Oracle® Application Testing Suite

Getting Started Guide Version 9.00 for Microsoft Windows (32-Bit) E15487-01

September 2009



Oracle Application Testing Suite Getting Started Guide, Version 9.00 for Microsoft Windows (32-Bit)

E15487-01

Copyright © 1997, 2009, Oracle and/or its affiliates. All rights reserved.

Primary Author: Rick Santos

Contributing Author: Mary Anna Brown

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this software or related documentation is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, the following notice is applicable:

U.S. GOVERNMENT RIGHTS Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, duplication, disclosure, modification, and adaptation shall be subject to the restrictions and license terms set forth in the applicable Government contract, and, to the extent applicable by the terms of the Government contract, the additional rights set forth in FAR 52.227-19, Commercial Computer Software License (December 2007). Oracle USA, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

This software is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications which may create a risk of personal injury. If you use this software in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure the safe use of this software. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software in dangerous applications.

Oracle is a registered trademark of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.

This software and documentation may provide access to or information on content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services.

Contents

Preface	ix
Audience	ix
Documentation Accessibility	х
Related Documents	х
Conventions	х

1 Introduction

About Oracle Functional Testing for Web Applications	1-1
Oracle Functional Testing for Web Applications Feature Highlights	1-2
About Job Scheduler	1-3
Job Scheduler Feature Highlights	1-4
About Oracle Load Testing for Web Applications	1-5
Oracle Load Testing for Web Applications Feature Highlights	1-6
	Oracle Functional Testing for Web Applications Feature Highlights About Job Scheduler Job Scheduler Feature Highlights About Oracle Load Testing for Web Applications

2 Oracle Application Testing Suite Basics

2.1	Installing and Starting Oracle Application Testing Suite	2-1
2.1.1	Using an Existing Database	2-1
2.1.2	Oracle Application Testing Suite System Setup	2-2
2.2	Oracle Functional Testing for Web Applications Main Window Features	2-4
2.2.1	Visual Script Pane	2-4
2.2.2	Browser Pane	2-5
2.2.3	Playback Results Pane	
2.3	Job Scheduler Main Window Features	2-6
2.3.1	Visual Script Job Pane	2-7
2.3.2	Results Pane	2-7
2.3.3	Job Scheduler Wizard	2-8
2.4	Oracle Load Testing for Web Applications Main Window Features	2-9
2.4.1	Build Scenario Tab	2-9
2.4.2	Set up Autopilot Tab	2-10
2.4.3	Watch Virtual User Grid Tab	2-11
2.4.4	View Run Graphs Tab	2-11
2.4.5	Create Reports Tab	2-12
2.4.6	ServerStats	2-13

3 Oracle Functional Testing for Web Applications Tutorial

3.1	Initializing the Tutorial	. 3-1
3.2	Example 1: Recording a New Visual Script	. 3-1
3.2.1	Start Oracle Functional Testing for Web Applications	. 3-2
3.2.2	Start a Recording	
3.2.3	Navigate the Web Site	. 3-3
3.2.4	Stop the Recording	. 3-3
3.2.5	Save the Script	. 3-3
3.3	Example 2: Working with Visual Scripts	. 3-3
3.3.1	Viewing Information About a Visual Script Item	. 3-4
3.3.2	Turning Automatic Testing On and Off	. 3-5
3.3.3	Modifying Default Tests	. 3-6
3.4	Example 3: Playing Back a Visual Script	. 3-7
3.5	Example 4: Analyzing Test Failures	3-10
3.5.1	Ignoring Failures	3-11
3.5.2	Accepting Changes Shown in the Script	3-12
3.5.3	Rejecting Problems Shown in the Script	3-12
3.6	Example 5: Adding Test Cases to the Visual Script	3-13
3.6.1	Record a New Script	3-14
3.6.2	Stop the Recording	3-14
3.6.3	Insert a Text Matching Test Case	3-14
3.6.4	Insert a Server Response Test Case	3-15
3.6.5	Insert a Form Element Test Case	3-16
3.6.6	Insert Table Test Test Case	3-18
3.7	Example 6: Using the Data Bank Wizard on a Search Form	3-20
3.7.1	Recording a Search	3-21
3.7.2	Viewing the Parameters in the Visual Script	3-21
3.7.3	Using the Data Bank Wizard to Map Variables	3-21
3.7.4	Using the Data Bank Wizard to Bind to a Data Bank	3-23
3.7.5	View the Data Bank Parameters in the Visual Script	3-26
3.7.6	Insert a Text Matching Test Case	3-27
3.7.7	Play Back the Script with Iterations	3-27
3.7.8	Analyzing a Playback Failure	3-28
3.7.9	Save the Script and the Results Log	3-29
3.8	Example 7: Using the Data Bank Wizard on a Registration Form	3-29
3.8.1	Recording Information in a Form	3-29
3.8.2	Inserting a Text Matching Test Case	3-29
3.8.3	Viewing the Parameters in the Visual Script	3-29
3.8.4	Using the Data Bank Wizard to Map Variables	3-30
3.8.5	Using the Data Bank Wizard to Bind to Data Source	3-31
3.8.6	View the Data Bank Parameters in the Visual Script	3-34
3.8.7	Play Back the Script with Data Iterations	3-34
3.8.8	Analyzing a Playback Failure	3-35
3.8.9	Save the Script and the Results Log	3-36

4 Job Scheduler Tutorial

4.1	Example 1: Creating a Job and Schedule	4-1	1
-----	--	-----	---

4.1.1	Starting Job Scheduler	4-1
4.1.2	Specifying the Scripts	4-1
4.1.3	Specifying the Job Notifications	4-2
4.1.4	Specifying Email Notifications	4-3
4.1.5	Scheduling the Job	4-4
4.1.6	Saving the Schedule	4-5
4.1.7	Playing Back the Job	4-5
4.1.8	Activating the Schedule	4-5
4.2	Example 2: Editing a Job	4-6
4.3	Example 3: Editing a Schedule	4-7

5 Oracle Load Testing for Web Applications Tutorial

5.1	Example 1: Performing a Simple Load Test	5-2
5.1.1	Starting Oracle Load Testing for Web Applications and Specifying the Workspace	5-2
5.1.2	Specifying a Scenario Profile	5-2
5.1.3	Running the Scenario Profile Using Autopilot	5-4
5.2	Example 2: Adding Data Sources	5-5
5.3	Example 3: Editing Data Sources	5-12
5.4	Example 4: Creating a Scenario with Multiple Profiles	5-13
5.4.1	Adding a Virtual User Profile to the Scenario	5-13
5.4.2	Saving Data for Reporting	5-13
5.4.3	Saving the Scenario	5-15
5.5	Example 5: Running Multiple Profiles	5-15
5.5.1	Running the Scenario Profiles Using Autopilot	5-15
5.5.2	Viewing Performance Statistics	5-17
5.5.3	Viewing Graphs	5-19
5.6	Example 6: Controlling Virtual Users	5-21
5.6.1	Modifying the Run Attributes	5-21
5.6.2	Viewing Virtual User Actions	5-22
5.6.3	Stopping an Individual Virtual User	5-23
5.6.4 Aborting an Individual Virtual User		5-23
5.6.5	Stopping All Virtual Users	5-23
5.6.6	Aborting All Virtual Users	5-23
5.7	Example 7: Generating Reports	5-23
5.7.1	Generating Reports from Oracle Load Testing for Web Applications	5-23
5.7.2	Opening the Chart in Microsoft Excel	5-25
5.7.3	Viewing Scenario and Session Reports	5-27
5.8	Example 8: Creating User-Defined Profiles	5-28
5.8.1	Adding Visual Scripts to the Sections Tree	5-29
5.8.2	Adding Synchronization Points to the Sections Tree	5-30
5.8.3	Moving Items in the Sections Tree	5-31
5.8.4	Editing User-Defined Profiles	5-33

List of Figures

1–1	Visual Script Tree Hierachy	. 1-2
1–2	Job Scheduler Main Window	. 1-3
1–3	Job Scheduler Wizard Window	. 1-4
1–4	Job Scheduler Schedule Window	1-4
1–5	Oracle Load Testing for Web Applications Virtual Users View	1-5
1–6	Concurrent Object Requests vs. Sequential Object Requests	
2–1	Oracle Fuctional Testing for Web Applications Main Window	
2–2	Visual Script Pane	
2–3	Browser Pane	
2–4	Playback Results Pane	
2-5	Job Scheduler Main Window	
2-6	Job Scheduler Job Pane	
2–7	Job Scheduler Results Pane	
2-8	Job Scheduler Wizard	
2–0 2–9	Job Scheduler Wizard	
2–9 2–10	Oracle Load Testing for Web Applications Main Window	
2-10	Build Scenarios Tab	
2-12	Autopilot Tab	
2–13	Watch Virtual User Grid Tab	
2–14	View Run Graphs Tab	
2–15	Create Reports Tab	
2–16	ServerStats Metric Profiles Window	
3–1	Oracle Funtional Testing for Web Applications Main Window	
3–2	Oracle Funtional Testing Visual Script Tree	
3–3	HTML Properties Dialog Box	. 3-5
3–4	Visual Script Frame Node	3-5
3–5	Script Options Dialog Box	3-6
3–6	Content Tests Manager Dialog Box	. 3-7
3–7	Images Script Node	
3–8	Resource Validation Window	3-8
3–9	Results Report Window	3-9
3–10	Playback Results Pane	
3–11	Visual Script with Failure Markers	
3–12	Different Html Script Node	
3–13	Ignore This Failure Script Node Marker	
3–14	New Links Script Node	
3–15	New Links Script Node with Marker	3-11
3–16	Different Html and Different Script Nodes	3-12
3–17	Master and Tested Page HTML Source Differences	3-12
3–17	8	3-12
3–18 3–19	Master and Tested Image Differences	
3–19 3–20	Master and Tested Image Differences	3-13
	Text Matching Test Properties Dialog Box	3-15
3–21	Text Matching Test Script Node	3-15
3–22	Server Response Test Properties Dialog Box	3-16
3–23	Server Response Test Script Node	3-16
3–24	Form Element Script Nodes	3-17
3–25	Form Element Test Properties Dialog Box	3-17
3–26	Form Element Test Script Node	3-18
3–27	Table Test Wizard Welcome Dialog Box	3-18
3–28	Select Table Element Dialog Box	3-18
3–29	Table Element Selected Example	3-19
3–30	Create Table Test Dialog Box	3-19
3–31	Create Table Test Dialog Box with Field Selected	3-20
3–32	Table Test Script Node	3-20
	-	

3–33	Parameters Script Node	3-21
3–34	Data Bank Wizard Variables Tab	3-22
3–35	Add Variable Dialog Box	
3–36	Data Bank Wizard with a Variable Added	
3–37	Data Bank Wizard Data Binding Tab	
3–38	Select Data Bank Dialog Box	
3–39	Data Bank Wizard with a Data Bank File	
3–40	Data Bank Wizard with Bindings	
3–41	Data Bank Wizard Fields and Records View	
3–42	Data Bank Parameters Script Nodes	
3–43	Iterations Dialog Box	
3–44	Parameters Script Nodes in the Script	
3–45	Data Bank Wizard Variables Tab	3-30
3–46	Data Bank Wizard with Auto-mapped Parameters	
3–47	Data Bank Wizard Data Binding Tab	
3–48	Select Data Bank Dialog Box	
3–49	Data Bank Wizard with Bound Parameters	
3–43 3–50	How Scripts Map and Bind to Data Bank Records	
3–50 3–51	Data Bank Wizard Fields and Records View	
3–52	Script Nodes with Mapped and Bound Data Bank Variables	
3–52 3–53		
3–53 4–1	Iterations Dialog Box	
	Job Scheduler Wizard Workspace and Scripts Dialog Box	
4–2	Job Scheduler Wizard Log Window Dialog Box	
4–3	Job Scheduler Wizard Email Dialog Box	
4-4	Job Scheduler Wizard Mail Server Configuration Dialog Box	
4–5	Job Scheduler Wizard Edit Schedule Dialog Box	
4–6	Job Scheduler Current Schedule Window	
4–7	Job Editor Window	
4-8	Job Editor Script Selection Window	
4–9	Job Editor Email Window	
4–10	Job Scheduler Current Schedule Window	
4–11	Edit Schedule Dialog Box	
4–12	Job Scheduler Current Schedule Window	
5–1	Oracle Load Testing for Web Applications Main Window	
5–2	Configure Parameters Pane	
5–3	Autopilot Window	
5–4	Virtual User Status Grid Window	
5–5	Add Configuration Dialog Box	
5–6	Add Monitors Step 1 Dialog Box	
5–7	Add Monitors Step 1 with Data Sources Expanded	5-7
5–8	Add Monitors Step 2 Dialog Box	5-7
5–9	Add Monitors Step 3 Dialog Box	5-8
5–10	Add Monitors Step 3 with Processors Listed	5-8
5–11	Add Monitors Step 3 with Processors Selected	5-9
5–12	Add Monitors Step 3 with Processors and Memory Selected	5-10
5–13	Configurations Dialog Box with Defined Monitors	5-11
5–14		5-11
5–15	õ	5-12
5–16		5-12
5–17		5-14
5–18		5-15
5–19		5-16
5–20		5-17
5-21		5-17
5–22		5-18

5–23	Performance Vs. Users Report	5-19
5–24	Users Vs. Time Report	5-20
5–25	Performance Vs. Time Report	5-20
5–26	Statistics Vs. Time Report	5-21
5–27	Virtual User Shortcut Menu	5-21
5–28	Modify Run Attributes Dialog Box	5-22
5–29	Virtual User Display Window	5-22
5–30	Create Reports Tab Filters	5-24
5–31	Sample Session Report Graph	5-25
5–32	Sample Excel Report Graph	5-26
5–33	Sample Excel Data Table Report	5-27
5–34	Sample Session Performance Report	5-28
5–35	Edit User Defined Profiles Dialog Box	5-29
5–36	Script Added to Run Section of User Defined Profile	5-30
5–37	Sync Point Added to Run Section of User Defined Profile	5-31
5–38	Edit User Defined Profile Dialog Box	5-32
5–39	Select Scripts and User-Defined Profiles Pane	5-32
5–40	User Defined Profile Manager Dialog Box	5-33
5–41	Edit User Defined Profile Dialog Box	5-34

Preface

Welcome to Getting Started with Oracle Application Testing Suite. This guide explains how to get started using the features and options of Oracle Functional Testing for Web Applications, Job Scheduler, and Oracle Load Testing for Web Applications for testing Web pages or applications.

Audience

This guide is for Web test engineers who will be using the Oracle Application Testing Suite applications for regression testing, performance testing (load and scalability), and monitoring of a Web site or application.

Prerequisites

The tutorials in this guide assume an understanding of software or Web application testing concepts. Test engineers using the Oracle Application Testing Suite should be familiar with the concepts of regression testing, load testing, scalability testing, and operational monitoring.

Using This Guide

This guide is organized as follows:

Chapter 1, "Introduction" provides an overview of the major features of the tools included in the Oracle Application Testing Suite.

Chapter 2, "Oracle Application Testing Suite Basics" provides descriptions of the products in the Oracle Application Testing Suite and the main features of each.

Chapter 3, "Oracle Functional Testing for Web Applications Tutorial" provides step-by-step instructions and explanations for building regression test scripts for testing Web pages or applications with Oracle Functional Testing for Web Applications. The tutorial includes examples that highlight the Visual Script features, the Data Bank Wizard, and test cases.

Chapter 4, "Job Scheduler Tutorial" provides step-by-step instructions for creating Job Scheduler jobs and schedules to play back multiple Oracle Functional Testing for Web Applications Visual Scripts for operational and performance monitoring of a site.

Chapter 5, "Oracle Load Testing for Web Applications Tutorial" provides step-by-step instruction for using multiple Oracle Functional Testing for Web Applications Visual Scripts to perform load and scalability testing of Web applications and back end systems. This chapter also explains how to configure ServerStats and generate reports from testing data.

Documentation Accessibility

Our goal is to make Oracle products, services, and supporting documentation accessible to all users, including users that are disabled. To that end, our documentation includes features that make information available to users of assistive technology. This documentation is available in HTML format, and contains markup to facilitate access by the disabled community. Accessibility standards will continue to evolve over time, and Oracle is actively engaged with other market-leading technology vendors to address technical obstacles so that our documentation can be accessible to all of our customers. For more information, visit the Oracle Accessibility Program Web site at http://www.oracle.com/accessibility/.

Accessibility of Code Examples in Documentation

Screen readers may not always correctly read the code examples in this document. The conventions for writing code require that closing braces should appear on an otherwise empty line; however, some screen readers may not always read a line of text that consists solely of a bracket or brace.

Accessibility of Links to External Web Sites in Documentation

This documentation may contain links to Web sites of other companies or organizations that Oracle does not own or control. Oracle neither evaluates nor makes any representations regarding the accessibility of these Web sites.

TTY Access to Oracle Support Services

Oracle provides dedicated Text Telephone (TTY) access to Oracle Support Services within the United States of America 24 hours a day, 7 days a week. For TTY support, call 800.446.2398. Outside the United States, call +1.407.458.2479.

Related Documents

For more information, see the following documents in the Oracle Application Testing Suite documentation set:

- Oracle Application Testing Suite Release Notes
- Oracle Functional Testing for Web Applications Functional Testing User's Guide
- Oracle Functional Testing for Web Applications Job Scheduler User's Guide
- Oracle Functional Testing for Web Applications Navigation Editor User's Guide
- Oracle Functional Testing for Web Applications Application Programming Interface Reference
- Oracle Functional Testing for Web Applications Result Objects Reference
- Oracle Load Testing for Web Applications Load Testing User's Guide
- Oracle Test Manager for Web Applications Test Manager User's Guide

Conventions

The following text conventions are used in this document:

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.

Convention	Meaning
italic	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
monospace	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

Introduction

Oracle Application Testing Suite is an integrated, comprehensive Web application testing solution that provides all the tools you need to ensure the scalability and reliability of your business-critical applications.

- Oracle Functional Testing for Web Applications for automated functional and regression testing
- Job Scheduler for scheduling functional and regression testing
- Oracle Load Testing for Web Applications for load, scalability and stress testing. Oracle Load Testing for Web Applications also includes tools for server side monitoring and reporting

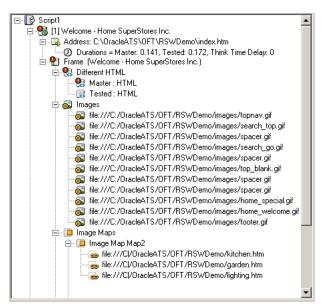
The applications are powered by a common set of Visual Scripts and require no programming. There are no proprietary languages to learn, no special proxies to set up, no training classes required. As your application changes, any differences in your tests are highlighted in the Visual Scripts, and can be automatically updated in-place. That means that your regression, and load tests will always stay synchronized with your application, and you can make automated testing a routine part of your Web development process.

This manual introduces you to the Oracle Application Testing Suite and provides step-by-step tutorials to help you get started using the tools.

1.1 About Oracle Functional Testing for Web Applications

Oracle Functional Testing for Web Applications is used for functional/regression testing and serves as a script recorder for the Oracle Application Testing Suite. Oracle Functional Testing for Web Applications records all of the objects on every page that you visit and automatically inserts tests to validate the objects. The components of each page are represented graphically in the Visual Script and can be masked or augmented using simple point and click actions.





Oracle Functional Testing for Web Applications lets you easily create, maintain, and execute regression testing scripts for your Web applications. Oracle Functional Testing for Web Applications features a powerful, intuitive visual script, an automated test case generator, a specialized text matching component, and the ability to execute data-driven tests using the Data Bank Wizard.

1.1.1 Oracle Functional Testing for Web Applications Feature Highlights

Oracle Functional Testing for Web Applications offers the following advantages for Web-based application testing:

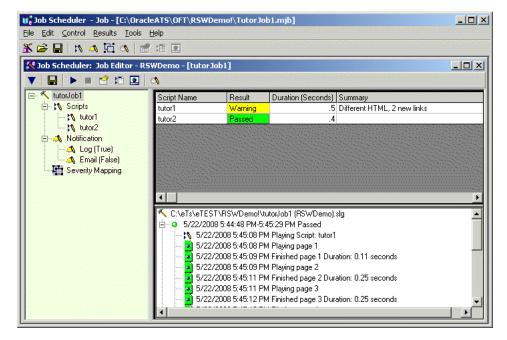
- Visual Script Technology/Automatic Test Generation you can record and test your entire application in minutes with reusable, object-oriented Visual Scripts. Recorded Visual Scripts automatically capture and test Anchors, Elements, Forms, Frames, HTML, Images, Image Maps, Links, ActiveX controls, Java Applets, VBScript, and JavaScript. Visual Scripts require no programming.
- Graphical Test Results and Simple Script Updating test failures and HTML differences are indicated by red flags annotated within the Visual Scripts for rapid diagnoses of application errors. Visual Scripts can be updated to reflect changes to the application with the click of a button.
- Data Bank Wizard create data-driven tests without programming. A single Visual Script can be used over and over with varying input and response data using values from an external data source.
- Visual Test Case Insertion additional test cases can be added to Visual Scripts to verify server response times, form elements, and the presence or absence of specific text in a page.
- Programming Interface full flexibility and extensibility to match your Web
 testing needs. Oracle Functional Testing for Web Applications provides six levels
 of testing extensibility from the simplicity of Visual Scripts to your own
 fully-customized external application that controls Oracle Functional Testing for
 Web Applications. Basic Visual Script capabilities can be extended using Oracle
 Functional Testing for Web Applications's built-in test cases or your own custom
 Test Scriptlets.

- HTTPs and SSL Support supports all popular protocols as well as certificates.
- Built-in Application Server Support automatically manages session variables for Net Dynamics, Broadvision, WebObjects, ColdFusion, and Microsoft ASP platforms.
- **High Throughput Resource Validation** automatically collects and verifies all referenced Web resources that include links and images.
- Test Case Librarian allows you to create and store re-usable test cases for use across multiple test scripts.

1.2 About Job Scheduler

Job Scheduler is a test management tool that lets you group and run multiple Oracle Functional Testing for Web Applications Visual Scripts in sequence as a single job. Job Scheduler jobs can be scheduled to run automatically at specific times or be run manually at any time.





