫   ⌂

BigNum/AddTest.java              Passed. OK
BigNum/CompareTest.java              Passed. OK
BigNum/EqualsTest.java              Passed. OK
BigNum/LongConstrTest.java              Passed. OK
BigNum/SubtractTest.java              Passed. OK
lists/DoublyLinkedList/AppendTest.java  Passed. OK
lists/DoublyLinkedList/EqualsTest.java  Passed. OK
lists/DoublyLinkedList/InsertTest.java  Failed. one or more test cases failed
lists/DoublyLinkedList/RemoveTest.java  Passed. OK
lists/LinkedList/AppendTest.java        Passed. OK
lists/LinkedList/EqualsTest.java        Passed. OK
lists/LinkedList/InsertTest.java        Passed. OK
lists/LinkedList/RemoveTest.java         Passed. OK
lists/SortedList/EqualsTest.java        Passed. OK
lists/SortedList/InsertTest.java        Passed. OK
lists/SortedList/RemoveTest.java        Passed. OK

file:/D:/DemoTSReport/text/summary.txt
```

If you choose the XML format version of the report, the browser displays a
`report.xml` hyperlink to the XML report page. The XML Report page
provides a report of the configuration values used to run the tests, the tests that
were run, and their test result status.

This output is for automated processing purposes. It is not designed for human
readability.
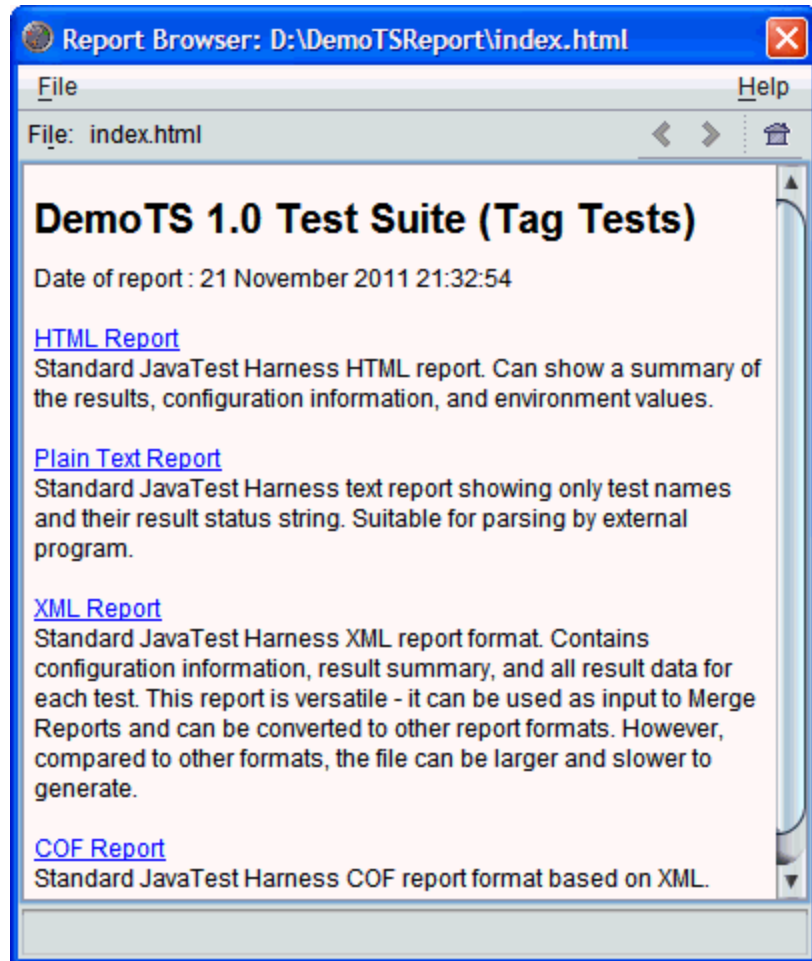
To see a list of the HTML reports click the Report directory hypertext link.