The Job Scheduler Wizard provides a convenient way to build Job Scheduler jobs, which can then be included on any schedule. The Job Scheduler Wizard includes steps for selecting Visual Scripts and setting notification options.

-Script Select	on	Data Bank	s
Workspace:	SWDemo 💌	Add 💽 Without	
Script		C With	
tutor1			
tutor2		C All Rec	ords
tutor3 tutor4		🔤 📄 🔿 Single F	Record
•			
Script Order			
Script	Workspace	Databank Iteration	X
tutor1	RSWDemo	Without Databank	
			*

Figure 1–3 Job Scheduler Wizard Window

The Job Scheduler Schedule lets you specify when to start a job.

Figure 1–4 Job Scheduler Schedule Window

🐺 Job Scheduler: Current Sc 🔻 🛃 🖉 💷 🖉 👔		uicj						
WorkSpaces Default MavEditorTemp MyTests KutorJob1 KutorJob2 KutorJob3 Sample	12 am 1 am 2 am 3 am 5 am 6 am 7 am 8 am 9 am 10 am 11 am 12 pm 12 pm 12 pm 3 pm 4 pm 5 pm 6 pm 9 pm 10 pm	07:00 AI	10n.	Tue.	VVed	Thu.	Fri.	Sat.
Name Iterations	Failures Warnings S	Status		Last	Error	Fi	st Error	
Show Details	Clear Results	Show Loo		Show Re	hore	Show HTh	//	<u> Seconder</u>

1.2.1 Job Scheduler Feature Highlights

Job Scheduler offers the following advantages for Web-based application testing:

Multiple Oracle Functional Testing for Web Applications Visual Scripts - play back a series of Oracle Functional Testing for Web Applications Visual Scripts as a single job. Jobs can be run immediately or scheduled to run on a specific set of days and times.

Schedule Window - lets you schedule multiple jobs to run on specific days and times.

Job Scheduler Wizard - guides you through creating jobs with Visual Scripts created earlier with Oracle Functional Testing for Web Applications. The wizard provides options for customizing error notifications and e-mail recipients for playback results.

Integrated HTML Viewer - view pages in real time as Job Scheduler plays back Visual Scripts. The HTML viewer shows page content and provides visual indications of pages with failures.

Job Notification Messages - specify customized error notification messages using the Job Scheduler Wizard. The messages appear in the results log.

Job Notification e-Mail - send job results via e-mail to one or more recipients using MAPI or SMTP e-mail.

HTML Format Job Results Reports - playback results reports are saved to an HTML page for later review and analysis.

1.3 About Oracle Load Testing for Web Applications

Oracle Load Testing for Web Applications provides an easy and accurate way to test the scalability of your e-Business applications. Oracle Load Testing for Web Applications emulates thousands of virtual users accessing your site simultaneously, and measures the effect of the load on application performance.

Figure 1–5 Oracle Load Testing for Web Applications Virtual Users View

Profile	Status	Iterations Faile	d Last Run Time	Current Page	System Data Bank	Current Error Previous Erro
tutor3	Running	9	24.876	[2] Search - Home SuperStore Inc.	localhost Record 5:Phones	
tutor1	Think time delay	49	9.463	[2] Kitchens - Home SuperStores Inc.	localhost	
tutor1	Running	48	8.892	[4] Electronics - Home SuperStores Inc.	localhost	
tutor3	Iteration delay	5	31.816	[3] Results - Home SuperStore Inc.	localhost Record 4:Cabinets	
tutor1	Think time delay	28	9.594	[2] Kitchens - Home SuperStores Inc.	localhost	
tutor1	Iteration delay	29	9.053	[4] Electronics - Home SuperStores Inc.	localhost	
tutor3	Starting				localhost	
tutor1	Think time delay	2	8.513	[2] Kitchens - Home SuperStores Inc.	localhost	
tutor1	Think time delay	2	7.37	[2] Kitchens - Home SuperStores Inc.	localhost	
	tutor3 tutor1 tutor1 tutor3 tutor1 tutor1 tutor3 tutor1	tutor3 Running tutor1 Think time delay tutor1 Running tutor3 Iteration delay tutor1 Think time delay tutor1 Iteration delay tutor3 Starting tutor1 Think time delay	tutor3 Running 9 tutor1 Think time delay 49 tutor1 Running 48 tutor3 Iteration delay 5 tutor1 Think time delay 28 tutor1 Iteration delay 29 tutor3 Starting 2	tutor3 Running924.876tutor1 Think time delay499.463tutor1 Running488.892tutor3 Iteration delay531.816tutor1 Think time delay289.594tutor1 Iteration delay299.053tutor3 Starting11tutor1 Think time delay28.513	tutor3 Running924.876[2] Search - Home SuperStore Inc.tutor1 Think time delay499.463[2] Kitchens - Home SuperStores Inc.tutor1 Running488.892[4] Electronics - Home SuperStores Inc.tutor3 Iteration delay531.816[3] Results - Home SuperStore Inc.tutor1 Think time delay289.594[2] Kitchens - Home SuperStores Inc.tutor1 Iteration delay299.053[4] Electronics - Home SuperStores Inc.tutor3 Startingtutor1 Think time delay28.513[2] Kitchens - Home SuperStores Inc.	tutor3 Running924.876[2] Search - Home SuperStore Inc.localhost Record 5; Phonestutor1 Think time delay499.463[2] Kitchens - Home SuperStores Inc.localhosttutor1 Running488.892[4] Electronics - Home SuperStores Inc.localhosttutor3 Iteration delay531.816[3] Results - Home SuperStores Inc.localhost Record 4; Cabinetstutor1 Think time delay289.594[2] Kitchens - Home SuperStores Inc.localhosttutor1 Iteration delay299.053[4] Electronics - Home SuperStores Inc.localhosttutor3 Startinglocalhosttutor1 Think time delay28.513[2] Kitchens - Home SuperStores Inc.localhost

Oracle Application Testing Suite TrueLoad Technology ensures that your tests will closely correlate with real user-load so you can confidently use Oracle Load Testing for Web Applications's results to help make key decisions about your system's architecture, tuning, and hosting alternatives.

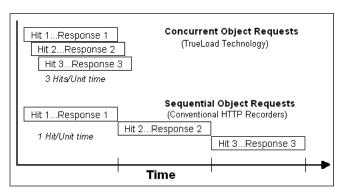


Figure 1–6 Concurrent Object Requests vs. Sequential Object Requests

1.3.1 Oracle Load Testing for Web Applications Feature Highlights

Oracle Load Testing for Web Applications offers the following advantages for Web-based application load testing:

- Trueload Technology accurately emulates multi-threaded browser requests and automatically validates server responses for test results that closely correlate with real user testing.
- Reusable Scripts uses the same Visual Scripts created for functional testing with Oracle Functional Testing for Web Applications to emulate hundreds or thousands of virtual users.
- Interactive What-If Analysis and Virtual User Display you can change the number and type of user on-the-fly to try "what-if" scenarios as you vary the loading conditions or application settings. You can even view the actual pages seen by virtual users to aid in debugging.
- Real-Time Graphs and Reports you can view real-time reports and graphs that include response time, error rates, number of users, and statistics such as hits per second, pages per second, etc.
- Single Point of Control with Distributed Agents virtual users can be simulated by a single server or distributed amongst multiple servers located anywhere on a LAN or WAN.
- Scenario Manager and Autopilot define any number of custom load scenarios by simply pointing and clicking on the names of the pre-recorded Visual Scripts and then specifying how many virtual users of each type you wish to run, and how you would like them to ramp up.
- Post-run Analysis performance data can be accumulated at varying levels of granularity including profiles, scripts, groups of pages, individual pages, and objects on pages. Oracle Load Testing for Web Applications provides a comprehensive set of graphs and reports, and can also export data to external programs such as Microsoft Excel for further analysis.
- Server-side monitoring with ServerStats server performance can be monitored for a variety of server-side application, database, system, and Web server statistics. You can configure ServerStats to display real-time performance statistics for the various hosts and services available from the server such as, percentage of CPU usage, memory usage, Web server statistics, etc.

Oracle Application Testing Suite Basics

This chapter explains how to get started using Oracle Functional Testing for Web Applications. It explains how to install and start the program, and the features of the main window.

2.1 Installing and Starting Oracle Application Testing Suite

There are two ways that you can install Oracle Application Testing Suite:

- Go to: http://www.oracle.com/technology/software/products/app-testing/index.html.
- **2.** Download the Oracle Application Testing Suite product from the Oracle Web site and save it to a temporary directory on your hard disk.
- 3. Unzip the the download file and then run oats###.exe.
- **4.** Follow the setup instructions to install the Oracle Application Testing Suite. The install program will install the WebLogic web server and the Oracle 10g Express Edition Database by default. If you chose to have the install program use an existing Oracle database, follow the instructions for using an existing database below.
- **5.** Select **Oracle Load Testing for Web Applications** from the **Oracle Application Testing Suite** start menu to start the Oracle Load Testing for Web Applications user interface.

Oracle Load Testing for Web Applications starts in an existing browser if one is available. To set Oracle Load Testing for Web Applications to always start in a new browser window, change the **Reuse windows for launching shortcuts** setting in Internet Explorer to be deselected. In IE, select **Internet Options** from the **Tools** menu then click the **Advanced** tab to change the setting under **Browsing** from the Advanced tab to access this setting.

2.1.1 Using an Existing Database

If you choose to use an Oracle Database 11*g* Enterprise Edition existing database during the installation procedure, use the following instructions to setup the database for use with the Oracle Application Testing Suite.

- **1.** Install the Oracle Database 11*g* Enterprise Edition and create a user with a minimum of the following privileges:
 - Create/Alter/Drop Tables
 - Create/Drop Views

- Create/Alter/Drop Sequences
- Create/Alter/Drop Triggers
- Create/Alter/Drop Indexes

If you are performing the database administration yourself, use the following steps:

Log Into Database Control

- 1. Install the Oracle Database 11g Enterprise Edition and launch Database Control.
- 2. Login to Database Control as a user with "system" privileges.

Create a Tablespace

- **1.** Click the **Server** tab.
- 2. Click Tablespaces.
- 3. Click Create.
- 4. On the **General** tab, give the tablespace a name.
- **5.** Under Datafiles, click **Add**. Specify a file for the new tablespace. Set the file size to 1000 MB.
- 6. Click Continue.
- 7. Click **OK** to create the new tablespace

Create a User

- 1. Return to the database instance, Server tab and click Users
- 2. Create a new user ("ATSuser"). During user creation, designate the user's default tablespace to the one created above. On the Roles tab, assign the user the RESOURCE role. On the System Privileges tab, add the required system privileges defined above to that user.
- 3. Click OK to create the new user.

2.1.2 Oracle Application Testing Suite System Setup

The following instructions explain how to setup an exsiting Oracle database for use with the Oracle Application Testing Suite.

Oracle Test Manager for Web Applications

- 1. Install the Oracle Database 11g Enterprise Edition Client OR
- 2. The Oracle Database Instant Client:
 - a. Basic Package, AND
 - **b.** ODBC Supplement

Note: Read the ODBC Release Notes for specific configuration instructions, including the need to add the Instant Client directory to the PATH environment variable and the creation of the TNS_ADMIN environment variable.

3. Create a text file called tnsnames.ora inside the folder defined by the TNS_ADMIN environment variable. The text file should contain the following text (to connect to an Oracle DB on a system with the name oracledbsys):

```
oracledbsys =
  (DESCRIPTION =
    (ADDRESS_LIST =
        (ADDRESS = (PROTOCOL = TCP)(HOST = oracledbsys.example.com)(PORT = 1521))
    )
    (CONNECT_DATA =
        (SID = ORCL)
```

- 4. Create a DSN pointing to the Oracle DB Server, as follows:
 - **a.** Open the Control Panel from the **Start** menu.
 - **b.** Open the Administrative Tools and select **Data Sources (ODBC)**
 - c. Double-click Data Sources (ODBC) and select System DSN
 - d. Click Add and select the Oracle DB driver
 - e. Click Finish to select this DB driver
 - **f.** In the "Oracle ODBC Driver Configuration" dialog box enter the following information:
 - Data Source Name: <ANY name of your choice>
 - Description: optional
 - TNS Server Name: oracledbsys (correlating to the contents of tnsnames.ora. Use the appropriate TNS name you have given.)
 - User Id: the username assigned to you on the DB server
 - Click Test, enter the password for your username and click OK
 - If the test is successful, you can proceed.

Oracle Load Testing for Web Applications

- 1. Log into OLT http://systemname:8088/olt
- 2. Select Databases from the Manage menu then select New
- **3.** Select Oracle from the drop-down list.
- 4. If you wish to create new tables in a schema, select the radio button for **Create tables in the Database**. If you wish to connect to an existing schema that has OLT tables in it, select the **User existing database** (default).
- **5.** Enter the relevant database connection information in the fields.
- 6. Click OK

Note: An SQL file is available on the Oracle Application Testing Suite system that can be used, if necessary, to delete the database components created by the product. **(This will result in loss of data!)** The file is delete tables.sql and is located in the <installdir>\OTM\Files\Oracle directory. This SQL can be executed in SQL Developer or similar utility for Oracle DB management.

2.2 Oracle Functional Testing for Web Applications Main Window Features

The Oracle Functional Testing for Web Applications main window is where you develop the Visual Scripts used for functional/regression testing, performance testing, and operational monitoring of your Web site or application. The Visual Scripts you develop using Oracle Functional Testing for Web Applications are also used by Job Scheduler, and Oracle Load Testing for Web Applications.

Visual Scripts represent a sequence of actions and tests performed on a Web site or application. Visual Scripts are used by Oracle Functional Testing for Web Applications and Job Scheduler for regression testing, Oracle Load Testing for Web Applications for performance (load and scalability) testing.

The Oracle Functional Testing for Web Applications main window consists of the menu bar, toolbar, and three panes: the Visual Script pane, Browser pane, and Playback Results Log pane.

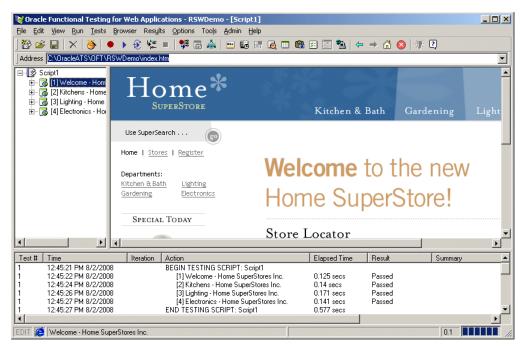


Figure 2–1 Oracle Fuctional Testing for Web Applications Main Window

The Title bar of the window shows the program name followed by the current Workspace and Visual Script name.

The Address box directly above the Web browser pane is where you enter the URL or file location of the Web page(s) to test. The bottom of the main window includes a status line.

2.2.1 Visual Script Pane

The Visual Script pane shows the tree hierarchy of recorded Web sites and pages. When you first start Oracle Functional Testing for Web Applications, the Visual Script pane is empty. When you record Web pages (either manually or using the e-Spider), Oracle Functional Testing for Web Applications creates the Visual Script tree for you.

⊟¶& Script1	
E- 6 [1] Welcome - Home SuperStores Inc.	T
Address: C:\OracleATS\OFT\RSWDemo\index.htm	
Durations = Master: 0.141, Tested: 0.172, Think Time Delay: 0	
□ □ Frame (Welcome - Home SuperStores Inc.)	
integration of the second s	
file:///C:/OracleATS/OFT/RSWDemo/images/search_top.gif	
file:///C:/OracleATS/OFT/RSW/Demo/images/spacer.gif	
file:///C:/OracleATS/OFT/RSW/Demo/images/search_go.gif	
file:///C:/OracleATS/OFT/RSW/Demo/images/spacer.gif	
file:///C:/OracleATS/OFT/RSWDemo/images/top_blank.gif	
file:///C:/OracleATS/OFT/RSW/Demo/images/spacer.gif	
file:///C:/OracleATS/OFT/RSWDemo/images/spacer.gif	
initial of the control of the con	
file:///C:/OracleATS/OFT/RSWDemo/images/home_welcome.gif	
Rile:///C:/OracleATS/OFT/RSWDemo/images/footer.gif	
	-
🚊 🛄 Image Map Map2	
ile:///CI/OracleATS/0FT/RSWDemo/kitchen.htm	
- File:///CI/OracleATS/OFT/RSWDemo/garden.htm	
ile:///Cl/OracleATS/OFT/RSW/Demo/lighting.htm	
	-

Figure 2–2 Visual Script Pane

Click the Plus icon to expand a branch or the Minus icon to collapse a branch.

The Visual Script tree will include any test cases you insert into the Visual Script. Each item in the tree is identified by an icon and a text description.

You can toggle the Visual Script pane width by selecting **Resize Visual Script View** from the **View** menu or by dragging the border between the Browser pane and the Visual Script pane.

The Visual Script uses additional icons in the tree to represent the following:

- Yellow flag: skip test.
- Red flag: test case failure.
- Red and yellow flags: ignore test failure.

2.2.2 Browser Pane

The browser pane contains a seamlessly integrated Web browser that you use to select the Web pages to test. It provides full navigation and Web access.

Figure 2–3 Browser Pane



Enter the full path and file name of the URL or local file, or drop down the list to select from recently accessed Web pages.

2.2.3 Playback Results Pane

The Playback Results pane shows a summary of the Visual Script test playback.

Figure 2–4 Playback Results Pane

Test #	Time	Iteration	Action	Elapsed Time	Result	Summary 🔺
1	10:49:39 AM 4/6/2004		[1] Welcome - Home SuperStores Inc.	0.29 secs	Passed	
1	10:49:40 AM 4/6/2004		[2] Registration - Home SuperStores Inc.	0.15 secs	Passed	
1	10:49:43 AM 4/6/2004		[3] Registered - Home SuperStores Inc.	0.17 secs	Passed	
1	10:49:47 AM 4/6/2004		Resource Validation		Passed	
1	10:49:47 AM 4/6/2004		END TESTING SCRIPT: tutor2	0.61 secs		-1
•						<u>ب</u>
EDIT 🔏	End of Playback					0.2

You can adjust the widths of the individual columns by dragging the dividers.

Icons in the Visual Script show the location of any specific failures of default tests or test cases. Resource Validation test results are listed in a separate window after playback of the script.

You can turn on and off the display of the Playback Results pane by selecting **Resize Results Window** from the **View** menu.

2.3 Job Scheduler Main Window Features

Job Scheduler is a regression testing tool used for running multiple Oracle Functional Testing for Web Applications Visual Scripts as a single job. The Job Scheduler main window is where you perform immediate or scheduled playback of a set of Oracle Functional Testing for Web Applications Visual Scripts.

The main window consists of the menu bar and toolbar. Job Scheduler has three windows that run within the main window: Current Schedule window, Current Job window, and Job Editor window.

<mark>₩ġ Job Scheduler - Job - [C:\Orac</mark> File <u>E</u> dit <u>C</u> ontrol <u>R</u> esults <u>T</u> ools		emo!\TutorJo	ob1.mjb]		
🌋 🗲 🖬 🛝 🗛 🔁 🔌 🖿	1 12 🖸				
🔀 Job Scheduler: Job Editor - R	SWDemo - [tutor]	lob1]			_ 🗆 ×
🔻 🔚 🕨 🗉 😭 🔝	4				
E- 1 tutorJob1	Script Name	Result	Duration (Seconds)		
Scripts Scripts Scripts	tutor1	Warning	.5	Different HTML, 2 new links	
tutor2	tutor2	Passed	.4		
🎦 Severity Mapping					<u>•</u>
	 ➡ ● 5/22/200 ➡ № 5/22, ➡ ● 5/22, ➡ ● 5/22, ➡ ● 5/22, ➡ ● 5/22, ➡ ■ 5/22, ➡ ■ 5/22, ➡ ■ 5/22,)8 5:44:48 PM-5 /2008 5:45:08 P /2008 5:45:08 P /2008 5:45:09 P /2008 5:45:09 P /2008 5:45:11 P /2008 5:45:11 P	tutorJob1 (RSWDemo): 45:29 PM Passed M Playing Script: tutor1 M Playing page 1 M Finished page 2 Dura M Playing page 3 M Finished page 3 Dura M Finished page 3 Dura	ation: 0.11 seconds	

Figure 2–5 Job Scheduler Main Window

You can open the Job Scheduler application from the Start menu or by selecting **Job Scheduler** from the **Tools** menu in Oracle Functional Testing for Web Applications.

2.3.1 Visual Script Job Pane

The Visual Script job pane list the Visual Scripts in an Job Scheduler job and the real-time playback results. You create Job Scheduler jobs and schedules using the Job Scheduler Wizard.

Figure 2–6 Job Scheduler Job Pane

Script Name	Result	Duration (Seconds)	Summary
tutor1	Failed	1.3	Different HTML, 2 missing links
tutor2	Passed	2.7	

2.3.2 Results Pane

The Results pane shows any log messages generated during playback of the job.

Figure 2–7 Job Scheduler Results Pane

🔨 C:\Ora	acleATS\DFT\RSWDemo!\TutorJob (RSWDemo).slg	
ė- 🔾 8/	5/2008 11:08:22 AM-11:08:44 AM Passed	
	8/5/2008 11:08:33 AM Playing Script: tutor1	
🕫	8/5/2008 11:08:33 AM Playing page 1	
🕫	8/5/2008 11:08:34 AM Finished page 1 Duration: 0.125 seconds	
🤞	8/5/2008 11:08:34 AM Playing page 2	
3	8/5/2008 11:08:36 AM Finished page 2 Duration: 0.234 seconds	1
	8/5/2008 11:08:36 AM Playing page 3	
	8/5/2008 11:08:38 AM Finished page 3 Duration: 0.266 seconds	

You can customize log messages as required using the Job Scheduler Wizard.

2.3.3 Job Scheduler Wizard

The Job Scheduler Wizard provides a convenient way to build and schedule Job Scheduler jobs. The Wizard includes steps for selecting Visual Scripts, setting notification options, and scheduling playback times.