Where to put the results:

Work directory	D:\DemoTSWork
Report directory	D:\DemoTSReport\html

Usually you access the report files using the links in the top-level HTML report. You can also use the hyperlinks within report.html to display the detailed report files in the report browser.

Report Browser Contents

The following table describes the contents of the Report Browser.

TABLE 54   Report Browser Contents

Component	Description
Menu bar	The menu bar contains a File and a Help menu.
	Use the File menu to generate new reports, open existing reports, and to close the Report Browser.
	Use the Help menu to display Report Browser online help.
tool bar	The tool bar contains a File field and three navigation buttons.
	The File field displays the name of the current report and provides a drop-down list of reports previously opened in the browser. As reports are opened, the harness adds their names to the drop-down list enabling you to navigate to any previously displayed report.
	Returns to the previously displayed report page.
	Opens the next report page that was displayed.
	Returns to the `report.html` file. The `report.html` file is the root page and links to all of the other test report pages.
Contents area	The Report Browser displays the report file contents in the area below the tool bar. The Report Browser displays text files as well as HTML files. For HTML files, you can use hyperlinks in the report to display additional related reports.

See [Keyboard access](#) for a description of how the keyboard can be used to navigate the Report Browser.

HTML Report Files

The following table lists and describes the HTML report files.

TABLE 55 View Reports Offline

Report Files	Description
config.html	Contains the configuration interview questions and your answers used for the test run.
env.html	Contains the name-value pairs derived from the configuration file that were used for the test run.
error.html	Contains a list of the tests that had errors and could not be run.
error_gr.html	Contains a list of the tests that had errors and could not be run, grouped by status.
excluded.html	Contains a list of the tests that were excluded from the test run.
failed.html	Contains a list of the tests that were executed during the test run but failed.
failed_gr.html	Contains a list of the tests that executed during the test run but failed. The test are grouped by status.
notRun.html	Contains a list of all tests that were not excluded from the test run but were not run.
passed.html	Contains a list of the tests that were executed during the test run and passed.
passed_gr.html	Contains a list of the tests that were executed during the test run and passed, organized by status.
report.html	Contains links to all of the HTML files and additional information.

This report may also list known failure lists and exclude lists if they have been included in your configuration.

Viewing Reports Offline

You can view the reports offline by opening the html files in a web browser or opening the summary.txt report in any text editor.

The report.html file is the root file that links to the other HTML reports. The summary.txt report contains a list of all tests that were run, their test results, and their status messages.

10

Merging and Converting XML Test Reports

You can use the Report Converter tool to merge harness generated XML test reports from different work directories into a single report. XML files not generated by the harness are not supported by using the XML Report type in the Test Manager report system.

▼ Merge XML Reports

To merge XML reports generated by the harness, perform the following steps:

Choose either Tools **>** Report Converter or Settings > New from the menu bar to launch the Report Converter wizard.

The harness opens the Report Converter wizard.
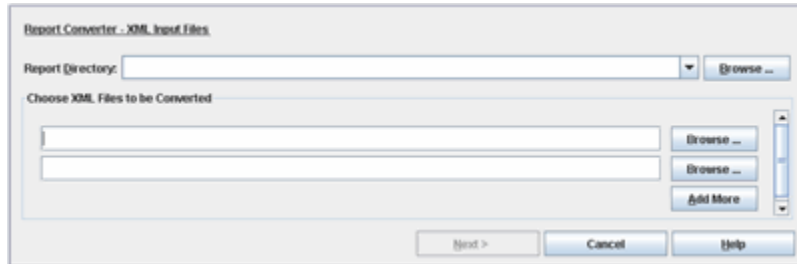
See [Report Converter Wizard](#) for a detailed description of the wizard.

In the XML Input Files panel, specify the directory where the Report Converter Tool writes the merged XML report file.

Specify the harness generated XML reports used as sources for generating the merged XML report file.

You can chose a single source file in the wizard if you are using the Report Converter tool to convert a harness generated XML test report to an output type provided by your test suite.