Figure 2–8 Job Scheduler Wizard

Job Scheduler Wizard -	Workspace an	d Scripts [Step 1 of 2]			
	- Script Selecti Workspace: R		_	Add © W	Banks
	Script tutor1 tutor2 tutor3 tutor4				th All Records Single Record
	-Script Order -				
STATES IN	Script tutor1	Workspace RSWDemo)atabank Iteration Vithout Databank	× + +
Help		Cancel ABack	Next >]	

The successive steps of the Wizard provide options for setting results notifications. When the Wizard finishes, you can add the job to any schedule.

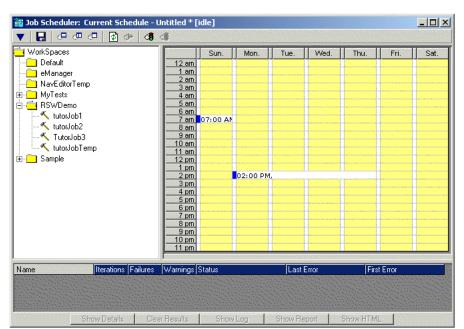


Figure 2–9 Job Scheduler Schedule Window

Schedules can be used with the current job or other saved jobs.

2.4 Oracle Load Testing for Web Applications Main Window Features

The Oracle Load Testing for Web Applications main window is where you perform the majority of your load/performance testing activities. Oracle Load Testing for Web Applications uses the same Visual Scripts that you develop using Oracle Functional Testing for Web Applications.

The main window consists of the menu bar, toolbar, and the controller tab dialogs.

Figure 2–10 Oracle Load Testing for Web Applications Main Window

Oracle Load Testing for Web Applications - Buil	Scenarios - Microsoft Intern	t Explorer				
e Edit View Favorites Tools Help						
Back 🔹 🕥 🖌 🗾 🛃 🎧 🔎 Search	숨 Favorites 🥝 🍰 🔹	🎍 🗹 • 🛄 🛍	-25			
dress 🗃 http://localhost:8088/olt/BuildScenario.do					💌 🔁 Go	Links
DRACLE Load Testing for 1	Veb Applications	irio 🖬 🛛 Session 🖬	ServerStats 🖬	Tools 👻 🛛 Manage 👻	Help 👻 Lo	gout
 Previous Session: SESSION0006 Current Scenario: <none></none> 			NARIO TONS		USER	D
Build Scenarios Set up Autopilot	Watch VU Grid	iew Run Graphs	Create Reports			
Select scripts & user-defined profiles	Configure parameters	of the scenario				
	Scripts/Profiles # VUs Sy		de Iteration Dela	y VU Pacing (Think Time)	
Repository: Default	[tutor 1 10 0	LT Server 💌 Thin Cl	ient 🔹 1	Recorded] 🔇 🙆 🕺	
Workspace: RSWDemo	tutor 2 10 0	LT Server 💽 Thin Cl	ient 🔹 1	Recorded] 🔇 🙆 🗴)
S tutor1	tutor3 10 0	LT Server 🔽 Thin Cl	ient 🔹 1	Recorded _] 🔇 🙆 🗴)
🚯 tutor2						-
🚯 tutor3 🚯 tutor4						
😵 tutor4						
Add to scenario				Add to	Autopilot]
	·					
					😔 Local intranet	

You can open the Oracle Load Testing for Web Applications application from the **Start** menu or by selecting **Oracle Load Testing for Web Applications** from the **Tools** menu in Oracle Functional Testing for Web Applications.

2.4.1 Build Scenario Tab

The Build Scenario tab is where you specify information about the virtual users to include in the load test and the attributes for each set of virtual users.

Figure 2–11	Build Scenarios	Tab
-------------	-----------------	-----

Build Scenarios Set up Autopilot	Watch VU Grid	1	View Run Graphs	Create Re	ports			
Build Scenarios Set up Autopilot Select scripts & user-defined profiles Repository: Default Workspace: RSWDemo ComDLLVBA GetAllLinksVBA Soript1 Soript2 Superstore1 tutor2 tutor2 tutor3 tutor4 tutor6			ers of the scenari	o User Mode Thin Client 💌	Iteration Delay	VU Pacing (Think Time) Recorded	() () ()	38
Add to scenario						💌 Add to /	Autopi	ilot

You can define user profiles that specify which visual scripts the users playback to emulate real users and how many virtual users to emulate.

2.4.2 Set up Autopilot Tab

The Autopilot tab is where you specify the information needed to control how the scenario starts and runs. The Autopilot controls the starting and stopping of the scenario, the rate at which new virtual users are started, and shows the total number of virtual users and the number of running virtual users.

You specify the session, start and stop times, and the virtual user rampup specifications for the Submitted Scenario Profile. It also shows the list of virtual user profiles submitted in the Oracle Load Testing for Web Applications scenario.

Figure 2–12 Autopilot Tab

Add per step After every Add per step After every 10 users © 5 seconds 10 percent C 10 iterations
Edit Configurations
r Finished System
0 e-load server

2.4.3 Watch Virtual User Grid Tab

The Watch Virtual User Grid tab lists the currently running virtual users and the profile and playback details associated with each.

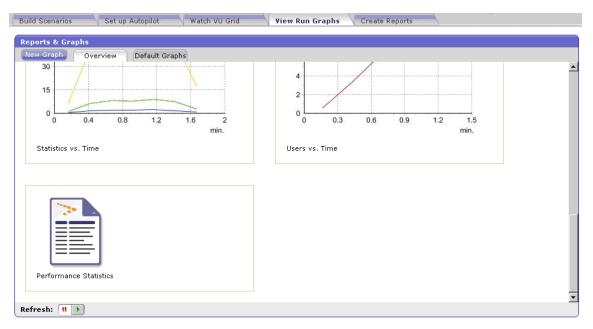
Figure 2–13 Watch Virtual User Grid Tab

1tutor1Finished57.09localhost2tutor1Finished56.649localhost3tutor1Finished56.179localhost4tutor1Finished56.149localhost5tutor1Finished56.109localhost6tutor1Finished56.249localhost7tutor1Finished56.269localhost8tutor1Finished56.669localhost9tutor1Finished56.149localhost10tutor1Finished56.089localhost
3tutor1Finished56.179localhost4tutor1Finished56.149localhost5tutor1Finished56.109localhost6tutor1Finished56.249localhost7tutor1Finished56.079localhost8tutor1Finished56.569localhost9tutor1Finished56.149localhost
4tutor1Finished56.149localhost5tutor1Finished56.109localhost6tutor1Finished56.249localhost7tutor1Finished56.079localhost8tutor1Finished56.569localhost9tutor1Finished56.149localhost
tutor1Finished56.109localhosttutor1Finished56.249localhosttutor1Finished56.079localhosttutor1Finished56.569localhosttutor1Finished56.149localhost
6 tutor1 Finished 5 6.249 localhost 7 tutor1 Finished 5 6.079 localhost 8 tutor1 Finished 5 6.569 localhost 9 tutor1 Finished 5 6.149 localhost
7 tutor1 Finished 5 6.079 localhost 8 tutor1 Finished 5 6.569 localhost 9 tutor1 Finished 5 6.149 localhost
8 tutor1 Finished 5 6.569 localhost 9 tutor1 Finished 5 6.149 localhost
8 tutor1 Finished 5 6.569 localhost 9 tutor1 Finished 5 6.149 localhost
10 tutor1 Einished 5 6-089 localbost

2.4.4 View Run Graphs Tab

The View Run Graphs tab lets you define graphs and view the graphs at run-time.

Figure 2–14 View Run Graphs Tab



You can also view the Performance Statistics report from the View Run Graphs tab. The Performance Statistics window shows a summary of the performance data for the running virtual users.

2.4.5 Create Reports Tab

The Create Reports tab lets you create post session reports and graphs.

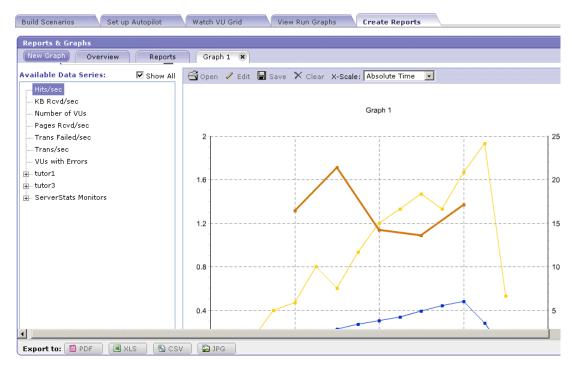
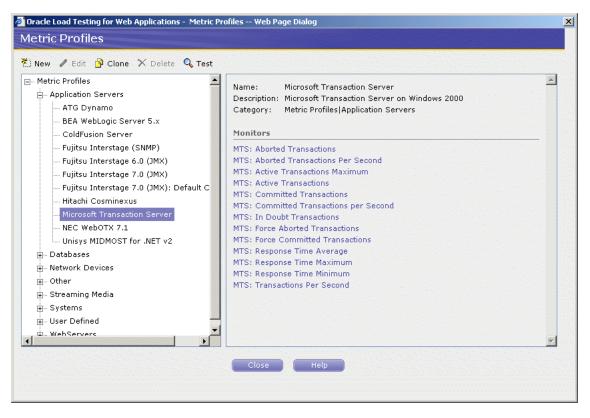


Figure 2–15 Create Reports Tab

2.4.6 ServerStats

The ServerStats component of Oracle Load Testing for Web Applications lets you monitor a variety of server-side application, database, system, and Web server statistics. You can configure ServerStats to display real-time performance statistics for the various hosts and services available from the server such as, percentage of CPU usage, memory usage, Web server statistics, etc.





You can monitor specific counters in real time using the visual indicator gauges or using graphs. In addition to performance monitoring, ServerStats let you define scripts that can log warnings or alarms if a server's counter performance goes outside a defined range.

Oracle Functional Testing for Web Applications Tutorial

This tutorial walks you through the main features of Oracle Functional Testing for Web Applications. The tutorial consists of the following examples:

- **Recording a New Visual Script** describes basic recording of Visual Scripts.
- Working with Visual Scripts describes the features and components of Visual Scripts and how to modify the default tests.
- Playing Back a Visual Script explains the procedure for playing back Visual Scripts and the option settings for playback and the results log.
- **Analyzing Test Failures** explains how to analyze the differences found between the baseline set of Web pages and a new version.
- Adding Test Cases to the Visual Script explains how to add test cases to your Visual Scripts.
- Using the Data Bank Wizard on a Search Form introduces the Data Bank Wizard and explains how to use the Data Bank Wizard to run iterative tests on a search form using data from an external file.
- Using the Data Bank Wizard on a Registration Form explains how to use the Data Bank Wizard to create automated data-driven tests.

The tutorial is designed to be followed sequentially from beginning to end. Many of the examples are interrelated and build upon the steps in previous examples.

3.1 Initializing the Tutorial

The tutorial uses two versions of web pages to demonstrate the capabilities of Oracle Functional Testing for Web Applications. To make sure the initial version of the tutorial web pages is the current version, do the following:

- 1. Select **Programs** from the **Start** menu and then select **Build A Home Superstores** from the **Oracle Application Testing Suite** submenu. A command window will appear briefly as the batch file copies the Build A files.
- 2. Close the command window, if necessary.

3.2 Example 1: Recording a New Visual Script

This example illustrates the creation and recording of a Visual Script.

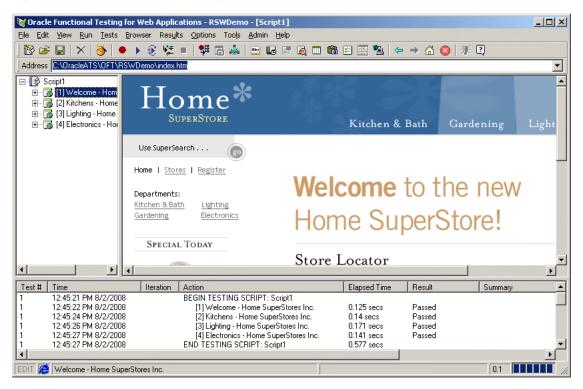
3.2.1 Start Oracle Functional Testing for Web Applications

- Select Programs from the Start menu and then select Oracle Functional Testing for Web Applications from the Oracle Application Testing Suite submenu to start Oracle Functional Testing for Web Applications.
- **2.** Select **Open Workspace** from the **File** menu, select RSWDemo as the Workspace, and click **OK** to get to the main window.

Note: If you installed this version of the Oracle Application Testing Suite over a previous version, your default installation directory will still be the old directory. If this version is the first time you installed the Oracle Application Testing Suite, the default installation directory is c:\OracleATS\OFT.

3. Type c:\OracleATS\OFT\rswdemo\index.htm in the URL drop down list and then press ENTER. (The tutorial assumes that you installed Oracle Functional Testing for Web Applications in the default c:\OracleATS\OFT directory. If you installed to another directory, enter the appropriate path.)

Figure 3–1 Oracle Funtional Testing for Web Applications Main Window



Oracle Functional Testing for Web Applications opens the Home Superstores tutorial Web page into the Browser pane.

3.2.2 Start a Recording

4. Select **New Script** from the **File** menu and select **No** if asked to save changes to Script1.

5. Click the Record button on the toolbar. Oracle Functional Testing for Web Applications is now recording your actions as indicated by the REC in the status bar. The [1] Welcome - Home Superstores Inc. title is recorded into the Visual Script pane.

3.2.3 Navigate the Web Site

- 6. Click on the <u>Kitchen and Bath</u> link on this page. The Kitchen and Bath page appears in the Browser pane and the address should show c:\OracleATS\OFT\rswdemo\kitchen.htm. You should now see [2] Kitchens Home Superstores Inc. in the Visual Script.
- **7.** Click on the <u>Lighting</u> link on this page. The Lightings page appears in the Browser pane and the address should show c:\OracleATS\OFT\rswdemo\lighting.htm. You should now see [3] Lightings Home Superstores Inc. in the Visual Script.
- **8.** Click on the <u>Electronics</u> link on this page. The Home Electronics page appears in the Browser pane and the address should show c:\OracleATS\OFT\rswdemo\etronics.htm. You should now see [4] Electronics Home Superstores Inc. in the Visual Script.

3.2.4 Stop the Recording

9. Click the Stop button on the toolbar to stop the recording. The Visual Script pane should list four pages in the script.

3.2.5 Save the Script

H

- **10.** Select **Save Script** from the **File** menu to save the script. The autonaming feature initially defaults the name of a new script to Script1.
- 11. Type in tutor1 for the name of the script in the Save As dialog box and click **Save**.

3.3 Example 2: Working with Visual Scripts

This example explains the features of the Visual Script tree and how to examine the structure and content of a Web page. It also explains how to modify the built-in Oracle Functional Testing for Web Applications tests.

Before starting this example, make sure the Visual Script that you recorded in Example 1 is still displayed.

- **1.** Select **Resize Visual Script View** from the **View** menu to expand the Visual Script pane.
- **2.** Click the [1] Welcome Home Superstores Inc. node in the Visual Script and then click the Plus next to the node. The script shows the Address and Frame nodes to the page tree.
- **3.** Click the right mouse button and select **Expand Page** to show the entire page tree, which should look similar to the following:

🖃 🕼 Script1	
😑 🖓 🚺 Welcome - Home SuperStores Inc.	
🚊 🗔 Address: C:\OracleATS\OFT\RS\WDemo\index.htm	
💭 Durations = Master: 0.141, Tested: 0.172, Think Time Delay: 0	
Frame (Welcome - Home SuperStores Inc.)	
file:///C:/OracleATS/OFT/RSWDemo/images/topnav.gif	
file:///C:/OracleATS/OFT/RSWDemo/images/search_top.gif	
file:///C:/OracleATS/OFT/RSWDemo/images/spacer.gif	
ile:///C:/OracleATS/OFT/RSWDemo/images/search_go.gif	
ilie:///C:/OracleATS/OFT/RSWDemo/images/spacer.gif	
ilie:///C:/OracleATS/OFT/RSWDemo/images/top_blank.gif	
ilie:///C:/OracleATS/OFT/RSWDemo/images/spacer.gif	
ine.///c./oracleATS/OFT/AS/WDemo/images/space.gil ine.///c./OracleATS/OFT/AS/WDemo/images/space.gif	
File:///C:/OracleATS/OFT/RSW/Demo/images/home_welcome.gif	
file:///C:/OracleATS/OFT/RSW/Demo/images/footer.gif	
🗄 🖳 Image Maps	
🖻 🖳 Image Map Map2	
🛁 🧓 file:///CJ/OracleATS/OFT/RSW/Demo/kitchen.htm	
👝 file:///C /OracleATS/OFT/RSWDemo/garden.htm	
🛁 🧓 file:///Cl/OracleATS/OFT/RSWDemo/lighting.htm	
	•

Figure 3–2 Oracle Funtional Testing Visual Script Tree

The [1] Welcome - Home Superstores Inc. page contains the following nodes in the tree:

- An Address node that has the recorded URL for the page.
- A Duration node under the Address node that displays how long it took to download the page and the think time delay associated with the page. The think time delay is the actual amount of time the user spent on the page before going to another page. For additional information about think time delay, see the online help.
- A Frame node that is the main frame for the entire page. Below this node are the other constituents of the page.
- An HTML node that has the HTML source for the page.
- An Images node that has all the images in the page under it.
- A Image Maps node that has all image maps in the page under it.
- A Scripts node that has all the VBScripts and JavaScripts under it.
- A Links node that has all the links in the page under it.

Note: Web pages that include Frame Sets, Anchors, Forms, Elements, Active X objects, Java Applets, etc. will have additional tree nodes displayed in a similar fashion.

3.3.1 Viewing Information About a Visual Script Item

- 4. Select the HTML node in the Visual Script.
- **5.** Click the right mouse button and select **HTML Properties**. The following dialog box is displayed:

	Properties	ОК	
Page:	[6] Registration - Home SuperStores Inc.	Cancel	
Туре:	HTML		
Name:	<none></none>	<u>R</u> eset	
	Execute Default Comparison Test Result: Passed	<u>H</u> elp	
HTML S	purce	Validate	
<br function { }	T language=JavaScript> register(url, date, fname, Iname, mail, phone) if (navigator.appName == "Microsoft Internet Explorer") event.returnValue = false location = url + "#" + escape(date) + "#" + escape(fname) + "#" + escape(Iname) + return false		
function {	Initialize() var theSystemDate = new Date(); var todaysDate = ([theSystemDate.getMonth()+1) + '/' + theSystemDate.getDate() document.theForm.TodaysDate value = todaysDate; document.theForm.MailList.checked = true;		
7.] (You can test for the presence or absence of certain text in the HTML on this page by adding a Text Matching Test to this page of the Visual Script. See "Adding a Text Matching Test" in the electronic help file for details.		

Figure 3–3 HTML Properties Dialog Box

You can get more information about any item in the Visual Script using the Properties option. The properties for the different items vary.

- 6. Close the Properties dialog box.
- 7. Repeat steps 5 and 6 with any other items in the tree and view the properties.

3.3.2 Turning Automatic Testing On and Off

- **8.** Select the Frame node in the Visual Script.
- **9.** Click the right mouse button and select **Don't Test Frame**. Notice a small yellow circle appears next to the Frame node to indicate that the automatic existence test for the frame is turned off, as shown below:

Figure 3–4 Visual Script Frame Node

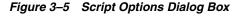
Frame (Welcome - Home SuperStores Inc.)

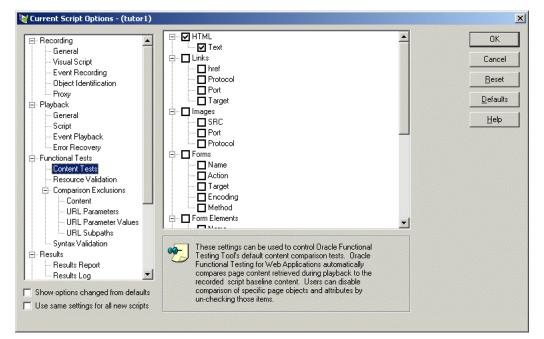
10. Click the right mouse button and select **Test Frame**. The yellow circle disappears to indicate that the automatic existence test is activated.

Note: Oracle Functional Testing for Web Applications maximizes your productivity by virtually eliminating the need to program test scripts. When a Visual Script is recorded, it captures your interaction with the Web application under test. A series of default test cases are automatically generated and added to the Visual Script. These tests are designed for Images, Links, Frames, Forms, Elements, HTML, Java Applets, ImageMaps, and Active-X controls and can be customized to suit your requirements.

3.3.3 Modifying Default Tests

11. Select **Current Script (tutor1)** from the **Options** menu and then select **Content Tests** in the **Functional Tests** section.





This dialog box indicates which comparison tests will be performed on which page attributes as default tests for the current script. You can enable or disable testing of specific attributes by selecting or clearing the appropriate check box(es). You can set global defaults for all Visual Scripts by selecting **New Scripts (Global)** from the **Options** menu and then selecting **Content Tests** in the **Functional Tests** section.

- **12.** Make sure the following check boxes are selected:
 - HTML.
 - Images.
 - Scripts.
 - Links.
 - Frames.
- 13. Click OK.
- 14. Select the [1] Welcome Home Superstores Inc. node in the Visual Script.
- **15.** Click the right mouse button and select **Page Content Tests Manager**. The Content Tests Manager dialog box opens.

ge: Welcome - Home SuperStores Inc.	OK
	Cancel
∃ □ Links □ href	<u> </u>
Protocol	Defaults
Target	
Images	
	🖵 🗖 Show Cha
Port	
Protocol	
E Forms	
🗖 Name	
🗖 Action	
🗖 Target	
🗖 Encoding	
Method	•

Figure 3–6 Content Tests Manager Dialog Box

This dialog box indicates which comparison tests will be performed on which page attributes as default test cases for a specific page. You can enable or disable testing of specific page attributes by selecting or clearing the appropriate check box(es), and clicking the **OK** button.

16. Clear the Images check box, and press the **OK** button. A yellow circle appears next to the Images node in the Visual Script for the image collection, as shown below:

Figure 3–7 Images Script Node



When the Visual Script is played back, all Image tests will be ignored.

3.4 Example 3: Playing Back a Visual Script

This example explains the procedure for playing back Visual Scripts that you have recorded. It also shows the option settings for playback and the results log.

- Select Current Script from the Options menu and then select Results Log in the Results section. Make sure the Append to log and All details radio buttons are selected.
- Select Results Report in the Results section and make sure the Automatically create report after script playback option is selected.
- Select General in the Playback section, make sure the Automatically Run Resource Validation After Playback check box is selected, and then click OK.
- **4.** Select **Resize Visual Script View** from the **View** menu to expand the Browser pane.

5. Click the Playback Script button to play back the recorded script. The pages in the script will be played back in the order recorded. The Browser navigates to each page, executes the default tests for each page, and shows the results visually in the script. At the end of the play back, Oracle Functional Testing for Web Applications runs a Resource Validation test and shows the results.

The Resource Validation test checks the integrity of the referenced resources (i.e. links, images, etc.) in your pages.

Figure 3–8	Resource	Validation	Window
------------	----------	------------	--------

N Ho:	st Resource	Status	Error	Internet S	Stats
📓 1 🛛 Loc	cal Host -	Finished		-	
1	C:\OracleATS\OFT\RSWDemo\index.ht	m Passed		200	
🕄 1	C:/OracleATS/OFT/RSWDemo/images/	'topn Passed		200	
🕄 1	C:/OracleATS/OFT/RSWDemo/images/	'sear Passed		200	
🕄 1	C:/OracleATS/OFT/RSWDemo/images/	'spac Passed		200	
🕄 1	C:/OracleATS/OFT/RSWDemo/images/	'sear Passed		200	
🕄 1	C:/OracleATS/OFT/RSWDemo/images/	'top Passed		200	
🕄 1	C:/OracleATS/OFT/RSWDemo/images/	'hom Passed		200	
🕄 1	C:/OracleATS/OFT/RSWDemo/images/	'hom Passed		200	1
🕄 1	C:/OracleATS/OFT/RSWDemo/images/	foote Passed		200	
N NEWSCON	<u> </u>				►

The "passed" results indicate that all referenced resources are available.

6. Click the **Close** button to close the Resource Validation window.

Oracle Functional Testing for Web Applications automatically generates a results report and opens the report in a new browser window:

•

Figure 3–9 Results Report Window

Back 🔻 🕑		es 🥝 🍰 🦉 .			
:ss 🙋 C:\Ora	cleATS\OFT\RSWDemo!\tutor1Results\tutor1Result.	htm		<u> </u>	🔁 Go 🛛 Lir
help protect y	our security, Internet Explorer has restricted this file	from showing active content that	could access your computer. C	lick here for options.	
ript Nam	e: tutor1				
pt Name : tuto :kspace: RSW	Demo				
e & Time: 8/2 rations: 1 al Pages: 4 al Tests: 129 al Failures: 0 al Warnings: 1 rall Result: @ Script Su	(0.00%) 0 (0.00%)) Passed				
ations: 1 al Pages: 4 al Tests: 129 al Failures: 0 al Warnings: 1 arall Result: 6	(0.00%) 0 (0.00%)) Passed	Recorded Time (sec)	Playback Time (sec)	Result	Summary
ations: 1 al Pages: 4 al Tests: 129 al Failures: 0 al Warnings: 1 rall Result: Script Su	(0.00%) 0 (0.00%)) Passed Immary	Recorded Time (sec) 0.562	Playback Time (sec) 0.594	Result @Passed	Summary
ations: 1 al Pages: 4 al Tests: 129 al Failures: 0 al Warnings: 1 rall Result: Script Su Iteration	(0.00%) 0 (0.00%) Passed Immary Page				Summary
ations: 1 al Pages: 4 al Tests: 129 al Failures: 0 al Warnings: 1 rall Result: Script Su Iteration	(0.00%) o (0.00%) Passed Immary Page Iteration Total (sec)	0.562	0.594	Passed	Summary
ations: 1 al Pages: 4 al Tests: 129 al Failures: 0 al Warnings: 1 rall Result: Script Su Iteration	(0.00%) o (0.00%) Passed Immary Page Iteration Total (sec) [1] Welcome - Home SuperStores Inc.	0.562 0.172	0.594 0.250	PassedPassed	Summary
ations: 1 al Pages: 4 al Tests: 129 al Failures: 0 al Warnings: 1 rall Result: Script Su Iteration	(0.00%) 0 (0.00%) Passed Immary Page Iteration Total (sec) [1] Welcome - Home SuperStores Inc. [2] Kitchens - Home SuperStores Inc.	0.562 0.172 0.187	0.594 0.250 0.141	 Passed Passed Passed 	Summary

Note: The Results Report uses active content. If the links in the report are not working, check if the browser shows the restricted active content security warning at the top of the browser. If so, click on the warning and select **Allow Blocked Content**. Use the browser Internet Options to set the default settings for allowing active content. Select **Browser** from the **Options** menu and click the **Advanced** tab. Select the **Allow active content to run in files...** setting under the Security section if you do not want the restricted active content security warning to appear each time the Results Report is generated.

7. Click on the page names in the Script Summary section to view the information for each page.

Notice that all tests passed. This is because you played back the script using the same version of the Web pages that was used to record the script. This establishes a baseline of tests for the Web application or Web site's content and structure.

8. Close the browser window when finished with the report.

The Results pane also shows a summary of the playback actions.

Figure 3–10 Playback Results Pane

Test #	Time	Iteration	Action	Elapsed Time	Result	Summary	
1 1 1 1 1	11:52:37 AM 4/5/2004 11:52:39 AM 4/5/2004 11:52:40 AM 4/5/2004 11:52:41 AM 4/5/2004 11:54:15 AM 4/5/2004 11:54:15 AM 4/5/2004		 Welcome - Home SuperStores Inc. Kitchens - Home SuperStores Inc. Lighting - Home SuperStores Inc. Electronics - Home SuperStores Inc. Resource Validation END TESTING SCRIPT: tutor1 	0.28 secs 0.2 secs 0.2 secs 0.17 secs 0.85 secs	Passed Passed Passed Passed Passed		
	End of Playback]	0.2	

In the next example, you'll see how playback and the results are affected by Web page changes.

3.5 Example 4: Analyzing Test Failures

This example explains how to analyze the differences found between the baseline Web pages and a new version with changes. The Oracle Functional Testing for Web Applications tutorial includes a batch file that copies a new version of three of the pages that you recorded in Example 1.

- 1. Select **Programs** from the **Start** menu and then select **Build B Home Superstores** from the **Oracle Application Testing Suite** submenu. This runs a batch file that updates the Home Superstores page to a new version.
- **2.** If necessary, close the command window after the batch file finishes copying the files.
- **3.** Click the Playback Script button to play back the recorded script again. The pages are played back in the order recorded.

The resource validation test found one resource that failed. There is one failed image called moviei.gif. The results of the resource validation test are displayed in the output log window and the failure will be displayed under the appropriate pages with red flags.

- **4.** Click the **Close** button to close the Resource Validation window and close the Results Report browser window.
- 5. Notice the red circles next to the pages in the Visual Script, as shown below:

Figure 3–11 Visual Script with Failure Markers



- 🗄 🏰 [1] Welcome Home SuperStores Inc.
- 🗄 🏰 [2] Kitchens Home SuperStores Inc.
- 🗄 🔚 🐻 [3] Lighting Home SuperStores Inc.
- 🛨 🥵 [4] Electronics Home SuperStores Inc.

Note: Oracle Functional Testing for Web Applications displays errors and problems encountered during playback using simple color-coded circles in the script itself. Test results are displayed dynamically in the Visual Script tree as the script is being played back. You can double-click on any error event in the result log pane to advance to the corresponding page in the script. Errors encountered upon Visual Script playback can be rejected, ignored, or can be accepted to create a modified baseline script.

6. Select **Find Next Failure** from the **Results** menu, or press the F3 key to locate the next failure in the Visual Script. The first page [1] Welcome - Home Superstores Inc. has two places where differences are indicated: Different HTML and New Links with two new links under the Tested node. The new links are admlist.htm and adminfo.htm.

3.5.1 Ignoring Failures

Occasionally, you may want to ignore a known problem or discrepancy that does not affect the overall test being performed.

7. Locate the Master: Html node under the Different Html node.

Figure 3–12 Different Html Script Node



8. Click on the Different Html node, and then click the right mouse button and select **Ignore This Failure**. This adds a yellow circle to the Different Html node to indicate that the failure caused by the HTML change should be ignored, as follows:

Figure 3–13 Ignore This Failure Script Node Marker



9. Press the F3 key to get to the New Links item and expand the Tested node to view the new links.

Figure 3–14 New Links Script Node



10. Click on the New Links item, and then click the right mouse button and select **Ignore This Failure**. This adds a yellow circle to the New Links node to indicate that the failure caused by the presence of new links should be ignored.

Figure 3–15 New Links Script Node with Marker



3.5.2 Accepting Changes Shown in the Script

Often, you will get new versions of Web pages that you want to use as the new baseline for testing.

11. Follow the red circles in the [2] Kitchens - Home Superstores Inc. page.

The problems on this page are indicated by the presence of two nodes called Different Html and Different Script. Below the Different Html node are the recorded and tested versions of the HTML for the current page. Below the Different Script node are the recorded and tested versions of the JavaScript function that has changed for the current page.

Figure 3–16 Different Html and Different Script Nodes



- **12.** Double-click on the Different HTML node and a dialog box with the differences for the HTML opens.
- **13.** Click **Next Difference** as many times as necessary to locate the following differences between Master and Tested page HTML source.

Figure 3–17 Master and Tested Page HTML Source Differences

<TD_vAlign=top>Departments:
Kitchen & Bath
Gardening</TD> <TD_vAlign=top>Other Departments:
Kitchen & Bath
Gardening</TD>

Notice that the changes are textual changes to the Web page content. The Master text (baseline for testing) is shown in blue. The tested text (new version) is shown in red.

- 14. Click the Cancel button to close the window.
- 15. Repeat the same process for the Different Script node.
- **16.** Select the [2] Kitchens Home Superstores Inc. page node and select **Accept Tested Page** from the right-click shortcut menu. The red circles in the Visual Script for that page disappear. The accepted change becomes the new baseline for future testing of this page.

3.5.3 Rejecting Problems Shown in the Script

There may be times when you do not want a change to a Web page to be accepted as the new baseline for testing.

17. Follow the red circles in the [4] Electronics - Home Superstores Inc. page.

The problems on this page are indicated by the presence of two nodes called Different Html and Different Image. Below the Different Html node are the

recorded and tested versions of the HTML for the current page. Below the Different Image node are the recorded and tested versions of the image found on this page.

18. If necessary, expand the Different Image node to display the Master and Tested Image nodes. You will notice that the file name of the image file is spelled differently [movie vs. movei]

Figure 3–18 Master and Tested Image Differences

🖻 👷 Different Image

--- 😫 Master : file:///C:/OracleATS/OFT/RSWDemo/images/movie.jpg

---🕵 Tested : file:///C:/OracleATS/OFT/RSWDemo/images/movei.jpg

The Master node has a red circle because playback revealed that the link to the image is no longer in the HTML. The Tested node has a red flag because the image failed the Resource Validation test.

19. Double click on the Different HTML node and you will see the same spelling change in the HTML source for the page.

Figure 3–19 Master and Tested Image Differences

```
<TD><IMG height=140 src="images/movie.jpg" width=164></TD>
<TD><IMG height=140 src="images/movei.jpg" width=164></TD>
```

We wish to discard these differences and continue to use the original spelling as the baseline for testing in the Visual Script.

20. Click the **Cancel** button.

H

- **21.** Select the [4] Electronics Home Superstores Inc. page node and select **Discard Tested Page** from the right-click shortcut menu. This causes the originally recorded baseline to be left intact. The baseline differences will be discarded and all red circles will be removed.
- **22.** Click the Playback toolbar button to play back the script again. Oracle Functional Testing for Web Applications still flags the Different HTML and the image name spelling problem in the [4] Electronics Home Superstores Inc. page of the Visual Script.
- 23. Close the Resource Validation window and the Results Report.
- 24. Select Save Output Log As from the File menu.
- **25.** Enter the name tutor1.log and click Save.
- 26. Select Save Script from the File menu to save the changed Visual Script.
- 27. Select Clear Results Window from the View menu to clear the results log pane.

3.6 Example 5: Adding Test Cases to the Visual Script

This example explains how to add four types of test cases to your Visual Scripts. In addition to the automatic existence and resource validation tests, Oracle Functional Testing for Web Applications provides the ability to add the following test cases to the pages in your Visual Script:

- Text Matching
- Server Response

- Form Element
- Table Test
- WinForms Test
- Siebel Test

3.6.1 Record a New Script

- 1. Select **Programs** from the **Start** menu and then select **Build A Home Superstores** from the **Oracle Application Testing Suite** submenu. This batch file restores the original Web pages for the Home Superstores site.
- 2. If necessary, close the command window.
- 3. Select New Script from the File menu to create a new Visual Script.
- **4.** Reload the c:\OracleATS\OFT\rswdemo\index.htm page in the Browser pane by selecting it from the Browser drop-down list.
- 1

*****8

- 5. Click the Record button on the toolbar.
- **6.** Click on the Register link in the Browser pane. The Registration page appears in the Browser pane and the address should show c:\OracleATS\OFT\rswdemo\register.htm.
- **7.** Type Admin as the first name, enter any email address, and phone in the text area, and click the **submit my entry** button. The Browser returns the Database Authorization and Administration options of he registration page (regres.htm).

3.6.2 Stop the Recording

8.

-

Apc

Click the Stop button on the toolbar to stop the recording. The Visual Script pane should list three pages in the script.

3.6.3 Insert a Text Matching Test Case

Text Matching test cases compare selected text from a Web page to the text you specify in the test case.

- 1. Select [3] Registered Home Superstores, Inc. in the Visual Script.
- **2.** Click the right mouse button and select **Goto Page** to open the Registered Home Superstores, Inc page in the Browser.
- **3.** Scroll the Browser pane so that the text "Database Authorization and Administration" is visible.
- 4. Highlight the "Database Authorization and Administration" text with the mouse.
- 5. Click the Insert Text Matching Test Case button on the toolbar.

Oracle Functional Testing for Web Applications captures the highlighted text and opens the Insert Text Matching Test Case dialog box.

2	perties		OK
Page:	3]Registered - Home SuperStores Inc.		Cancel
Туре: 🛐	ext Matching Test		Cancer
🔽 Execu	te Test Result: Passed		<u>R</u> eset
Text Match	ing Test		<u>H</u> elp
Name:	tutortext1	Select From	
Pass when	Selected text is present	Test Library	
Selected T	ext	Add to Test Library	
Database	Authorization and Administration		
		and the second	
		and the second second	
Test on I	HTML Result: Passed		
	ding a Text Matching Test to the test library al	lows you to	

Figure 3–20 Text Matching Test Properties Dialog Box

- **6.** Type tutortext1 as the test case name.
- 7. Make sure the Pass when: option is set to Selected text is present.
- **8.** Click **OK** and view the test case in the Visual Script. Oracle Functional Testing for Web Applications adds the test case to the Visual Script under the Frame node.

Figure 3–21 Text Matching Test Script Node



3.6.4 Insert a Server Response Test Case

Server Response test cases measure the response time of a server access for a page in the Visual Script.

- **9.** Select the [2] Registration Home Superstores, Inc. item in the Visual Script.
- **10.** Click the Insert Server Response test case button on the toolbar. Oracle Functional Testing for Web Applications opens the Insert Server Response Test Case dialog box.



🔆 Server Response Test Properties	X			
Server Response Test Properties General Properties Page: [[2] Registration - Home SuperStores Inc. Type: Server Response Test Execute Test Result: Passed Specify Name and Page Information	OK <u>C</u> ancel <u>R</u> eset <u>H</u> elp			
Name: Image:				
Elapsed Time During Playback: 0 seconds You may use this test to time multiple pages of your Visual Script by inserting separate Start and End test nodes in the respective pages. The Start and End nodes of a Server Response Test must have the same name.				

Figure 3–22 Server Response Test Properties Dialog Box

- **11.** Type TutorTimer1 as the test case name.
- **12.** Set the **Maximum Time Allowed for Playback** option to 5 seconds. Leave the **Minimum Time** at 0 seconds.
- **13.** Click **OK** and view the test case in the Visual Script. Oracle Functional Testing for Web Applications adds the test case to the Visual Script between the Address and Frame nodes.

Figure 3–23 Server Response Test Script Node

🖻 🐻 [2] Registration - Home SuperStores Inc.

- 🗉 🗄 🖳 🛃 Address#: C:\OracleATS\OFT\RSWDemo\register.htm (blur + window(index=0).link)
- 🔤 🔀 Server Response Test:TutorTimer1, Minimum: 0, 🛛 Maximum: 5

3.6.5 Insert a Form Element Test Case

Form Element test cases compare attributes and values of the elements in an HTML form.

- Select Current Script from the Options menu and then select Content Tests in the Functional Test section. Make sure the Forms and Form Elements check box are selected and click OK.
- 15. Select the [2] Registration Home Superstores, Inc. item in the Visual Script.
- **16.** Expand the page and select the TodaysDate element of the regres.htm form.

Figure 3–24 Form Element Script Nodes

🖻 📲 🐻 Forms

E-B Form theForm get regres.htm application/x-www-form-urlencoded

- 💾 Type : text, Name : TodaysDate, Value : 8/2/2008, Disabled : False
- 📳 Type : text, Name : FirstName, Value : , Disabled : False -
- 🖶 Type : text, Name : LastName, Value : , Disabled : False
- -- 📳 Type : text, Name : Mail, Value : , Disabled : False
- -- 🔡 Type : text, Name : Phone, Value : , Disabled : False
- ---- 🔡 Type : checkbox, Name : MailList, Value : YES, Disabled : False, Checked : True
- **17.** Select **Insert Form Element Test** from the **Tests** menu. Oracle Functional Testing for Web Applications opens the Form Element Test dialog box.

Figure 3–25 Form Element Test Properties Dialog Box

쓚 Form Element Test Properties	×
General Properties	ОК
Page: [2]Registration - Home SuperStores Inc.	
Type: Form Element Test	<u>C</u> ancel
Execute Test Result: Knone>	<u>R</u> eset
Form Element Test Properties	<u>H</u> elp
Name:	
Specify Comparison Conditions	
Select Property: Current Value:	
Type 🔽 "text"	
Select Test Criteria: Property Must Satisfy Expression Specify Expression:	
Use this test criteria to specify an expression as follows: Type = "text"	
Test Expression	
Test Result:	
You may modify the VBA expression used to test this form element by editing the "Expression" field above. Before clicking DK, click "Test" to ensure that your new expression is valid.	

- **18.** Type TodaysDate as the test case name.
- **19.** Set the **Select Property** option to Value.
- 20. Set the Select Test Criteria to Property Must Satisfy Expression.
- **21.** Change the **Specify Expression** field to Value = Date.
- **22.** Click the **Test** button. Oracle Functional Testing for Web Applications should return True in the **Result** field.
- **23.** Click **OK** and view the test case in the Visual Script. Oracle Functional Testing for Web Applications adds the test case to the Visual Script under the Form Element node.

Figure 3–26 Form Element Test Script Node

🖻 📲 🐻 Forms
🚊 📆 Form theForm get regres.htm application/x-www-form-urlencoded
🚊 📲 Type : text, Name : TodaysDate, Value : 8/2/2008, Disabled : False
🔤 🖶 Form Element Test, Name: TodaysDate,Value=Date
📴 Type : text, Name : Mail, Value : , Disabled : False
🛄 📴 Type : checkbox, Name : MailList, Value : YES, Disabled : False, Checked : True

3.6.6 Insert Table Test Test Case

Table test test cases let you define a custom test on a Web page table object. The Table Test Wizard lets you select the table object directly from the Web page by highlighting it with the mouse. The wizard also lets you specify the table object property to test and the type of test to perform.

- 24. Select [1] Welcome Home Superstores, Inc. in the Visual Script.
- **25.** Click the right mouse button and select **Goto Page** to open the Welcome Home Superstores, Inc page in the Browser.
- **26.** Select **Insert Table Test** from the **Tests** menu. Oracle Functional Testing for Web Applications opens the Table Test Wizard.

ew Tabl	e Test Wizard Step 1 out of 3: Name your test.
	Welcome to the Table Test Wizard
st Name:	

Figure 3–27 Table Test Wizard Welcome Dialog Box

27. Type Depts as the test case name.

Cancel

Click Next to select the table to test.

28. Click Next. The Select Table Element dialog box is displayed.

Next>

Figure 3–28 Select Table Element Dialog Box

New Table Test V	Wizard Step 2 out of 3: Select Table Ele 🖄
Selected Path:	
Current Path:	window(index=0).Tag(TABLE)(index=2)
Cancel	<back next=""> Finish</back>

29. Moving the cursor over the web page highlights the tables on the page. Highlight the Departments area in the upper left-hand corner of the page.

Figure 3–29 Table Element Selected Example



30. Select the table. The path is displayed in the Select Table Element dialog box. Click Next to display the Create your tests dialog box.