If you are merging multiple XML test reports, click the Add More button to open additional source file fields.

Click the Next button.

The harness displays the Report Options panel.
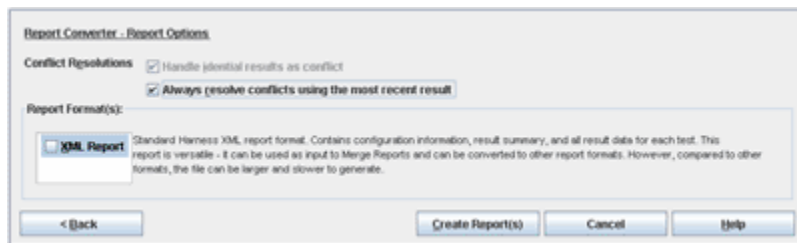
In the Report Options panel, specify how conflicts between merged reports are handled.

The default setting is to always handle identical results in the two reports as a conflict.

The option of resolving conflicts using the most recent result is not selected. When the report is created, the tool displays a Resolve Conflict dialog box for

each conflict that is detected. Choosing the option of having the tool automatically resolve the conflicts enables to tool to create the report without displaying the Resolve Conflict dialog box.

Choose the report output format.

Click the Create Report(s) button to create the report.

Resolve any conflicts identified by the tool.

If conflicts exist in the reports and the option to resolve conflicts by using the most recent result was not enabled, the tool displays a Resolve Conflict dialog box for each conflict that is detected.
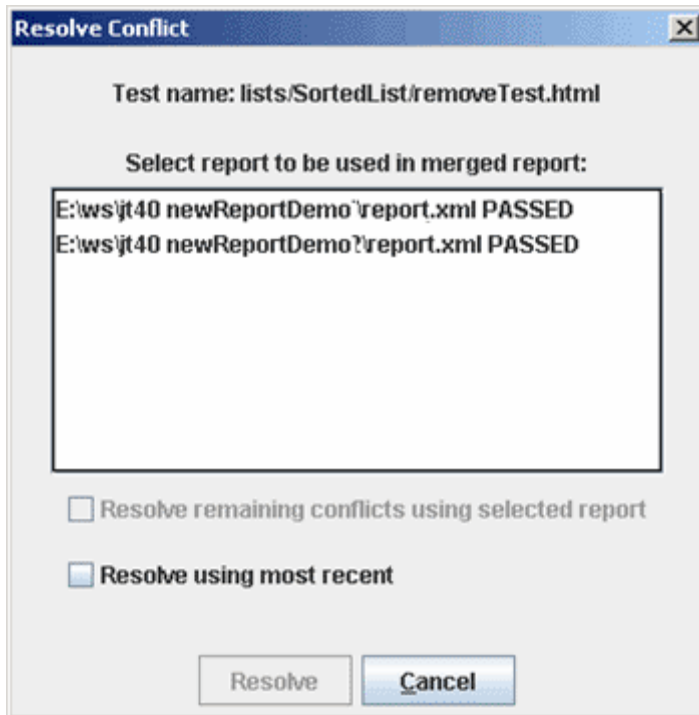
If you only click the Resolve using most recent checkbox and click the Resolve button, the tool resolves the conflict for the specified test and displays the next conflict detected.

If you select a report in the list, the tool enables the Resolve remaining conflicts using selected report checkbox. If you only click the Resolve remaining conflicts using selected report checkbox and click the Resolve button, the tool resolves all conflicts in the generated tests.

Resolve Conflict ✕

Test name: lists/LinkedList/appendTest.html

Select report to be used in merged report:

E:\ws\jt40 newReportDemo`\report.xml PASSED
E:\ws\jt40 newReportDemc\report.xml PASSED

☐ Resolve remaining conflicts using selected report

☐ Resolve using most recent

[Resolve] [Cancel]

11
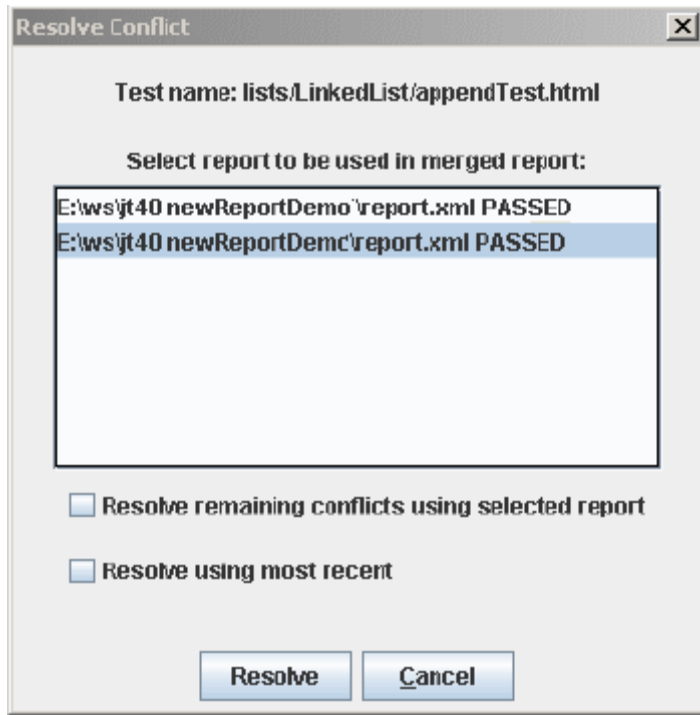
Troubleshooting With the GUI

The harness provides information you can use in troubleshooting problems. To troubleshoot problems using the harness see the following topics:

- [Harness Fails During Use](#)
- [Problems Running Tests](#)
- [Problems Viewing Test Results](#)
- [Output Overflow Message Displayed in Test Run Messages](#)
- [Problems Viewing Reports](#)
- [Problems Writing Reports](#)
- [Problems Moving Reports](#)

Harness Fails During Use

If the harness fails, you can use the `harness.trace` file in your work directory to help troubleshoot the problem. The `harness.trace` file is a plain-text file that contains a log of harness activities during the test run. It is written in the work directory, is incrementally updated, and is intended primarily as a log of harness activity.

Problems Running Tests

The goal of a test run is for all tests in the test suite that are not filtered out to have passing results.

If the root test suite folder contains tests with errors or failing results, you must troubleshoot and correct the cause to satisfactorily complete the test run. See Troubleshooting a Test Run for information about the resources that the harness provides for troubleshooting.

Tests with Errors

Tests with errors are tests that could not be executed by the harness. These errors usually occur because the test environment is not properly configured. Use the Test tabbed panes and the Configuration Editor to help determine the change required in the configuration.

The following is an example of how the Test Manager tabbed panes and the Configuration Editor can be used to identify and correct a configuration error:

Use the test tree to identify the folder containing test that had errors.

Click the folder icon to open its Summary tab in the Test Manager window.

Click the Error tab to display the list of tests in the folder that had errors.

Double-click a test in the list to display it in the test tree and view its detailed test information.

Click the Test Run Messages tab to display detailed messages describing what happened during the running of each section of the test.

The contents of each output section vary from test suite to test suite. Refer to your test suite documentation for detailed descriptions of the test section messages when troubleshooting a test run.

> Click the Configuration tab to display a two-column table of the name-value pairs that were derived from the configuration file and actually used to run the test.

The names in the table identify test environment properties used by the harness to run the test. The values displayed were used to run the test. Refer to your test suite documentation for detailed descriptions of the name-value pairs for your test.

Choose View > Configuration **>** Show Question Log to view the Question Log of the current, saved configuration.

Use the question log to identify the configuration value that is incorrect and its configuration question.

Search the configuration file for the specific characters or character strings that must be changed.

See Searching a Configuration for a detailed description of how the current configuration can be searched for a character or string of characters.