Figure 3–30 Create Table Test Dialog Box

New	v Table T	est Wizard	Step 3 ou	t of 3: Crea	te your to	ests.	
Re	corded T	able					
	A		В	C		D	
1	V		✔Home				
2	Image: Image: Image: Image: Amage: Image: Image	/C:/Oracle/	🗹 Departme	ents: 🗹 f	ile:///C:/0	racle/ 🗹	
•							
[able	Path:	window(inde	x=0).Tag[TA	BLE](index=	3)		✓ Enabled
Fest E	Element:	A1					
[est]	Туре:	String				-	
- Strir	ng						
						_	
	itches the I Exact		WildCard	C Por	gular Expres		
<u> </u>	Exact	L		C Lei	julai Expre:	ssion	
	t Table Te						
		si			Ies		
He	sult:				<u></u>		
De	tails:						
		C.	ncel	< <u>B</u> ack	Next		Finish

31. This dialog box displays the table in a spreadsheet format, showing the data that is in each table cell. The dialog box options change depending on the type of data in the cell. Cells that are checked will be tested. Select cell B2.

est Element: B2 est Type: String						1 Table	corded	Re
2 ♥ file:///C:/Oracle/ ♥ Departments: ♥ file:///C:/Oracle/ ♥ able Path: window(index=0).Tag[TABLE](index=3) ♥ ● est Element: B2 ● ● est Element: B2 ● ● String ♥ ● ● Departments: IKitchen & BathIIGardening ♥ Matches the following: © <u>Matches the following:</u> ● © Exact © WildCard © Begular Expression ● Departments: Ikitchen & BathIIGardening ♥ Test Table Test Iest		D						
Image: String Image: String String Image: String Departments: IIK itchen & BathIIGardening Matches the following: Image: String Image: String Image: String Image: String Departments: IIK itchen & BathIIGardening Image: String Image: String Image: String String Image: String		⊻	ile:///C:/Uracle4	nts: 🗹 fil	e/ 🗹 Departme	:///C:/Uracle	I file:/	2
est Element: B2 est Type: String String Departments: IIK itchen & Bath IIG ardening Matches the following: © Exact © WildCard © Regular Expression Departments: IIK itchen & Bath IIG ardening ♥ Test Table Test Result:Iest	Enable	_	31	LE lindex=3	idex=0).Tao(TA	window(in	Path:	∢ able
est Type: String String Departments: IIK itchen & BathIIGardening Matches the following: © Exact © WildCard © Begular Expression Departments: IIK itchen & BathIIGardening Test Table Test Result:Iest		E	-,		ioon oj.rogi.ri	11	Iement	ant E
String Departments: Kitchen & Bath Gardening Matches the following: Exact C WildCard C Regular Expression Departments: Kitchen & Bath Gardening Test Table Test Result:								
Departments: Kitchen & Bath Gardening Matches the following: Exact WildCard Begular Expression Departments: Kitchen & Bath Gardening Test Table Test Result: Iest		<u>-</u>				String		
Matches the following: Exact WildCard Regular Expression Departments: IIK itchen & BathIIG ardening Test Table Test Result:							ng	Strin
© Exact © WildCard © Begular Expression Departments: Kitchen & Bath Gardening ▼ Test Table Test Result:				ng	& Bath ll Garder	nts: Kitchen	epartmen	De
Departments: Kitchen & Bath Gardening						ne following:	tches the	Ma
Test Table Test			jular Expression	C <u>R</u> egu	○ <u>W</u> ildCard		Exact	ſ
Result:			•	ng	& Bath ll Garder	nts: Kitchen	epartmen	De
Result:						<u>.</u>		<u>.</u>
			Test [l est		830
Details:			Test					
D ordine.							tails:	Del
Cancel < Back Next> Fir			1					

Figure 3–31 Create Table Test Dialog Box with Field Selected

- **32.** Make sure that Exact is selected in the Matches the following field.
- **33.** Click **Test** to display the Last Played Table Tab. The result is displayed in the Result field.
- **34.** Click **Finish** and view the test case in the Visual Script. Oracle Functional Testing for Web Applications adds the test case to the Visual Script under the User Defined Tests node.





35. Save the script as tutor2.

3.7 Example 6: Using the Data Bank Wizard on a Search Form

This example introduces the Data Bank Wizard and explains one way to use the Data Bank Wizard with the Text Matching test case to verify Search results pages. The Data Bank Wizard provides the capability to run iterative tests using data from a Data Bank file.



- 1. Select **New Script** from the **File** menu to create a new Visual Script (save the previous script if prompted).
- 2. Reload the c:\OracleATS\OFT\rswdemo\index.htm page in the Browser pane by selecting it from the Browser drop down list.

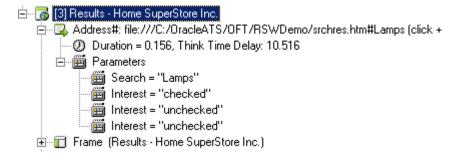
3.7.1 Recording a Search

- **3.** Click the Record button on the toolbar.
- **4.** Scroll the Browser pane and click the Go graphic next to Use SuperSearch.
- **5.** Enter Lamps in the **Product Name** field and click the **Search** button. Oracle Functional Testing for Web Applications records the search including the text you typed into the field.
- 6. Click the Stop Record button on the toolbar.

3.7.2 Viewing the Parameters in the Visual Script

7. Expand the [3] Results - Home Superstore, Inc page in the Visual Script. Notice the Search Parameter under the Address node of the tree.

Figure 3–33 Parameters Script Node



3.7.3 Using the Data Bank Wizard to Map Variables



8. Select Data Bank Wizard from the Edit menu.

Oracle Functional Testing for Web Applications opens the Data Bank Wizard window with the parameters from the Visual Script in the **Parameters** list.

arameters:		Variables:			OK Cance
Name	Recorded Valu	Name	Туре	Add Variable	
Search[3.1] nterest[3.2] nterest[3.3] nterest[3.4]	Lamps checked unchecked unchecked			Delete Variable	Help
				Add Mapping	1.1
	F	4		Remove Mapping	
Parameter Name	Variable Name	Variable Value	Recorded Value	Tested Value	
	Variable Name	Variable Value	Recorded Value		
Internet of Internet			1	Tested Value	

Figure 3–34 Data Bank Wizard Variables Tab

9. Click the **Add Variable** button. The Data Bank Wizard opens a dialog box for specifying a variable name.

Figure 3–35 Add Variable Dialog Box

🔆 Add Variable	×
Name:	OK
Type: External - Map to External Data Source	Cancel
	Reset
Expression:	Help
Value: Evaluate	
Use the EVALUATE button to compute the curren value of the expression you have specified.	t

10. Type SuperSearch as the variable name, and then click **OK**.

The Data Bank Wizard adds the name to the Variables list.

		Variables:			Cance
Name Search[3.1] Interest[3.2] Interest[3.3] Interest[3.4]	Recorded Valu Lamps checked unchecked unchecked	Name SuperSearch	Type External	Add Variable Delete Variable Add Mapping	Help
Iappings: Parameter Name	▶ Variable Name	Variable Value	Recorded Value	Remove Mapping	
				College College	
 Internet Record From 	Data Bank: 1	Feb	ch Record G	→ Dio Record	

Figure 3–36 Data Bank Wizard with a Variable Added

11. Select the Search[3.1] item in the **Parameters** list, and then click **Add Mapping**. The Data Bank Wizard creates a mapping between the Search[3.1] parameter and the SuperSearch variable. You now need to bind the variable name to a field in a Data Bank file.

3.7.4 Using the Data Bank Wizard to Bind to a Data Bank

12. Click the **Data Binding** tab. The Data Bank Wizard opens the Data Binding options with the variable name in the **Variables** list.

ata Bank						ОК
o Data Bank Sel	ected Sele	ct Clear	Edit	Update	Synthesize	Cance
'ariable Bindings- ariables:		Data Bank Fields	:			Help
Name	Туре			Add	Binding	
SuperSearch	External			Delete	e Binding	
and and a state of the state of						
(· ·	(1.1.1.)				
ndinae:		1				
ndings: Variable Name	Field Name	Field Value	Field Number			
Variable Name	Field Name		Field Number	Goto Reco	rd	

Figure 3–37 Data Bank Wizard Data Binding Tab

Now we want to select the Data Bank file that contains the values we want to use for iterative testing.

13. Click the **Select** button. The Data Bank Wizard opens a dialog box for selecting the Data Bank file.



Figure 3–38 Select Data Bank Dialog Box

Note: The .csv file name extension may or may not be visible depending upon your system settings.

14. Select search.csv, and then click **Open**. The Data Bank Wizard adds the Field name from the Data Bank file.

ϟ Data Bank Wizard	×
⊻ariables Data Binding	
Data Bank	ОК
search.csv Select Clear Edit Update Synthesize	Cancel
Variable Bindings Variables: Data Bank Fields:	Help
Name Type SuperSearch External Image: Character of the search Add Binding Delete Binding	
Bindings:	
Variable Name Field Value Field Number Current Record From Databank: 0 Fetch Record	
Unique Vars per Page Auto-Map Auto-Bind	
Unique Vars per Page is for Auto_Map. When it is checked, the variables name will be different even in some cases that the parameters name are identical for different pages.	

Note: The Data Bank file is a comma-delimited ASCII file with the field names as column headers on the first line of the file. Subsequent lines of the file contain data. You can view the contents of any of the sample files in the c:\OracleATS\OFT\DataBank directory using Notepad or any other ASCII editor.

15. Select the Search field in the **Data Bank Fields** list and then click the **Add Binding** button. The Data Bank Wizard adds the variable name to the Bindings list.

) ata Bank							OK
earch.csv	Selec	t Clear	Edit	Update	Synthesize.		Cancel
/ariable Bindings— 'ariables:		Data Bank Fields:					Help
Name SuperSearch	Type External	Search			dd Binding lete Binding		
Variable Name	Field Name	Field Value	Field Numbe	ar [1	
Current Record From	n Databank: 0	Fe	etch Record	Goto Re	ecord		
Current Record From			etch Record	Goto Re	cord		

Figure 3–40 Data Bank Wizard with Bindings

16. Click **Fetch Record** to cycle through the records in the external data file.

Figure 3–41 Data Bank Wizard Fields and Records View.

Bindings:			
Variable Name	Field Name	Field Value	Field Number
var_Search	Search	Plants	1
•			>
Current Record From Dat	abank: 1	Fetch Record	Goto Record

- **17.** Continue clicking the **Fetch Record** button to cycle through all of the records in the file. There are five records in the sample Data Bank file.
- 18. Click the OK button to close the Data Bank Wizard.

3.7.5 View the Data Bank Parameters in the Visual Script

19. Examine the Parameters node under the Address node of the Visual Script tree.

È <mark>[3] Results - Home SuperStore Inc.</mark> È [3] Address#: file:///C:/OracleATS/0FT/RSW/Demo/srchres.htm#Lamps (click +
Duration = 0.156, Think Time Delay: 10.516
🖻 🖷 Parameters
√ Search <supersearch> : Recorded = "Lamps", Tested = "Lamps"</supersearch>
🔤 Interest = "checked"
🔤 Interest = "unchecked"
🛅 Interest = "unchecked"

Figure 3–42 Data Bank Parameters Script Nodes

The Visual Script now includes the variable names as part of the Parameters. The check mark indicates that the parameter is mapped to a variable and bound to a field in a Data Bank file.

3.7.6 Insert a Text Matching Test Case

Now we want to insert a Text Matching text case that verifies that the search results were successful.

- **20.** Highlight the text "successfully found" in the search results page shown in the Browser pane.
- **21.** Click the Insert Text Matching test case button on the tool bar. Notice the text you highlighted is automatically captured by Oracle Functional Testing for Web Applications.
- **22.** Type VerifySearch as the case name, make sure **Pass when:** is set to **Selected text is present**, and then click **OK**.
- **23.** Select the HTML node in the Visual Script. Notice the yellow flag next to the HTML node. Oracle Functional Testing for Web Applications automatically turns off the HTML comparison test when you insert a Text Matching test.

Since the search criteria will be different for each search during iterative play back, we know the HTML and product images on the page will be different from the recorded master each time. Instead of testing the HTML and images, the Text Matching test case will be used to verify a successful search.

24. Save the Visual Script as tutor3.

3.7.7 Play Back the Script with Iterations

25. Select **Playback** from the **Run** menu and then select **Iterate**. Oracle Functional Testing for Web Applications opens the Iterations dialog box.



Apc

ayback without Data Bank	
Number of Iterations: 1	<u>H</u> elp
Delay Between Iterations: 0 seconds	
ayback with Data Bank	
C Use All Records C Use Current Record Only	
Filename: search.csv	
Current Record Number: 1	
Start at Record: First	
End at Record:	
Iterate Over Range:	
Norace e for Hange.	

Figure 3–43 Iterations Dialog Box

- 26. Select Playback with Data Bank and Use All Records.
- **27.** Click the **OK** button to playback the Visual Script.
- **28.** Watch as Oracle Functional Testing for Web Applications plays back the script several times using a different data value for the search each time.

3.7.8 Analyzing a Playback Failure

29. Close the Results Report.

Notice the Results log indicates that the search using the Furniture value (record 3) fails. We now need to analyze this failure.

- **30.** Select **Data Bank Wizard** from the **Edit** menu.
- **31.** Click the **Goto Record** button. The Data Bank Wizard opens a dialog box for entering the record number.
- **32.** Enter 3, and then click **OK**.
- 33. Click OK to close the Data Bank Wizard.
- **34.** Click the Playback button on the toolbar to play back this one record.
- **35.** Scroll the Browser pane to view the Search Results page. Notice that the page indicates there is no product information for Furniture. Oracle Functional Testing for Web Applications was able to find this problem because the Text Matching test case you added to test for a successful search produced a failure.
- **36.** Make sure the [3] Results Home Superstore, Inc page is still selected in the Visual Script.
- **37.** Click the right mouse button and select **Discard Tested Page** from the shortcut menu. The red circles are removed from the Visual Script. The Master version of the Visual Script is still the baseline to use for testing of the Web page.



×.

3.7.9 Save the Script and the Results Log

- **38.** Save the Visual Script.
- **39.** Save the Output Log as tutor3.log.
- **40.** Select **Clear Results Window** from the **View** menu to clear the results log for the next test.

3.8 Example 7: Using the Data Bank Wizard on a Registration Form

This example explains how the Data Bank Wizard makes it easy to create automated data-driven tests. Data Banks are used to hold unlimited amounts of input data that can be fed automatically into your Web application.

- 1. Select New Script from the File menu to create a new Visual Script.
- **2.** Reload the c:\OracleATS\OFT\rswdemo\index.htm page in the Browser pane by selecting it from the Browser drop-down list.

3.8.1 Recording Information in a Form

- **3.** Click the Record button on the toolbar.
- **4.** Scroll the Browser pane and click the <u>Register</u> link. The Registration page contains a form for entering Name, Email Address, and Phone number information.
- **5.** Enter your own information into the form and click the **submit my entry** button. The Results page returns showing the information you entered with a "successful registration" message.

3.8.2 Inserting a Text Matching Test Case

Now we want to insert a Text Matching text case that verifies that the Registration results were successful.

- 6. Highlight the text "Your registration has been added" in the search results page.
- **7.** Click the Insert Text Matching test case button on the tool bar. Notice the text you highlighted is automatically captured by Oracle Functional Testing for Web Applications.
- **8.** Type VerifyRegistration as the test case name, make sure **Pass when:** is set to **Selected text is present**, and then click **OK**.
- 9. Click the Stop Record button on the toolbar.

Apc

10. Select **Save Script As** from the **File** menu and save the file as tutor4.

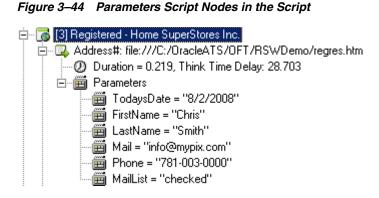
3.8.3 Viewing the Parameters in the Visual Script

11. Expand the [3] Registered - Home Superstore, Inc page in the Visual Script. Notice the Parameters under the Address node of the tree.



*****@|





12. Make sure the [3] Registered - Home Superstore, Inc page is selected in the Visual Script.

3.8.4 Using the Data Bank Wizard to Map Variables

ð

13. Select Data Bank Wizard from the Edit menu.

Oracle Functional Testing for Web Applications opens the Data Bank Wizard window with all the parameters from the Visual Script in the **Parameters** list.

💑 Data Bank Wizard			×
⊻ariables Data Binding			
Variable Mappings:			ОК
Parameters:	Variables:		Cancel
Name Recorded Valu	Name Type	Add Variable	
TodaysDate[3.1] 5/28/2008 FirstName[3.2] Chris			Help
LastName[3.3] Smith		Delete Variable	
Mail[3.4] info@mypix.cot Phone[3.5] 781-003-0000		Add Mapping	
MailList[3.6] checked			
	۱	Remove Mapping	
Mappings:			
Parameter Name Variable Name	Variable Value Recorded Value	Tested Value	
Current Record From Databank: 0	Fetch Record (Goto Record	
🔲 Unique Vars per Page	Auto-Map	Auto-Bind	
Unique Vars per Page is for Auto	Map. When it is checked, the variable meters name are identical for different p	s name will be different	
	and are real to the first of directing		
L			

Figure 3–45 Data Bank Wizard Variables Tab

14. Click the **Auto-Map** button. The Data Bank Wizard automatically creates variable names and maps the variable names to the Parameter names.

<u>V</u> ariables <u>D</u> ata Bind -Variable Mappings: -					ОК
Parameters: Name TodaysDate[3.1] FirstName[3.2] LastName[3.3] Mail[3.4] Phone[3.5] MailList[3.6]	Recorded Valu 9/17/2001 Chris Smith info@mypix.co 781-003-0000 checked	Variables: Var TodavsDate var_TirstName var_LastName var_Mail var_Phone var_MailList	Type External External External External External External	Add Variable Delete Variable Add Mapping	Cancel Help
Mappings:	<u> </u>				
Parameter Name TodaysDate[3.1] FirstName[3.2] LastName[3.3] Mail[3.4] Phone[3.5] MailList[3.6]	Variable Name var_TodaysDate var_FirstName var_LastName var_Mail var_Phone var_MailList	Variable Value 9/17/2001 Chris Smith info@mypix.com 781-003-0000 checked	Recorded Value 9/17/2001 Chris Smith info@mypix.com 781-003-0000 checked	Tested Value 9/17/2001 John Smith JohnS@IBM.com (603)993-0000 checked	
Current Record From	Databank: 0	Fetcl	h Record G	oto Record	
Unique Vars per P	age	Auto-M	ſар	Auto-Bind	
	per Page is for Auto	Map. When it is che	cked, the variables	name will be different	

Figure 3–46 Data Bank Wizard with Auto-mapped Parameters

You now need to bind the variable names to fields in a Data Bank file.

3.8.5 Using the Data Bank Wizard to Bind to Data Source

15. Click the **Auto Bind** button. The Data Bank Wizard opens the Data Binding options with the variable names in the **Variables** list.

Note: Auto Bind automatically creates a Data Bank file with field definitions and one record of data. It also automatically binds the fields to the variables created.

Data Bank							OK
utor4.csv	Selec	t Clear	Edit	Update	Synthesize] [=	Cancel
/ariable Bindings— /ariables:		Data Bank Fields:				7 [Help
Name var_LastName var_Mail var_Phone var_MailList	Type External External External External	TodaysDate FirstName LastName Mail Phone MailList			odd Binding elete Binding		
Bindings:		1					
Variable Name var_TodaysDate var_FirstName var_LastName var_Mail var_Phone var_MailList	Field Name TodaysDate FirstName LastName Mail Phone MailList	Field Value 5/28/2008 Chris Smith info@mypix.c 781-003-0000 checked	Field Number 1 2 3 4 5 6				
Current Record From	n Databank: 1	Fet	ch Record	Goto R	ecord		
Unique Vars per F	age	Auto	Мар	(Auto-Bind		
and Unious Visco	per Page is for Aul	o_Map. When it is cl	pecked the var	iables name	a will be differen		

Figure 3–47 Data Bank Wizard Data Binding Tab

Now we want to select the data source that contains the values we want to use for iterative testing.

16. Click the **Select** button. The Data Bank Wizard opens a dialog box for selecting the Data Bank file.

? ×

Open

Cancel

Select DataBank
Look in: DataBank
I Cook in: D

CSV(*.csv),TXT(*.txt)

🔲 Open as read-only

Figure 3–48 Select Data Bank Dialog Box

17. Select customer.csv, and then click **Open**.

File name:

Files of type:

The Data Bank Wizard automatically re-binds the appropriate data field names from the Data Bank file to the variable names.

•

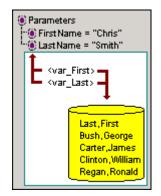
•

Data Bank	Select Clear	Edit Ur	odate Synthesize	OK
customer.csv				Cancel
Variable Bindings /ariables:	Data Bank Fields:			Help
Name Type var_LastName External var_Mail External var_Phone External var_MailList External	Phone Mail		Add Binding Delete Binding	
Bindings:				
Variable Name Field Na var_TirstName FirstNan var_LastName LastNar var_Mail Mail var_Phone Phone	ie John	Field Number 1 2 3 4		
Current Record From Databan	c 1 Fe	ch Record	Goto Record	
Unique Vars per Page	Auto	-Map	Auto-Bind	_

Figure 3–49 Data Bank Wizard with Bound Parameters

Note: The Data Bank file is a comma-delimited ASCII file with the field names as column headers on the first line of the file. Subsequent lines of the file contain data. You can view the contents of any of the sample files in the c:\OracleATS\OFT\DataBank directory using Notepad or any other ASCII editor. The following illustration shows how Data Banks map to variables and Visual Scripts.

Figure 3–50 How Scripts Map and Bind to Data Bank Records



18. Click the **Fetch Record** button to cycle through the records in the Data Bank file.

Variable Name	Field Name	Field Value	Field Number
var_FirstName	FirstName	John	1
var_LastName	LastName	Smith	2
var_Mail	Mail	JohnS@IBM.com	3
var_Phone	Phone	(603) 993-0000	4
•			•
Current Record From	Databank: 1	Eetch Becord	

Figure 3–51 Data Bank Wizard Fields and Records View

- **19.** Continue clicking the **Fetch Record** button to cycle through all of the records in the Data Bank file.
- 20. Click the OK button to close the Data Bank Wizard.

3.8.6 View the Data Bank Parameters in the Visual Script

21. Examine the Parameters node under the Address node of the Visual Script tree.

Figure 3–52 Script Nodes with Mapped and Bound Data Bank Variables

🚊 🕝 🐻 [3] Registered - Home SuperStores Inc.
🚊 🖳 Address#: file:///C:/OracleATS/OFT/RSWDemo/regres.htm#8/2/2008#Chris#Smith#email@
🥑 Duration = 0.219, Think Time Delay: 28.703
🖻 🧰 Parameters
— 🕨 TodaysDate <var_todaysdate> = ''8/2/2008''</var_todaysdate>
√ LastName <var_lastname> : Recorded = "Smith", Tested = "Smith"</var_lastname>
√ Mail < var_Mail> : Recorded = "info@mypix.com", Tested = "info@mypix.com"
🕨 🕨 MailList <var_maillist> = "checked"</var_maillist>

The Visual Script now includes the variable names as part of the Parameters. The check marks indicate the parameters that are mapped to variables and bound to fields in a Data Bank file. The triangles indicate variables that are mapped, but not bound to fields in a Data Bank file.