Click the Done button to save your changes to the configuration file

Rerun the tests.

⊠Tests that Fail

Tests that fail are tests that were executed but had failing results. The test or the implementation might have errors.

The following is an example of how the Test Manager tabbed panes can be used to identify and correct a test failure:

1. Use the test tree to identify the folder containing test that had errors.

2. Click the folder icon to open its Summary tab in the Test Manager window.

3. Click the Error tab to display the list of tests in the folder that had errors.

4. Double-click a test in the list to display it in the test tree and view its detailed test information.

5. Click the Test Run Messages tab to display detailed messages describing what happened during the running of each section of the test.

The contents of each output section vary from test suite to test suite. Refer to your test suite documentation for detailed descriptions of the test section messages when troubleshooting a test run.

Problems Viewing Test Results

Most problems in viewing test results in the Test Manager result from the use of view filters (other than the All Tests view filter) with a current configuration set to run specific tests (such as, running tests based on their prior status). When a view filter other than All Tests is used in the Test Manager, only the fields of the filtered category appear to be updated in the Test Manager Summary tab. This is normal behavior of the GUI when view filters are used.

The harness *only* displays tests in the GUI that match the specified view filter criteria. All other tests are displayed as Filtered Out.

As test results change, the harness moves the tests to the Filtered Out category (not to a test result category) in the Test Manager and turns the appropriate node in the test tree gray. Consequently, the fields for the other categories in the Test Manager Summary tab might not appear to be updated.

Example:
If you change the current configuration to run only tests with a prior status of Failed and use the Current Configuration view filter, the Test Manager displays all tests that are not a Failed status as Filtered Out. This enables you to monitor only those tests with a previous Failed status. When you begin the test run, any tests with Passed or Error results are displayed in the Filtered Out category of the Test Manager. To view the actual test results, change to a different view filter. See View Filters for additional information. of the view filters.

Output Overflow Message Displayed in Test Run Messages

An output stream in the Test Run Messages Pane might display the following message:

```
 Output overflow:the harness has limited the test output of
 the text to that at the beginning and the end, so that you
 can see how the test began, and how it completed.

 If you need to see more of the output from the test,
 set the system property javatest.maxOutputSize to a higher
 value. The current value is 100000
Set the system property of javatest.maxOutputSize
 by using the syntax in:
```

```
java -Djavatest.maxOutputSize=200000 -jar lib/javatest.jar
....
```

If you see this message, restart the harness with the harness VM system property `javatest.maxOutputSize` value set to a greater number and rerun the test to ensure that all of the data is saved.

The value of `javatest.maxOutputSize` is an integer subject to the maximum integer size of the VM. It sets the number of characters allowed in the output.

Problems Viewing Reports

The harness does not automatically generate reports of test results after a test run. You must generate test reports either from the command line or from the harness GUI. See <u>Creating Reports</u> for detailed information.

Problems Writing Reports

Filters are used to write test reports for a specific set of test criteria. Verify that you are using the appropriate filter to generate reports of test results. See <u>Creating Reports</u> for detailed information.

Problems Moving Reports

Test reports contain relative and fixed links to other files that might be broken when you move reports to other directories.

You must update these links when moving reports to other directories. The harness provides an `EditLinks` utility that updates the links in the reports for you when moving reports. See the *Command-Line Interface User's Guide*.

12

Glossary

.jte Files See [environment files](#).

.jti Files See [configuration file](#).

.jtp Files See [parameter files](#).

.jtr File See [test result files](#).

.jtx Files See [exclude list](#).

.kfl Files See [Known Failures List File](#).

A

All Tests View Filter A test tree [view filter](#) that displays all folders and tests in the [test suite](#).

B

C

Class The prototype for an object in an object-oriented language. A class might also be considered a set of objects that share a common structure and behavior. The structure of a class is determined by the class variables that represent the state of an object of that class and the behavior is given by a set of methods associated with the class.

Configuration   Information about the computing environment required to execute a test suite.

In the GUI, you can use the Configuration Editor to collect or modify configuration information or to load an existing configuration. See Configuration Editor. The Configuration Editor collects the following two types of data in a configuration file:

- Test environment

- Standard Values

In the command-line interface, you can perform the following tasks:

Use the `EditJTI` utility to modify configuration information (see `EditJTI`).

Set specific configuration values in the command line when starting the harness.

Configuration Editor   The Configuration Editor provides two editing modes, the Question mode and the Quick Set mode. Use the Question mode to create a configuration file for the test run and to search interview titles, questions, and answers for character strings. Use the Quick Set mode to modify specific runtime values.

To open the Configuration Editor, choose Configure **->** Edit Configuration from the Test Manager menu bar.

Configuration File   Contains all of the information collected by the configuration editor about the test platform.

The harness derives the configuration values required to execute the test suite from environment entries in a configuration file (`.jti`).

Use the Configuration Editor or EditJTI to change configuration values in a `.jti` file.

You can also set specific values in the command line.

Configuration Value   A value specified by the user for the purpose of configuring a test run.

Configuration values are derived from environment entries in a configuration file (`.jti`) and used by test suite specific plugin code to execute and run tests.

Use the Configuration Editor or EditJTI to change the configuration values in the `.jti` file. You can also set specific configuration values in the command line.

For legacy test suites the configuration value is read from an environment file (`.jte`). Current test suites do not use or support the environment file.

Current Configuration   The configuration containing the test environment and standard values currently loaded in the test manager or specified in the command line for use in running tests and displaying test status.

Current Configuration Filter   A filter that only displays or generates reports for the folders and tests that have not been filtered out in the configuration.

D

Desktop   The configuration and layout of the windows used by the JT harness.

The desktop is saved when you exit from the harness and is automatically restored in your next session.

The JT harness desktop is displayed using <u>tabs</u>.

E

Environment   See <u>Test Environment</u>.

Environment Entry   A name-value pair derived from a configuration file and used by test suite specific plugin code to execute and run tests. These name-value pairs provide information (<u>configuration values</u>) about how to run tests of a test suite on a particular platform.

For legacy test suites, the name-value pairs are read from an <u>environment file</u> (.jte) and derived from the <u>configuration file</u> (.jti). Current test suites do not use or support the environment file.

Environment Files   Contain one or more test environments used by legacy test suites. Environment files are identified by the .jte extension in the file name. Current test suites do not use or support the environment file.

Error   The test is not filtered out and the JT harness could not execute it. There are no test results for tests having errors. Errors usually occur because the test environment is not properly configured.

In the GUI, the JT harness displays error icons for tests with errors and for folders containing any tests with errors. Folders marked with error icons can also contain tests and folders that are Failed, Not Run, Passed, and Filtered out.

Exclude List   Exclude list files (*.jtx), supply a list of invalid tests to be filtered out of a test run by the test harness. The exclude list provides a level playing field for all implementors by ensuring that when a test is determined to be invalid, then no implementation is required to pass it. Exclude lists are maintained by the technology specification Maintenance Lead and are made available to all technology licensees.

In the GUI, use the configuration editor to add or remove exclude lists from a test run. In the command line, you can specify an exclude list in the command.

To view the contents of an exclude list, choose Configure **->** Show Exclude List from the Test Manager menu bar. Exclude lists can only be edited or modified by the test suite Maintenance Lead.

F

Fail Test results determined by the JT harness that do not meet passing criteria.

In the GUI, the JT harness displays Failed icons for tests that the test suite has determined have failing results and for folders containing any tests with fail results. Folders marked with Failed icons can also contain tests and folders that are Not Run, Passed, and Filtered out.

Filtered Out Folders and their tests that are excluded from the test run by one or more [test run filters](#).