22. Save the Visual Script.

H

£

3.8.7 Play Back the Script with Data Iterations

- **23.** Select **Current Script** from the **Options** menu and then select **Results Log**, make sure the **Failures Only** radio button in the **Results** section is selected, and then click **OK**.
- **24.** Select **Playback** from the **Run** menu and then select **Iterate**. Oracle Functional Testing for Web Applications opens the Iterations dialog box.

	Bank	ОК
Playback with Data Bar	nk	<u>C</u> anc
ayback without Data Ban	k	<u>R</u> ese
Number of Iterations:	1	Help
Delay Between Iteration:	s: 0 seconds	
ayback with Data Bank—		1
O Use All Records	C Use Current Record Only	
Filename:	search.csv	
Current Record Number:	1	
Start at Record:	First	
End at Record:	Last	
	1	
Iterate Over Range:		
Iterate Over Range:		

Figure 3–53 Iterations Dialog Box

- 25. Select Playback with Data Bank and Use All Records.
- **26.** Click the **OK** button to play back the Visual Script.
- **27.** Watch as Oracle Functional Testing for Web Applications plays back the script several times using different data values for the registration.

3.8.8 Analyzing a Playback Failure

Notice that the Results log indicates a failure for Record 4 of the playback iteration. We now need to analyze this failure.

- 28. Select Data Bank Wizard from the Edit menu.
- **29.** Click the **Goto Record** button. The Data Bank Wizard opens a dialog box for entering the record number.
- **30.** Enter 4, and then click **OK**.
- **31.** Click the **OK** button to close the Data Bank Wizard.
- **32.** Click the Playback button on the toolbar to play back this one record.
- **33.** Close Results Report.

You can play back the current record repeatedly using **Current Script** from the **Options** menu, select the **Playback** section, and then select the **Use Current Record** option.

- **34.** Scroll the Browser pane to view the Registration Results page. Notice that the page indicates a server error. Oracle Functional Testing for Web Applications was able to find this error because the Text Matching test case you added to test for successful registration produced a failure.
- **35.** Make sure the [3] Results Home Superstore, Inc page is still selected in the Visual Script.



36. Click the right mouse button and select Discard Tested Page from the shortcut menu. The red circles are removed from the Visual Script. The Master version of the Visual Script is still the baseline to use for testing of the Web page.

3.8.9 Save the Script and the Results Log

H

- **37.** Save the Visual Script.
- **38.** Save the Output Log as tutor4.log.
- 39. Select Clear Results Window from the View menu to clear the results log.

This completes the Oracle Functional Testing for Web Applications tutorial.

Job Scheduler Tutorial

This tutorial walks you through the main features of Job Scheduler. You can follow the examples in this chapter to become familiar with the features and use of Job Scheduler.

This tutorial consists of the following examples:

- **Creating a job and schedule** describes how to create a job using the Job Scheduler Wizard and then add it to a schedule.
- Editing a Job explains how to edit a job after you have created it.
- Editing a Schedule explains how to edit a schedule after you have created it.

The tutorial is designed to be followed sequentially from beginning to end and assumes you have completed the Oracle Functional Testing for Web Applications tutorial in Chapter 3. The examples in this tutorial refer to Visual Scripts recorded in the Oracle Functional Testing for Web Applications tutorial.

4.1 Example 1: Creating a Job and Schedule

This example illustrates how to create a job using the Job Scheduler Wizard and then add it to the Current Schedule.

4.1.1 Starting Job Scheduler

- 1. Select **Programs** from the **Start** and then select **Job Scheduler** from the **Oracle Application Testing Suite** submenu to start Job Scheduler.
- 2. Select New Job from the File menu or click the Job Scheduler Wizard toolbar button. Job Scheduler opens the Wizard Welcome screen. If you do not want the Welcome screen to appear each time you run Job Scheduler, select the kip this screen in the future check box.
- 3. Click the Next button to continue to the Workspace and Scripts [Step 1 of 2] screen.

4.1.2 Specifying the Scripts

The Workspace and Scripts [Step 1 of 2] screen of the Job Scheduler Wizard is where you specify the scripts to include in the job.

Script Selection	DN	Data Banks
Workspace: RS	WDemo 💌	Add 📀 Without
Script		With
tutor1		
tutor2		C All Record
tutor3		Single Re
tutor4		
•		
Script Order		
Script	Workspace	Databank Iteration
tutor1	RSWDemo	Without Databank

Figure 4–1 Job Scheduler Wizard Workspace and Scripts Dialog Box

- **4.** Select the RSWDemo workspace. Note that you can include scripts from more than one workspace in a job.
- 5. Select tutor3 and click on the Add button to add the script to the Script Order list.
- 6. Select tutor4 and click on the Add button.
- **7.** Click the **Next** button to continue to the Job Notifications Log Window [Step 2a of 2] screen of the Job Scheduler Wizard.

4.1.3 Specifying the Job Notifications

The Job Notifications Log Window [Step 2a of 2] screen of the Job Scheduler Wizard is where you specify the message that will appear in the Results Pane and the Results Log if an error occurs during playback of a Visual Script.

Job Scheduler Wizard	- Job Notifications - Log Window [Step 2a of 2]	
	Specify the format of the message to be displayed in the log window.	
	Log Message Format	
	Error was encountered at: %Time Current Data Bank:%curRecord Error Description: %errDescription	
Help	Cancel (Back Next)	_

Figure 4–2 Job Scheduler Wizard Log Window Dialog Box

The default log message writes the following information to the results log if an error occurs during playback of a Visual Script:

Time of the error

- Databank record
- An error number
- A description of the error
- **8.** Click the **Next** button to continue to the Job Notifications Email [Step 2b of 2] screen of the Job Scheduler Wizard.

4.1.4 Specifying Email Notifications

The Job Notifications Email [Step 2b of 2] screen of the Job Scheduler Wizard is where you specify who to notify when the job is finished and what information to send.

Figure 4–3 Job Scheduler Wizard Email Dialog Box

	Specify the	e recipient address, subject and message to send.	
	On Job F	tification Configuration inished nd HTML Summary nd Text Log	
· Sol 雪	Address: Subject:	Enter email addresses of recipients here Job Scheduler Job Report: %job	Test
			Mail Server

- **9.** Select the **Send HTML Summary** check box to send the Job Report to the recipients.
- **10.** Select the **Send Text Log** to send a text version of the log.
- **11.** Enter your email address in the **Address** field. Separate additional email addresses by a comma or semicolon.
- 12. Click Mail Server. Job Scheduler opens the server configuration dialog box.

Figure 4–4 Job Scheduler Wizard Mail Server Configuration Dialog Box

Job Scheduler Mail Server (-mail Server Type	_oningeraci		
SMTP Server			
C MAPI server			
MTP Information			
E-Mail Sender:			
Outgoing Mail Server (SMTP):]
est e-mail recipient:		Test	E-mail

- **13.** Select your email server type and enter the email information for your email system. Check with your system administrator if you are not sure of the information to enter for your email configuration. You can use the **Test email recipient** address and **Test Email** button to verify email capabilities from within Job Scheduler.
- 14. Click OK to return to the Job Scheduler Wizard.
- **15.** Click the **Test** button next to the **Address** field if you want to verify the email notification from the wizard.
- 16. Click the Finish button. The Save Job As dialog box is displayed.
- **17.** Enter tutorJob1 for the job name and click **Save**.
- 18. You are then asked if you want to schedule the job. Click Yes.

4.1.5 Scheduling the Job

The Current Schedule window is displayed as well as the Edit dialog box for scheduling the job you just created.

Figure 4–5 Job Scheduler Wizard Edit Schedule Dialog Box

dit Schedule - C:\Ora	cleATS\OFT	\RSWDemo!\Tut.
Run At 💽:00 PM	-	
From Saturday	▼ Through	Saturday 💌
ОК		Cancel

- **19.** Change the **Run At** time to 12:00 pm.
- **20.** Change the starting day in the From field to **Sunday** and the ending day in the Through field to **Thursday**
- **21.** Click **OK** to schedule the job.

	luler: Current Sched	and the second	idle]	in egy					<u>_ ×</u>
WorkSpace	es t emo torJob1	12 am 1 am 2 am 3 am 4 am 5 am 6 am 7 am 8 am 9 am 10 am	Sun.	Mon.	Tue.	Wed.	Thu.	Fri.	Sat.
Name	Iterations Fa	ailures Warnings	Status		Last E	Error	Fi	rst Error	
	Show Details	Clear Results	Show L	1	Show Re		Show HTI		

Figure 4–6 Job Scheduler Current Schedule Window

4.1.6 Saving the Schedule

The job file is separate from the schedule file. The job is the list of scripts to run and the notification information. The schedule file contains the information about when the scripts will be played back and which jobs will be played back.

- 1. Select Save Schedule As from the File menu.
- 2. Enter tutorSch1 for the schedule name and click Save.

The file name is displayed in the title bar of the Current Schedule window.

4.1.7 Playing Back the Job

- 1. In the Job Editor window, click the **Start** button to play back the job.
- **2.** Watch as Job Scheduler processes the Visual Scripts. A yellow bar in the row of the script indicates that the script is being processed. A red bar indicates that the script failed; a green bar indicates that the script passed.

Upon completion an email will be sent to you with the job report and text log if you used your email address when setting up the job notifications.

4.1.8 Activating the Schedule

1

When the schedule is activated, the scheduled jobs automatically run at the scheduled times.

- 1. Make the Current Schedule window active by clicking anywhere in it.
- 2. Select Activate Schedule from the Control menu or click on the Activate Schedule toolbar button. The results are displayed in the Job Summary pane of the Current Schedule window.

Only one schedule can be opened or activated at any one time.

Double-click on the job name in the Job Summary pane to view Job Details as the scripts are played back.

3. Select **Deactivate Schedule** from the **Control** menu or click on the Deactivate Schedule toolbar button to deactivate the schedule.

4.2 Example 2: Editing a Job

This example illustrates how to edit a job using the Job Editor window after the job has been created with the Job Scheduler Wizard. Note that you can have more than one job open at any one time.

1. Select **Open Job** from the **File** menu to open a job or make the Job Editor window active by clicking on it to edit the currently open job.

Figure 4–7 Job Editor Window

🛠 Job Scheduler: Job Editor - RS	#Demo - [TutorJol	o1]		
▼ 🔜 ► 🗉 🔮 🔝 🤅	1			
	Script Name tutor3 tutor4	Result Passed Passed	Duration (Seconds) .4 .6	Summary

2. Click Scripts in the job tree to display the script selection options.

Figure 4–8 Job Editor Script Selection Window

▼ Iutorjob11 → Notification → Log (True) ↓ Evail (False) ↓ Servity Mapping	Script Selection Workspace: FSV Script tutor1 tutor2 tutor3 tutor4	A REAL PROPERTY AND A REAL	Add C without C All F	ut
	Script Order	Workspace RSWDemo RSWDemo	Databank Iteration Without Databank Without Databank	×

- 3. Select the RSWDemo workspace and add the tutor2 script to the Script Order list.
- 4. Click Email in the job tree to display the email options.

Figure 4–9 Job Editor Email Window

🛠 Job Scheduler: Job Editor - RSV	VDemo - [Tutorjob11*]	_ 🗆 ×
	WDemo - [Tutorjob11*] Specify the recipient address, subject and message to send. E-Mail Notification Configuration On Job Finished Send HTML Summary Send Text Log Address: Enter email addresses of recipients here Test Subject: Job Scheduler Job Report: %job	
	C Enable C Disable Mail Serve	ər

- 5. Select the **Disable** option button to disable sending emails on error.
- 6. Save the job by selecting **Save tutorJob1** from the **File** menu.

4.3 Example 3: Editing a Schedule

This example illustrates how to edit a schedule by adding and changing playback times of jobs. This example uses only one job but you can schedule more than one job.

1. Select **Open Schedule** from the **File** menu to open tutorSch1.mjs or make the Current Schedule window active by clicking on it to edit the currently open job.

WorkSpaces		a second s	Sun.	Mon.	Tue.	Wed.	Thu.	Fri.	Sat
🗖 Default		12 am	Cont.	The second		1100.		** * * *	Con
📥 RSWDer	mo	1 am	1.12.20	196 (196 (197	a second a s	and the second second		12.5	
Tutor		2 am 3 am	and a second						
Cample		4 am		and the second			and the second		
		<u>5 am</u>							1.1
		<u>6 am</u> 7 am	and the second s						
		8 am	1999 - 1999 -	Sec. Sec.					222
		<u> </u>	and the second second						
		11 am	*******	1999			and a second second	1111	
		12 pm	12:00 PM	h					1999 - 1999 1997 - 1997 1997 - 1997
		1 pm 2 pm		and the second			an an Anna an Anna. An Anna Anna Anna Anna	an ta ta la	
		ma C	Sec. 25	1999 (1997) 1997 - 1997 (1997)	the second second			1.1.1	
		4 pm 5 pm		All and a second					
		6 pm		and the second second	j i i			and the second s	
		7 pm	and the second	and the second second		a de la companya de			
		ma 8 ma 9		<u>- 14 - 41 - 14 - 14 - 1</u> 74			· · · · · · · · · · · · · · · · · · ·		
		10 pm	· ·	9	· ·				
		11 pm							
	and the second sec	a server a la			*****	*****			
me	Iterations Fa	ailures Warnings	Status		Last Err	or	First Err	or	

Figure 4–10 Job Scheduler Current Schedule Window

2. Double-click on the job in the calendar or right-click on the job and select **Edit Schedule**.

Figure 4–11 Edit Schedule Dialog Box

Edit Schedule - C:\Or	acleATS\OFT\R	SWDemo!\Tut
Run At 1 200 PM		
From Sunday	▼ Through	Thursday 💌
ОК	_	Cancel

- **3.** Change the time to 11:00 am, the starting day to **Monday** and the ending day to **Friday**.
- 4. Click on tutorJob1 in the job tree and drag it to Sunday at 6:00 pm.

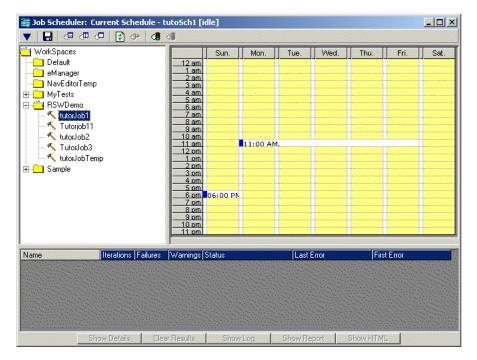


Figure 4–12 Job Scheduler Current Schedule Window

5. Save the schedule by selecting **Save tutorSch1** from the **File** menu.

Oracle Load Testing for Web Applications Tutorial

This tutorial walks you through the main features of Oracle Load Testing for Web Applications. Oracle Load Testing for Web Applications is a separate product in the Oracle Application Testing Suite, which you may or may not have purchased. If you have the Oracle Load Testing for Web Applications version of the Oracle Application Testing Suite, you can follow the examples in this chapter to become familiar with the features and use of Oracle Load Testing for Web Applications.

The tutorial consists of the following examples:

- Performing a Simple Load Test shows how to use Oracle Load Testing for Web Applications to run virtual users to simulate load on a Web application.
- Adding Data Sources shows how to add data sources to the Oracle Load Testing for Web Applications ServerStats configuration to monitor server-side statistics, such as CPU usage, and available memory.
- Editing Data Sources shows how to edit existing Oracle Load Testing for Web Applications ServerStats configurations to modify specific counters.
- Creating a Scenario with Multiple Profiles shows how to add a new Visual Script to the Oracle Load Testing for Web Applications Scenario. This example also shows how to set the Reporting options and Session Start/Stop options to save data for use in post-run analysis.
- Running Multiple Profiles shows how to use Oracle Load Testing for Web Applications to run multiple Scenario profiles with different amounts of virtual users and how to view statistical and performance information.
- **Controlling Virtual Users** shows how to modify individual virtual user attributes, view actions, and stop and abort virtual users.
- Generating Reports explains how to view the default reports and generate reports for post-run analysis.
- Creating User-Defined Profiles explains how to create user-defined virtual user profiles.

The tutorial is designed to be followed sequentially from beginning to end and assumes you have completed the Oracle Functional Testing for Web Applications tutorial in Chapter 3. The examples in this tutorial refer to Visual Scripts recorded in the Oracle Functional Testing for Web Applications tutorial.

5.1 Example 1: Performing a Simple Load Test

This example shows how to use Oracle Load Testing for Web Applications to run virtual users to simulate load on a Web application. The example illustrates how to run a previously recorded Visual Script to simulate multiple users accessing a Web application.

5.1.1 Starting Oracle Load Testing for Web Applications and Specifying the Workspace

1. Select **Programs** from the **Start** menu and then select **Oracle Load Testing for Web Applications** from the **Oracle Application Testing Suite** submenu to start Oracle Load Testing for Web Applications.

Oracle Load Testing for Web Applications - Build	Scenarios - Microsoft Internet Explorer	
File Edit View Favorites Tools Help		
🕒 Back 🔹 🕥 🖌 💌 🛃 🏠 🔎 Search	🛧 Favorites 🔣 😥 🛬 🔚 - 📴 🎇 🦓	
Address 🛃 http://localhost:8088/olt/LoginSubmit.do		💌 🛃 Go 🛛 Links 🎽
	eb Applications Scenario 🖬 Session 🖬 ServerStats 👻 Tools 👻 M	anage 👻 Help 👻 Logout 🔼
 Previous Session: <none></none> Current Scenario: <none></none> 	SCENARIO ACTIONS DE GI RUN TEST	VIRTUAL USER
Build Scenarios Set up Autopilot	Watch VU Grid View Run Graphs Create Reports	
Select scripts & user-defined profiles	Configure parameters of the scenario	
Repository: Default	Scripts/Profiles # VUs System User Mode Iteration Delay VU Pacing (Thir	ik Time)
Workspace: Default		
Add to scenario		Add to Autopilot
		¥
E Done		Secol intranet

Figure 5–1 Oracle Load Testing for Web Applications Main Window

- 2. Enter your user name and password to log in.
- 3. Select RSWDemo in the Workspace list.

5.1.2 Specifying a Scenario Profile

- **4.** Make sure the **Build Scenario** tab is displayed in Oracle Load Testing for Web Applications.
- **5.** Select tutor1 in the **Select scripts** list. These are the Visual Scripts that you record using Oracle Functional Testing for Web Applications.

6. Click the **Add to scenario** button to add tutor1 to the **Configure Parameters** list. You can also double-click the script name to add it to the **Configure Parameters** list.

Scripts/Profiles	# ∨Us	System	User Mode	Iteration Delay	VU Pacing (Think Time)	
🕼 tutor1	10	e-Load Server 💌	Thin Client 👤	1	Recorded	-
•						

Figure 5–2 Configure Parameters Pane

Oracle Load Testing for Web Applications automatically specifies a set of default virtual user attributes for the Scenario Profile in the Scenario tab. For this example, we'll use the default attributes.

7. Click the Add to Autopilot button on the Build Scenario tab.

Oracle Load Testing for Web Applications automatically opens the Set up Autopilot tab with the tutor1 Scenario Profile listed in **Submitted Scenario Profiles** list.

Figure 5–3 Autopilot Window

Timing and eve	nt controls						
 Start the load When the star After a delay At a specific ti Synchronize V 	t button is pre of (hh:mm:ss) ime (hh:mm:s:	00 : 00 : 00	O After each us O After a delay	p button is pressed			
ServerStats Co	nfiguration					🥖 Edit Configu	urations
Configuration: Description:			×	Monitors:			
Submitted Scen							
Profiles	VUs	Remaining	Running	with Error	Finished	System	
🕼 tutor1	10	10	0	0	0	e-load server	
					🛪 Clear A	utopilot 🛛 🔞 Pause Auto	pilot

5.1.3 Running the Scenario Profile Using Autopilot

- **8.** Select After each user plays [#] iteration option from the **Stop the load test** group of the Autopilot tab.
- **9.** Enter 5 in the After each user plays edit box.
- 10. Select [#] users below Add per step in the Virtual User ramp up group.
- **11.** Enter 1 in the Add per step [#] users edit box.
- 12. Select [#] seconds below After every in the Virtual User ramp up group.
- **13.** Enter 10 in the [#] seconds edit box.
- 14. Click the Run Test button on the Autopilot tab or the toolbar.
- 15. Select Yes when asked to record session data and click OK.

Oracle Load Testing for Web Applications starts running the virtual users in the Virtual User Grid. Watch as the Autopilot starts running the tutor1 Visual Script as ten virtual users.

1tutor1Finished57.09localhost2tutor1Finished56.649localhost3tutor1Finished56.179localhost4tutor1Finished56.149localhost5tutor1Finished56.109localhost6tutor1Finished56.249localhost7tutor1Finished56.079localhost8tutor1Finished56.569localhost9tutor1Finished56.149localhost	/U-ID	Profile	Status	Iterations	Failed	Last Run Time	Current Page	System	Data Bank	Current Error	Previous Error
3tutor1Finished56.179localhost4tutor1Finished56.149localhost5tutor1Finished56.109localhost6tutor1Finished56.249localhost7tutor1Finished56.079localhost8tutor1Finished56.569localhost	1	tutor1	Finished	5		7.09		localhost			
4tutor1Finished56.149localhost5tutor1Finished56.109localhost6tutor1Finished56.249localhost7tutor1Finished56.079localhost8tutor1Finished56.569localhost	2	tutor1	Finished	5		6.649		localhost			
4tutor1Finished56.149localhost5tutor1Finished56.109localhost6tutor1Finished56.249localhost7tutor1Finished56.079localhost8tutor1Finished56.569localhost		tutor1	Finished	5		6.179		localhost			
6tutor1Finished56.249localhost7tutor1Finished56.079localhost8tutor1Finished56.569localhost		tutor1	Finished	5		6.149		localhost			
7 tutor1 Finished 5 6.079 localhost 8 tutor1 Finished 5 6.569 localhost	5	tutor1	Finished	5		6.109		localhost			
8 tutor1 Finished 5 6.569 localhost	6	tutor1	Finished	5		6.249		localhost			
	7	tutor1	Finished	5		6.079		localhost			
9 tutor1 Finished 5 6.149 localhost	8	tutor1	Finished	5		6.569		localhost			
	9	tutor1	Finished	5		6.149		localhost			
10 tutor1 Finished 5 6.089 localhost	10	tutor1	Finished	5		6.089		localhost			

Figure 5–4 Virtual User Status Grid Window

16. Allow the virtual users to continue running until all of them indicate Finished in the **Status** column of the virtual user grid.