In the GUI, Filtered Out folders and tests are identified in the test tree by grey ▇folder and ▍test icons.

Filters A facility in the JT harness that accepts or rejects tests based on a set of criteria. There are two types of filters in the JT harness, view filters and run filters. View filters are set in the [Test Manager](#) to display the results for specific folders and tests and to create test reports. Run filters are set in the [Configuration Editor](#) or are specified as commands in the command-line to specify which tests are run.

G

Glossary Use the Glossary tab in the left pane of the help viewer to open the glossary.

The glossary pane is divided into two areas. The top area displays the list of terms and the bottom area displays their definition.

Glossary entries are listed in alphabetical order. You can scroll through the list or search for a specific term.

To search for a specific term, enter it in the Find field and press Return. If the term is in the glossary, the harness highlights it in the list and displays its definition. Press Return to repeat the search.

H

I

Interview   For the JT harness to execute a test suite, it requires information about how your computing environment is configured. The Configuration Editor uses an interview to simplify the process of collecting this information.

Because the quantity and scope of this information depends on the test suite, a test suite might include a specific interview for the Configuration Editor to use.

The information collected by the interview is written to a configuration file (`.jti`) and used by the JT harness to derive the [environment variables](#) required execute the test suite.

Interview File   See [configuration file](#).

J

**JavaTest Harness
Preferences**   A dialog box that you can use to set the window style and tool tip options used by the JT harness. To open the Preferences dialog box, choose File **->** Preferences from the Test Manager menu bar.

JTI   Standard file extension for a configuration file. See [configuration file](#).

K

Keywords   Special values in a [test description](#) that describe how the test is executed.

Keywords are provided by the test suite for use in the Configuration Editor or command line as a filter to exclude or include tests in a test run.

**Known Failures List
Files**

Known Failures List (KFL) files (`*.kfl`), list tests that are known to fail.

The purpose of this option is to enrich the reporting output so you can monitor the status of certain tests across test runs. If a KFL is specified, the HTML report includes a Known Failure Analysis section. This section provides hypertext links to the tests listed in the KFL. Also, if bug IDs are specified in the KFL the IDs are hypertext linked to a bug tracking system (specified as a URL in File > Preferences > Test Manager > Reporting).

One or more KFLs can be specified as a parameter in the configuration editor. KFLs can also be specified from the command line.

L

Last Test Run View

Filter   A <u>filter</u> that displays the results of those tests and folders included in the last test run either in views of the <u>test tree</u> or in test reports.

Log Level   A descriptive criteria (Critical, Warning, Monitoring Info, and Debug Info) applied to each message within a log.

M

N

Not Run   The test is not filtered out but the JT harness has not yet run it.

The JT harness displays Not Run icons for tests that have not *yet* been run. Folders containing tests that have not been run are displayed with the Not Run icon only when they do not also contain tests or folders with <u>Error</u> or <u>Failed</u> results.

O

P

Parameter Files   Legacy files used to configure how the harness runs legacy test suites on your system. Parameter files use the file name extension `.jtp`.

Although parameter files are deprecated, the harness provides support for those test suites that use parameter files. Current test suites do not use or support the parameter files.

Pass   Test results determined by the JT harness to meet passing criteria.

The JT harness displays Passed icons for tests that the test suite has determined have passing results and for folders containing only tests with passing results.

Port Number   A number assigned to the JT harness that is used to link specific incoming data to an appropriate service.

Prior Status   A <u>filter</u> used to restrict the set of tests in a test run based on the last test result information stored in the <u>test result</u> files (`.jtr`) in the work directory.

Use the configuration editor or command line to enable the Prior Status filter for a test run.

Progress Monitor A dialog box that displays detailed information about the current configuration of a test run. Information displayed in the Progress Monitor is not altered by <u>view filter</u> settings.

Q

R

Report Directory The directory in which the harness writes test reports.