Congratulations. You have just performed a simple load test on the Demo Web application. Oracle Load Testing for Web Applications performs the virtual user Web interaction in the background. You can monitor the virtual users in the grid as they are running. In the later examples of this tutorial, you'll see how to use Oracle Load Testing for Web Applications to view statistical and performance information, and how to view virtual user actions.

5.2 Example 2: Adding Data Sources

This example shows how to add data sources to the Oracle Load Testing for Web Applications ServerStats configuration to monitor server-side statistics, such as CPU usage, and available memory.

Oracle Load Testing for Web Applications ServerStats can monitor statistics from a variety of systems and server types. This tutorial adds counters from your local Windows 200x/XP system to demonstrate the features of Oracle Load Testing for Web Applications ServerStats. If you are not running the Oracle Application Testing Suite on a Windows 200X/XP machine, you should skip examples two and three and continue the tutorial with example 4.

To configure counters from a Windows 200x/XP data source, do the following:

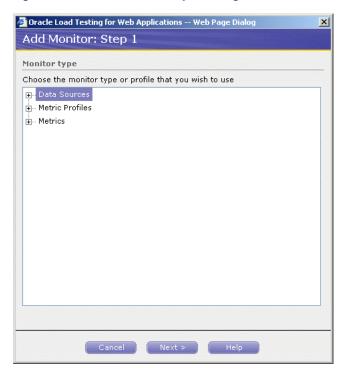
- 1. Select **Configurations** from the **ServerStats** menu. Oracle Load Testing for Web Applications opens the ServerStats Configurations window.
- 2. Click New to add a new configuration.

Figure 5–5 Add Configuration Dialog Box

Add Config	guration			
Name:				
Description:				
		Cancel	Help	

- **3.** Type Tutorial for the Name and the Description.
- **4.** Click **Save**. The Configuration window changes to include the Monitors configuration options.
- 5. Click New in the Monitor pane to open the Add Monitor Step 1 window.

Figure 5–6 Add Monitors Step 1 Dialog Box



6. Click the Plus icon next to Data Sources to expand the list of data sources.

Oracle Load Testing for Web Applications Web Page Dialog	2
Add Monitor: Step 1	
1onitor type	
Choose the monitor type or profile that you wish to use	
<mark>≓</mark> Data Sources	
COM+	
Database	
IBM WebSphere 4.0+	
XMC	
Perfmon (Windows Performance Monitor)	
Ping	
SNMP	
Virtual Agent	
Cancel Next > Help	

Figure 5–7 Add Monitors Step 1 with Data Sources Expanded

7. Select Perfmon (Windows Performance Monitor) and click Next.

Figure 5–8 Add Monitors Step 2 Dialog Box

of dere coud resen	ng for Web Applications Web Page Dialog	2
Add Monitor:	: Step 2	
Monitor		
Type: Perfmon (W	indows Performance Monitor)	
Monitored Syster	n	🐮 Ado
Name:	OLT Server	🥒 Edit
Host Name or IP:	bed-I-risa	
Perfmon (Windo	ws Performance Monitor)	
Username:		
Password:		
Domain Name:		
Domain Name:		
Domain Name: Data Collector		街 (M
Data Collector	OLT Server •	
Data Collector Name:	OLT Server •	
Data Collector Name: Host Name or IP:		
Data Collector Name: Host Name or IP:	bed-I-risa	
	bed-I-risa 1099	
Data Collector Name: Host Name or IP: Port: Collection Interv	bed-I-risa 1099 al	
Data Collector Name: Host Name or IP: Port:	bed-I-risa 1099 al	¥) Adc
Data Collector Name: Host Name or IP: Port: Collection Interv	bed-I-risa 1099 al	
Data Collector Name: Host Name or IP: Port: Collection Interv	bed-I-risa 1099 al	
Data Collector Name: Host Name or IP: Port: Collection Interv	bed-I-risa 1099 al	/ Edit

This step lets you specify which to monitor and which system to use for the data collector.

8. Leave the default settings for Monitored System and Data Collector.

9. Click Next to select the specific counters to monitor.

ormance objec			
T CLR Data			
All counters	(All instances	
Select counters	s from list 🦚	Select instances from list	
Client: Total # Client: Total #	failed commands failed connects		Add
otore			
nters Object	Counter	Instance	

Figure 5–9 Add Monitors Step 3 Dialog Box

10. Select Processor in the **Performance object** list.

Figure 5–10 Add Monitors Step 3 with Processors Listed

🚰 Oracle Load Testin	g for Web Applical	tions Web Page Dialog	X
Add Monitor:	Step 3		
Select Counters			
Performance object	:		
Processor			
C All counters		C All instances	
Select counters	from list	Select instances from list	
% C1 Time % C2 Time % C3 Time % DPC Time % Idle Time % Interrupt Time Counters	, ,	D Total	Add
Object	Counter	Instance	
Cancel	Test <	Back Finish	Help

11. Select % Processor Time in the **Select counters from list** group.

12. Select the **All instances** option.

13. Click Add. The counters are added to the Counters list.

Figure 5–11 Add Monitors Step 3 with Processors Selected

🕴 Orac	cle Load Testin	g for Web Applications We	eb Page Dialog 🛛 🗙
Add	Monitor:	Step 3	
Selec	ct Counters		
Perfo	rmance object	ti	
Proc	essor		
C A	ll counters	All ir	nstances
€s	elect counters	from list C Sele	ct instances from list
% P % P % U C1 T	nterrupt Time rivileged Time rocessor Time ser Time fransitions/sec fransitions/sec		
Coun	ters		Add
	Object	Counter	Instance
×ε	Processor	% Processor Time	0
×г	Processor	% Processor Time	_Total
C.	ancel	Test < Back	Finish Help

- **14.** Select Memory in the **Performance object** list.
- **15.** Select Available Kbytes and click **Add**.

Add Monito	r: Step 3	
Select Counter	S	
Performance obj	ject:	
Memory	•	
C All counters	All instar	nces
• Select counte	ers from list 🧖 Select in	stances from list
Write Copies/se Transition Fault		
Transition Fault		Add
Transition Fault		Add
Transition Fault ounters Object	s/sec 🗾	
Transition Fault counters Object X Processor	counter	Instance
Transition Fault Counters Object X Processor	s/sec Counter % Processor Time	Instance O
Transition Fault ounters Object X Processor X Processor	s/sec Counter % Processor Time % Processor Time	Instance O
Transition Fault ounters Object X Processor X Processor	s/sec Counter % Processor Time % Processor Time	Instance O

Figure 5–12 Add Monitors Step 3 with Processors and Memory Selected

16. Click **Finish** to complete adding monitors. ServerStats display a status while it verifies the counters (this may take a few moments).

When the verification is complete, the Configuration window is updated with the list of monitors.

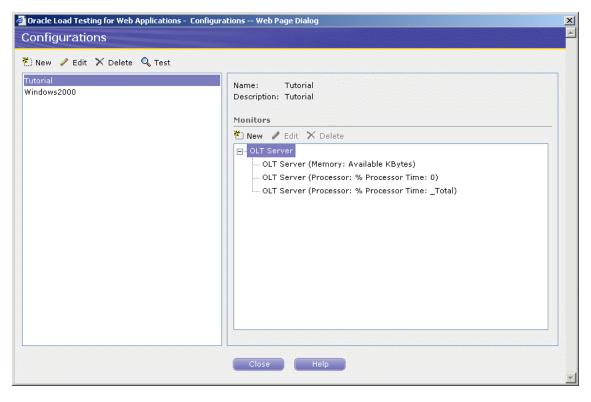


Figure 5–13 Configurations Dialog Box with Defined Monitors

17. Click Test to test the counters.

Figure 5–14 Test Monitors Dialog Box

Results			
Counter Definition	Last Value	Time Stamp E	T OI
OLT Server (Memory: Available KBytes)	562,476	1:34:09 PM	
OLT Server (Processor: % Processor Time: 0)	37.60821	1:34:10 PM	
OLT Server (Processor: % Processor Time: _Total)	37.60821	1:34:10 PM	

- **18.** Review the results to verify the counters are working properly.
- **19.** Click **Close** to exit the test results.

The next example explains the procedures for editing existing ServerStats configurations.

5.3 Example 3: Editing Data Sources

This example shows how to edit existing Oracle Load Testing for Web Applications ServerStats configurations to modify specific counters. The steps in this example are based upon steps completed in the previous example.

- **1.** If not already open, select **Configurations** from the **ServerStats** menu to open the ServerStats Configurations window.
- **2.** Select the Tutorial configuration.
- 3. Click the Memory (Available Kbytes) monitor and click Edit.

Figure 5–15 Edit Monitor Dialog Box

Oracle Load Testi	ing for Web Applications Web Page Dialog	
Edit Monitor		
lonitor		
Name: OLI	Server (Memory: Available KBytes)	
Data Source: Perf	mon (Windows Performance Monitor)	
1onitored Syste	m	🔭 Add
Name:	OLT Server	🥒 Edit
Host Name or IP:	bed-I-risa	
Perfmon (Windo	ows Performance Monitor)	
Username:		
Password:		
Domain Name:		
ata Collector		bbA (🏋
Name:	OLT Server	🥒 Edit
Host Name or IP:	bed-I-risa	
Port:	1099	
collection Interv	val	
Collect data every	30 seconds	
A CONTRACTOR CONTRACTOR CONTRACTOR		
ОК	Cancel Test Hel	

4. Click Add next to Monitored System.

Figure 5–16 Add Monitored System Dialog Box

🚰 Oracle Load Testing	g for Web Applications - Add Monitored System -	- Web 🗙
Add Monitore	d System	
General		
Name:		
Host Name or IP:		
Perfmon (Window	s Performance Monitor)	
Username:		
Password:		
Domain Name:		
ОК	Cancel Test Help	

If you have additional systems to monitor you can specify the information here to add the system to the ServerStats Configuration.

- 5. Click Cancel.
- 6. Change the Collection Interval to 45 seconds.
- 7. Click OK.
- 8. Click Close to close the ServerStats Configurations window.

See the Oracle Load Testing for Web Applications User's Guide for additional information about using the features and options of Oracle Load Testing for Web Applications ServerStats.

5.4 Example 4: Creating a Scenario with Multiple Profiles

This example shows how to create scenarios with multiple virtual user profiles and how to set the attributes for each scenario. It also shows how to specify the reporting options.

5.4.1 Adding a Virtual User Profile to the Scenario

- **1.** Click the **Build Scenarios** tab.
- **2.** Double-click tutor3 in the **Default Profiles** list to add it to the **Configure parameters of the scenario** list.
- **3.** Click the **Configure all parameters** button on the tutor3 line to display the Edit Scenario Details dialog box.
- 4. Change the **#VUs** value to 3.
- **5.** Make sure the **Virtual User Pacing** is set to Recorded and the **Maximum** value is set to 1 second.
- 6. Change the Caching Emulation to Repeat User.
- 7. Change the User Mode to Thick Client.
- 8. Make sure the Use Databanks field is True.
- 9. Leave the default settings for remainder of the attributes and click OK.
- **10.** Click the **Configure all parameters** button on the tutor1 line to display the Edit Scenario Details dialog box.
- **11.** Change the **# VUs** value to 6.
- **12.** Make sure the **Virtual User Pacing** is set to Recorded and the **Maximum** value is set to 1 second and click **OK**.

Notice that each profile in the **Scenario Profiles** list can have a different set of attributes.

5.4.2 Saving Data for Reporting

٩

٩,

The data generated by a Oracle Load Testing for Web Applications Autopilot session can be saved to the Oracle Load Testing for Web Applications database for post-session analysis. The Session Start/Stop options let you specify if Oracle Load Testing for Web Applications should save the data.

13. Select Options from the Tools menu and then select Session Start/Stop.

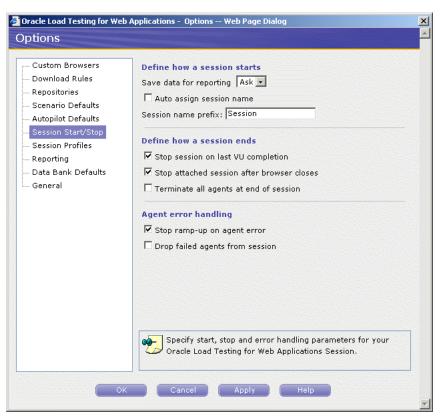


Figure 5–17 Session Start/Stop Options Dialog Box

- 14. Set the Save data for reporting option to Yes.
- **15.** Select the **Terminate all agents at end of session** checkbox.
- **16.** Select **Scenario Defaults**.

Custom Browsers	Main	Show	w Default Value	
Download Rules	# VUs:	•	10	
Repositories	System:	V	OLT Server -	
Scenario Defaults	User Mode:	V	Thin Client 💽	
Autopilot Defaults Session Start/Stop	Iteration Delay:	2	1 (secs)	
Session Stary Stop Session Profiles	VU Pacing (Think Time):		Recorded -	
Reporting	Minimum:		0 (secs)	
Data Bank Defaults	Maximum:		10 (secs)	
General	Use Data Bank:		True -	
	Browser Settings Browser Emulation:		Default	
		e tande	True Line Speed -	
	Connection Speed Emulation:			
	Cache Emulation:		Do Not Cache 💌	
	Use IP Spoofing:		False 💌	
	Use Download Manager:		False 💌	
	Use Wininet:		False 💌	
	Enable Cookies:		True -	
	Eutopcibilitu			
	Specify the main settings and the default values for			tab

Figure 5–18 Scenario Defaults Options Dialog Box

- 17. Set View All Responses in the VU Display section to Always.
- **18.** Scroll down the screen and set **Auto generate timers for all resources** in the Reporting section to True.
- **19.** Click **OK**.

5.4.3 Saving the Scenario

- 20. Select Save As from the Scenario menu.
- **21.** Leave the filename as LoadTest1 and click **OK**.

5.5 Example 5: Running Multiple Profiles

This example shows how to use Oracle Load Testing for Web Applications to run multiple Scenario profiles with different amounts of virtual users and how to view statistical and performance information.

5.5.1 Running the Scenario Profiles Using Autopilot

- **1.** Make sure the Scenario from the previous example is still shown in the Build Scenarios tab.
- 2. Click the Add to Autopilot button on the Scenario tab or the toolbar.
- **3.** Oracle Load Testing for Web Applications automatically opens the Set Up Autopilot tab with the tutor1 and tutor3 Scenario Profiles listed in **Submitted Scenario Profiles** list.

Figure 5–19 Set Up Autopilot Options

Timing and ever	nt controls					
 Start the load test When the start button is pressed After a delay of (hh:mm:ss) At a specific time (hh:mm:ss) Synchronize VU start up 			 Stop the load t When the stop After each use After a delay o At a specific tir 	user (VU) ramp-up tep After every users ⓒ 10 seconds percent ○ 10 iterations		
ServerStats Con	nfiguration					Edit Configurations
Configuration: Description:	<none></none>		×	Monitors:		
Submitted Scen	ario Profiles					
Profiles	VUs	Remaining	Running	with Error	Finished	System
🕼 tutor1	6	6	0	0	0	localhost
🕼 tutor3	3	3	0	0	0	localhost
					🛞 Clear Autopilot	Pause Autopilot

- **4.** Select When the stop button is pressed in the **Stop the load test** group of the Set Up Autopilot tab.
- 5. Enter 3 in the edit box next to **# users** under **Add per step** in the **Virtual User (VU) Ramp-up** section.
- 6. Enter 5 in the edit box next to **# iterations** under After every in the Virtual User (VU) Ramp-up section.
- **7.** In **ServerStats Configuration** section, select Tutorial from the Configuration drop down list to add the configuration to the load test.
- 8. Click Save to save the Rampup Specification in the Scenario file.
- 9. Click the **Run test** button on the Oracle Load Testing for Web Applications toolbar.
- **10.** Oracle Load Testing for Web Applications opens the Save Session data dialog box.
- 11. Click Ok.

The Save Session data dialog box appears because we used the Ask setting in the **Session Start/Stop** options (select **Options** from the **Tools** menu). You can bypass this dialog box and use automatic or default values when running virtual users under routine testing conditions by changing the **Session Start/Stop** options.

12. Watch as the Autopilot starts running the tutor1 and tutor3 Visual Scripts as virtual users. Notice also that tutor3 is playing back records from the Data Bank.

Figure 5–20 Virtual User Grid

U-ID	Profile	Status	Iterations Failed Las	t Run Time	Current Page	System Data Bank	Current Error Previous Error
1	tutor3	Running	9	24.876	[2] Search - Home SuperStore Inc.	localhost Record 5:Phones	
2	tutor1	Think time delay	49	9.463	[2] Kitchens - Home SuperStores Inc.	localhost	
3	tutor1	Running	48	8.892	[4] Electronics - Home SuperStores Inc.	localhost	
4	tutor3	Iteration delay	5	31.816	[3] Results - Home SuperStore Inc.	localhost Record 4:Cabinets	
5	tutor1	Think time delay	28	9.594	[2] Kitchens - Home SuperStores Inc.	localhost	
6	tutor1	Iteration delay	29	9.053	[4] Electronics - Home SuperStores Inc.	localhost	
7	tutor3	Starting				localhost	
8	tutor1	Think time delay	2	8.513	[2] Kitchens - Home SuperStores Inc.	localhost	
9	tutor1	Think time delay	2	7.37	[2] Kitchens - Home SuperStores Inc.	localhost	

Initially, the Autopilot starts only three virtual users. After the first three have completed five iterations, the Autopilot starts another three virtual users. Once the second three virtual users have completed five iterations, the remaining three virtual users start. The **Virtual User (VU) Ramp-up** options of the Autopilot let you control the rate at which virtual users start running.

5.5.2 Viewing Performance Statistics

Oracle Load Testing for Web Applications automatically displays run time graphs in the View Run Graphs tab. Scroll to the bottom of the window until you see the Performance Statistics icon.

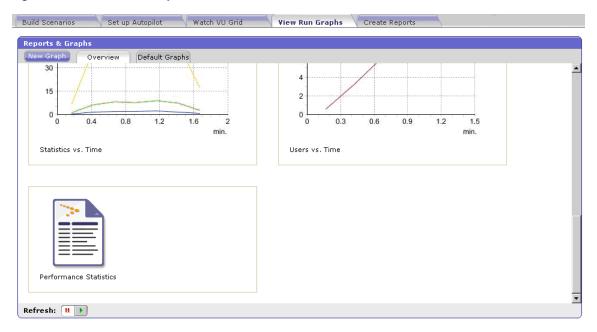


Figure 5–21 View Run Graphs Tab

Click on the icon to view the Performance Statistics report. The Performance Statistics window shows a summary of the performance data for the running virtual users.

Figure 5–22 Performance Statistics Report

ew Graph Overview	Default G	raphs			
View:		D	etail Level:		
Performance Statistics	×		Summary Vie	ew	Include Think Times in Profile timers
SESSION0001					
Name	Value	Min	Max	Avg	
Active Virtual Users	9.345	0	10	6.426	
Virtual Users with Errors	0	0	0	0	
Transactions Per Second	0.433	0.027	0.459	0.311	
Pages Per Second	1.933	0.2	2.4	1.556	
Hits Per Second	2.733	0.533	3.733	2.489	
Kilobytes Per Second	10.621	0.861	12.981	8.361	
	Totals				
Transactions	42				
Transactions with Errors	0				
Pages	210				
Hits	336				
Kilobytes	1129				

The statistics show the values for the following performance categories:

<Session Name> Current

- Active Virtual Users the number of virtual users currently running in the Autopilot.
- Virtual Users with Errors the number of virtual users with errors.
- **Transactions Per Second** the number of times the virtual user played back the Visual Script per second.
- Pages Per Second the number of pages returned by the server per second. A "page" consists of all of the resources (i.e. page HTML, all images, and all frames) that make up a Web page.
- Hits Per Second the number of resource requests to the server per second. Each request for a page, individual images, and individual frames is counted as a "hit" by Oracle Load Testing for Web Applications. If Oracle Load Testing for Web Applications does not request images from the server (as specified in the Download Manager), images are not included in the hit count. The Hits Per Second and Pages Per Second counts will be the same if images are not requested and there are no frames in the page.
- Kilobytes Per Second the number of kilobytes transferred between the server and browser client per second.

<Session Name> Totals

- Transactions the total number of times the virtual user played back the virtual user profile.
- Transactions with Errors the total number of virtual user profile iterations that had errors.
- Pages the total number of number of pages returned by the server.

- Hits the total number of resource requests to the server.
- Kilobytes the total number of kilobytes transferred between the server and browser client.

Performance by Profile and Timer

- <Profile Name> the latest, minimum, maximum, and average performance for the virtual user profile in seconds.
- <Timer Name> the latest, minimum, maximum, and average performance for the server response timers in seconds. Server Response timers are added to Visual Scripts using Oracle Functional Testing for Web Applications.

Performance by Profile and VUs

• <Profile Name> # VUs - shows the time it took to run the virtual user profile with the indicated number of virtual users running. When ramping up virtual users, Performance by Profile and VUs values are added when additional virtual users start running. Once additional Performance by Profile and VUs values are added, the previous Performance by Profile and VUs values are no longer updated. For example, the statistics show elapsed time values for each profile for three, six, and nine virtual users. The profile name> 3 VUs values are updated only while three virtual users are running. Once the Autopilot ramps up to run six virtual users, the profile name> 3 VUs values stop updating and the profile name> 6 VUs values are added and are updated while six virtual users are running. Once the Autopilot ramps up to run nine virtual users, the profile name> 6 VUs values stop updating and the profile name> 9 VUs values are added and are updated while nine virtual users are running.