The location of the report directory is set in the GUI or from the command line by the user when generating test reports.

S

Standard Values The Quick Set mode of the Configuration Editor displays the standard values of a configuration.

T

Tabbed A window style in which the JT harness <u>desktop</u> is a single top-level window that displays all JT windows as tabbed panes.

Test Case A subsection inside of a test. The terminology for a 'test case' varies among test harnesses, but in this harness, a test is a larger entity than the test case. A test contains zero or more test cases; a test with no test cases is simply a basic test entity in this harness. The 'test' is the primary entity rendered in the main tree display, summaries, etc. 'Test cases' are currently only recognized in limited areas of the harness.

Test Description Machine readable information that describes a test to the JT harness so that it can correctly process and run the related test. The actual form and type of test description depends on the attributes of the test suite. When using the JT harness, the test description is a set of test-suite-specific name-values pairs.

Each test in a <u>test suite</u> has a corresponding test description that is typically contained in an HTML file.

Test Environment   A collection of <u>configuration values</u> derived from <u>environment entries</u> in the configuration file that provide information used by test suite specific plugin code about how to execute and run each test on a particular platform.

When a test in a legacy <u>test suite</u> is run, the harness gives the script a test environment containing environment entries from configuration data collected by the configuration editor. See <u>configuration</u>.

For legacy test suites, the environment entries were read from an environment file. Use of environment files is deprecated, but the harness continues to provide support for test suites that use environment files. See <u>environment file</u>.

Current test suites do not use or support environment files.

Test Manager   The JT harness window used to configure, run, monitor, and manage tests from its panels, menus, and controls.

The Test Manager window is divided into two panes. It displays the folders and tests of a test suite in the tree pane on the left and provides information about the selected test or folder in the information panes on the right.

A new Test Manager window is used for each test suite that is opened.

Test Paths   The locations of the test folders and tests in the <u>test tree</u> specified in the Configuration Editor for running tests.

Test paths are used to specify the tests in a specific branch of the test tree or to specify a specific test to run. The harness walks the test tree and runs all specified tests (unless otherwise <u>filtered out</u>) that it finds.

Use the configuration editor to set the test paths listed in the test run configuration.

Test Result Files   Contains all of the information gathered by the JT harness during a test run.

The test result files (`.jtr`) are stored in a cache in the <u>work directory</u> associated with the test suite.

You can view the test result files in a web browser configured to use the JT harness `ResultBrowser` servlet.

Test Run Filters   Include or exclude tests in a test run. Tests are included or excluded from test runs by the following means:

- <u>Exclude lists</u>

- <u>Keywords</u>

- <u>Prior status</u>

Test run filters are set using the Configuration Editor or the command-line interface.

Test Script   A script used by the JT harness, responsible for running the tests and returning the status (pass, fail, error) to the harness. The test script must interpret the test description information returned to it by the test finder. The test script is a plug-in provided by the test suite. In the GUI, the Test Manager Properties dialog box lists the plug-ins that are provided by the test suite.

Test Suite   A collection of tests, used in conjunction with the JT harness, to verify compliance of the licensee's implementation of the technology specifications.

A test suite must be associated with a [work directory](#) before the harness can run its tests.

Test Tree   The hierarchical representation of the folders and tests in a [test suite](#).

The test tree is displayed in the [Test Manager](#) window and uses colored status icons to indicate the test status of the folders and tests. Use [view filters](#) to specify the folders and tests whose test status are displayed in the test tree.

U

V

View Filters   Include or exclude the results of tests in views of the [test tree](#).

View filter settings only change the Test Manager display and do not include or exclude tests from a test run. Use the [test run filter](#) settings to include or exclude tests from a test run.

W

Work Directory   A directory associated with a specific [test suite](#) and used by the JT harness to store files containing information about the test suite and its tests.

Until a test suite is associated with a work directory, the JT harness cannot run tests.

X

Y

Z

Index