5.5.3 Viewing Graphs

13. Click the Overview tab in the View Run Graphs tab.

Oracle Load Testing for Web Applications provides several types of graphs that show performance, error, and statistical information for the running virtual users. Click on the graph in the Overview tab to view a larger image in the Default Graphs tab.

14. Select the Performance Vs. Users in the View dropdown of the Default Graphs tab.

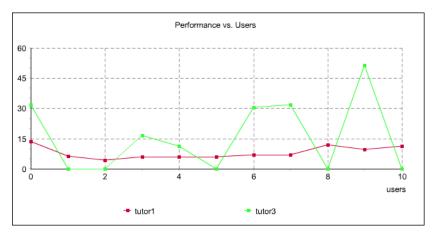


Figure 5–23 Performance Vs. Users Report

This graph shows the average run time for the number of running virtual users in each profile. The plot points represent the Autopilot rampup of virtual users. In

the above graph, the first plot points for each profile shows the average run time while three virtual users were running. Once the Autopilot ramps up to run six virtual users, the plot points for three virtual users are no longer updated.

The second plot points show the average run time while six virtual users were running. Once the Autopilot ramps up to run nine virtual users, the plot points six virtual users are no longer updated.

The third plot points show the average run time while nine virtual users are running. In this example, nine virtual users is the total number of virtual users the Autopilot ramps up to run. The third plot points will be updated continuously while the nine virtual users are running.

15. Select the Users Vs. Time in the View dropdown of the Default Graphs tab.

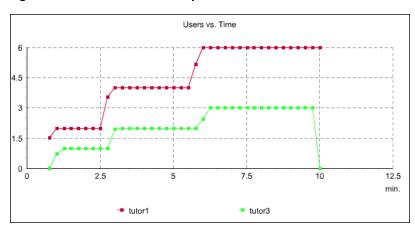
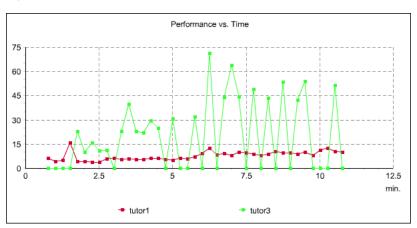


Figure 5–24 Users Vs. Time Report

This graph shows the relative time when the virtual users for each profile started running. The graph represents the Autopilot ramp up times and the number of virtual users ramped up for each profile.

16. Select the Performance Vs. Time in the View dropdown of the Default Graphs tab.

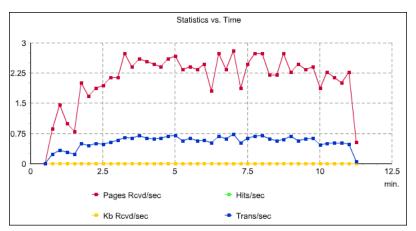
Figure 5–25 Performance Vs. Time Report



This graph shows the average run time for the active virtual users running each profile over time.

17. Select the Statistics Vs. Time in the View dropdown of the Default Graphs tab.





This graph shows averages for virtual user hits, pages, transactions, and Kilobytes per second over time.

The error graphs show percentages of errors vs. virtual users over time.

5.6 Example 6: Controlling Virtual Users

This example shows how to modify individual virtual user attributes, view actions, and stop and abort virtual users in Oracle Load Testing for Web Applications.

1. Make sure the virtual users from Example 3 are still running.

5.6.1 Modifying the Run Attributes

- 2. Click on any virtual user in the virtual user grid.
- 3. Click the right mouse-button to open the popup menu.

Figure 5–27 Virtual User Shortcut Menu



4. Select **Modify Run Attributes**. Oracle Load Testing for Web Applications opens a dialog box for changing the run attributes for the selected virtual user.

View All Responses:	On Error 💌
Show Request Headers:	On Error 💌
Show Response Headers:	On Error 🗾

Figure 5–28 Modify Run Attributes Dialog Box

You can change the attributes of each virtual user individually.

5. Click **Cancel** to close the dialog box.

5.6.2 Viewing Virtual User Actions

6. Select **VU Display** from the **Tools** menu or you can also use the right-click popup menu from the virtual user grid. Oracle Load Testing for Web Applications opens a browser window in which you can view the actions of the virtual user.

🎒 Or	acle Load Testi	ing for Web	Application	ns - Virtual	User Display	y - Microsofi	t Internet E	xplorer					
V	'irtual Use	r Displa	ay - SES	SIONOO	08								
6	i 🖬 × 🛛 v	'iew: 🔳	D Auto	Display:) II			Items 🛛	1 to 39	of 39 🚺 🖣		Refresh:	On Off
×													Ì
Use	SuperSearch	n 🗵											
	ome <u>Stor</u> egister	res		Ple	ase enter	the follow	ing produ	ct infor	nation:				
\square_{De}	epartments: 🗆				Product N	ame:							
<u>Ki</u> <u>Ba</u>	itchen &	<u>Lighting</u> <u>Electronic</u>	<u>:s</u>		I'm most in	nterested in	⊙ Low	vest Cos Sea	Ū	est Quality	C Bran	d Names	
	×											_	
	Current Page Home SuperS ()		JRL ile:C:\Orac	eATS\OFT	\RSWDemo\	search.htm	-	/pe ¥U-1 3	51	Cabinets	k Error	Time 2:00:57 PM	Size 9130
34	tutor3[3] Res Home SuperS ()	Store Inc. T	ïle:C:/Orac 'srchres.htn		/RSWDemo		unknown	3	51	Record 4: Cabinets		08/12/2008 2:00:57 PM	29592
35	tutor3[1] Wel Home SuperS Inc. ()		ile:C:\Orac	eATS\OFT	\RSWDemo\	index.htm	unknown	3	52	Record 2: Paints		08/12/2008 2:01:02 PM	7226
36	tutor3[2] Sea Home SuperS ()		ile:C:\Orac	eATS\OFT	\RSWDemo\	search.htm	unknown	3	52	Record 2: Paints		08/12/2008 2:01:02 PM	
🙆 Dis	splaying: 36							******				Local intranet	******

Figure 5–29 Virtual User Display Window

4

7. Click the Navigate to Previous Page toolbar button. The viewer shows only the previous page.

- **8.** Click the Navigate to Next Page toolbar button. The viewer shows only the next page.
- **9.** Click the Auto Mode toolbar button. The view shows new pages accessed by the virtual user as they arrive to the viewer.
- **10.** Click the Stop Accepting New Pages toolbar button. The viewer stops accepting pages from the virtual user.

Note: Because of the speed at which new pages arrive in the viewer, it may take a few moments for cached pages to stop appearing.

11. Close the window to exit the viewer.

5.6.3 Stopping an Individual Virtual User

- **12.** Click on any virtual user in the virtual user grid.
- **13.** Click the right mouse-button to open the popup menu.
- **14.** Select **Stop**. Oracle Load Testing for Web Applications stops running the selected virtual user. The virtual user will complete the current Visual Script iteration and then stop.

5.6.4 Aborting an Individual Virtual User

- **15.** Click on any virtual user in the virtual user grid.
- **16.** Click the right mouse-button to open the popup menu.
- **17.** Select **Abort**. Oracle Load Testing for Web Applications aborts running the selected virtual user without completing the current visual script iteration.

5.6.5 Stopping All Virtual Users

4

b

•

11

18. Click the Stop toolbar button to stop all virtual users. The virtual users will complete the current visual script iteration and then stop.

5.6.6 Aborting All Virtual Users

19. Click the Abort toolbar button to abort all virtual users. The virtual users will abort the virtual user without completing the current visual script iteration.

5.7 Example 7: Generating Reports

This example explains the automatic report generation features of Oracle Load Testing for Web Applications. The data collected by Oracle Load Testing for Web Applications and Oracle Load Testing for Web Applications ServerStats while the Autopilot is running virtual users is saved to a database when the **Save Data for Reporting** option in the Oracle Load Testing for Web Applications Session Start/Stop options is set to Yes or Ask. You can use Oracle Load Testing for Web Applications to analyze the data and generate a variety of graphs and reports.

5.7.1 Generating Reports from Oracle Load Testing for Web Applications

This example shows how to use previously saved data to create custom reports and to view Session and Scenario reports.

1. Select the Create Reports tab.

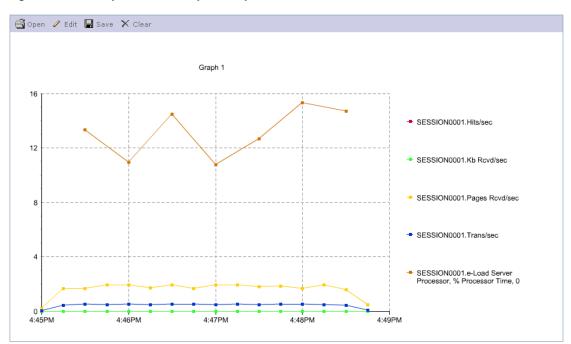
- **2.** The Graph 1 tab is displayed with a blank graph. Select Session0001 from the **Session** list. The data categories appear in the Available Data Series list.
- 3. Click Show All.

Figure 5–30 Create Reports Tab Filters

Build Scenarios Set up Autopilot Watch VU Grid View Run Graphs Create Reports
Reports & Graphs
New Graph Overview Reports Graph 1 🗴
Session: SESSION0001
Available Data Series: 🔽 Show All 🛱 Open 🖉 Edit 🖬 Save 🗡 Clear X-Scale: Absolute Time 🔽
Hits/sec KB Rcvd/sec Number of VUs Pages Rcvd/sec Trans Failed/sec Trans/sec VUs with Errors Image: tutor1 Image: tutor3
ServerStats Monitors
Export to: PDF I XLS S DFG

- **4.** Double-click Hits/Sec, kb Rcvd/Sec, Pages Rcvd/Sec, and Trans/sec at the top of the **Available Data Series** field to add the counters to the graph. These are the overall counters. You can also select them and click the **Add Data Series** button.
- **5.** Expand the ServerStats Monitors node then double-click the Oracle Load Testing for Web Applications Server (Processor, % Processor Time, 0) node to add it to the graph.

Figure 5–31 Sample Session Report Graph



The legends show which color line represents which virtual user profile, Visual Script page, and Oracle Load Testing for Web Applications ServerStats counter. The legends for Oracle Load Testing for Web Applications data show the session, the virtual user profile, and the Visual Scripts page in the form session.profile.page[#]. The legends for ServerStats data show the session, counter object, counter instance and counter in the form session.object.instance.counter.

You can export the data to an HTML file, a comma separated value file, or a Microsoft Excel Workbook file.

5.7.2 Opening the Chart in Microsoft Excel

Note: Skip this section if you do not have Microsoft Excel installed on your system.

🗷 XLS

- 6. Click the Export to Excel button.
- 7. The File Download dialog box is displayed. Click Save.
- **8.** The Save As dialog box is displayed. Select a location to save the report and click **Save**.
- 9. The Download Complete dialog box is displayed. Click Open.

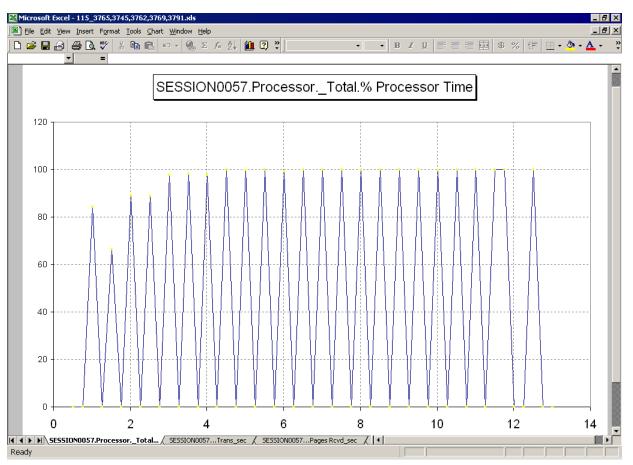


Figure 5–32 Sample Excel Report Graph

- **10.** The workbook contains a chart sheet and a worksheet. You can use the features and capabilities of Microsoft Excel to change the chart format.
- **11.** Click the Data Table tab to view the actual data values.

-		ew Insert Format Tools Da		4.2								1 1	
]	🛎 🖬 🔒		κη τ 🍓 Σ f≈ ੈ‡	1	♀ Arial	• 1	0 - B	Ι	[E E		\$?	6 💷 🗌	• 🕭 • <u>A</u>
	D1		157Pages Rcvd/sec							_			-
ļ	A	B	C C	05001	D			E	- ,	0500	F		G
	Time	SESSION0057Hits/sec		1 <u></u>	JNUU57Pages Ro		SESSIONU	057			UNUUS		rlotal.% F
	0.516667	0			0.000	0				0		0	
	0.766667	C	-			666675			23947645		0.	~	
	1.016667				1.466	666698 1			33502015 28154495		84	4.19095612 0	
	1.516667		-		0.000	000012			20154495 23238563		co	0 6.43212891	
	1.766667				0.000	2			23230303 48954930	-	00	0.43212091 N	
	2.016667				1 000	∠ 666627			40954930 43987563	-		0 39.0617218	
	2.266667					666675			43907563).4892446	-	0	1210017210 N	
	2.200007					333278			48371940 48371940		ç	0 38.9006424	
1	2.516667	0	-			333445			4837 1940 53111767	-		10.9006424 N	
1	3.016667					333445			53111767 58838576	-	00	7 65746307	
	3.266667		-			333349		-).6477062		57	.00740007 N	
ł	3.516667					000095			63796138		qg	3 34191132	
1	3.766667		-			999905			69262671	-		.54151132 N	
1	4.016667	0	-			333302		-	63456779	-	93	7.81159973	
1	4.266667	0				666698			60597008			01100070	
	4.516667		-			000095			62514448	-	qq	3.90200043	
1	4 766667	0				999905			67264789			04002000-0	
1	5.016667					666746			68885552	-	99	9.66309357	
1	5.266667					333254			56641435	-		0	
1	5.516667	0				000095			63068497	-		100	
1	5.766667	0	-			333254		-	56895285			0	
1	6.016667	0				666698			57544463		99	9.23466492	
ĺ	6.266667	0	-			999952			51634198	-		0	
i	6.516667	0				333349			67313420			100	
ĺ	6.766667				2.333	333254			60878819	-		0	
1	7.016667	0	C		2.799	999952		1).7296666	5		100	
l	7.266667	0	0		1.866	666675		0	51188540	5		0	
1	7.516667	0	C		2.466	666698		0	62380623	8		100	
	7.766667	0	C		2.733	333349		0	67290258	4		0	
ĺ	8.016667	0	C		2.733	333349		0	69617110	6		100	
	8.266667	0	C		2.200	000048		0	60862326	6		0	
I	8.516667	0	C		2.200	000048		1).5642579	2		100	
	8.766667	0	C		2.733	333349		0	59032785	9		0	
đ		SSION0057Kb Rovd sec 🔏	SESSION0057Hits sec 🔏 Gr	1.4.) 0	ata Table /			-					F

Figure 5–33 Sample Excel Data Table Report

- **12.** Row one contains the counter names. Subsequent rows contain the actual data values for the chart.
- 13. Select Exit from the File menu to close Microsoft Excel.

5.7.3 Viewing Scenario and Session Reports

The report shows the current and total performance over time for the Oracle Load Testing for Web Applications scenario. The report also shows the Oracle Load Testing for Web Applications scenario settings used for the session.

- 14. Oracle Load Testing for Web Applications also generates textual reports for Oracle Load Testing for Web Applications Scenario settings and Oracle Load Testing for Web Applications and Oracle Load Testing for Web Applications ServerStats session data.
- **15.** Select the **Reports** tab.
- **16.** Select the session for which you want to view the report from the **Session** dropdown list and click **Generate**. Oracle Load Testing for Web Applications displays the report.

Reports & Graphs				
New Graph Overview	Repo	orts	Graph 1	
Session: SESSION0001	💽 Repo	rt: Sessior	n Report 🗾 🗙 Delete 🛛 🛛 Generate	
Session Performa	ince Rej	oort - S	SESSION0001	
Start Time:	1/15/20	07 14:24:13	3	
End Time:	1/15/20	07 14:27:21	1	in the second
Duration:	00:03:0	8 (188 sec)		
Name	Min	Max	Avg	
Active Virtual Users	0	8.723	4.626	
Virtual Users with Errors	0	0	0	
Transactions Per Second	0.009	0.483	0.246	
Pages Per Second	0.133	1.933	0.985	
Hits Per Second	0	0	0	
Kilobytes Per Second	0	0	0	
	Totals			
Transactions	48			
Transactions with Errors	0			
Pages	192			
Hits	0			
Kilobytes	0			

Figure 5–34 Sample Session Performance Report

- **17.** Scroll to the end of the Session Performance Report to view the Oracle Load Testing for Web Applications Scenario Report.
- **18.** You can print the report by clicking the **Print Friendly** button and selecting **Print** from the **File** menu in the browser window.

5.8 Example 8: Creating User-Defined Profiles

This example explains how to create user-defined virtual user profiles in Oracle Load Testing for Web Applications.

- **1.** Select **User Defined Profiles** from the **Manage** menu. Oracle Load Testing for Web Applications opens a dialog for managing profiles.
- 2. Click New.

Repository: Default Workspace: RSWDemo	Name: Name: Add Sync 🖉 Edit 🗙 Delete	
🔀 ChartPortfolio	🗆 Prolog	
🚱 tutor1	E Run	
🕼 tutor2 🕼 tutor3	🖃 Epilog	
tutor4	Errors	+++
	d >	e Load

Figure 5–35 Edit User Defined Profiles Dialog Box

- **3.** The dialog box shows the section tree, the available Visual Scripts, and the default synchronization points.
- **4.** Select the workspace where you want to save the profile and enter a name for the profile in the **Name** editbox.
- **5.** The profile sections tree allows you to specify which Visual Scripts and synchronization points to include in the Sections tree of the profile.

Prolog - the Visual Scripts in this section play back only once at the beginning of the Scenario run. An example of what you may add to this section is a login script.

Run - the Visual Scripts in this section iterate over as many times as is specified in the Autopilot. An example of what you may add in this section is the business transaction that you wish to load test.

Epilog - the Visual Scripts in this section play back only once at the end of the Scenario run. An example of what you add to this section is a logoff script.

Errors - the Visual Scripts in this section play back only if an error occurs during the Scenario run. An example of what you may add to this section is a visual script that resets your application on an error.

5.8.1 Adding Visual Scripts to the Sections Tree

- 6. Select the section in the tree where you want to add a Visual Script.
- **7.** Double-click the Visual Script to add to the section or select the script and click the **Add** button.

The Visual Script appears as a node of the tree.

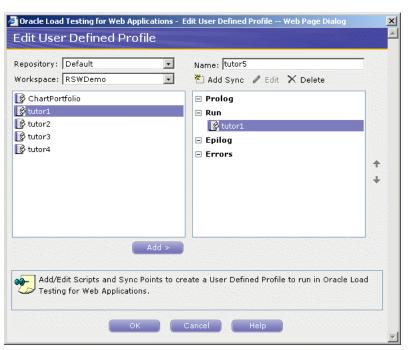


Figure 5–36 Script Added to Run Section of User Defined Profile

8. Repeat steps 4 and 5 to add additional Visual Scripts to the Sections tree.

5.8.2 Adding Synchronization Points to the Sections Tree

A sync point allows multiple virtual users to synchronize their actions and interactions with the application under test. Sync points provide the ability to create realistic multi-user situations that may expose resource conflicts such as deadlocks. When you specify a sync point, multiple virtual users executing the script will reach this sync point at various times depending on a number of factors (for example, the speed of the machine).

Sync points cause each virtual user to wait until all virtual users have reached that sync point. Each of the virtual users notifies the master upon reaching the sync point. The master waits for all of the virtual users to notify it and then issues the go-ahead for all the virtual users to continue past that sync point.

9. Select the section in the tree where you want to add a Sync point and click **Add Sync**.

The Sync point appears as a node of the tree.

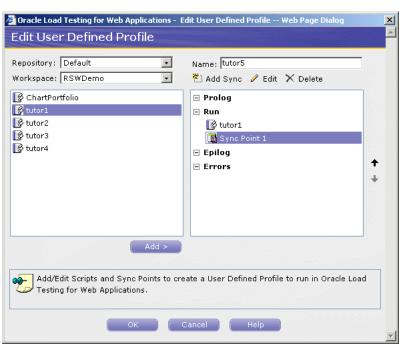


Figure 5–37 Sync Point Added to Run Section of User Defined Profile

10. Repeat steps 7 and 8 to add additional Sync points to the Sections tree.

5.8.3 Moving Items in the Sections Tree

When you have multiple items under any section of the tree, you can move the items up and down within that section.

- **11.** Select the item to move in a section.
- **12.** Click the up or down button as appropriate.

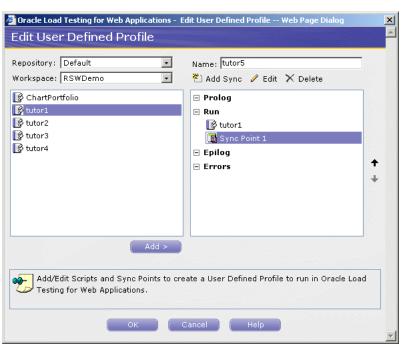


Figure 5–38 Edit User Defined Profile Dialog Box

- 13. Click the OK button when you finish defining the profile.
- **14.** The new profile appears in the **Select scripts and user-defined profiles** list of the **Build Scenarios** tab.

Select scripts & user-defined profiles
Repository: Default 🔹 Workspace: RSWDemo 💽
 ChartPortfolio tutor1 tutor2 tutor3
l∲ tutor4 i∰ tutor5
Add to scenario

You can include user-defined profiles as part of the Scenario Profiles the same way you use the default profiles.

5.8.4 Editing User-Defined Profiles

After you have created a user-defined virtual user profile, you can make changes to the profile at any time.

15. Select **User Defined Profiles** from the **Manage** menu.

		<u></u>	-
orkspace: R	SWDemo		-
)New 🥒 Ec	lit 🗙 Delete		
ame			
tor5			
		ofiles for con	ohinina
	edit and delete l nto modular tran	011105 101 001	(ib) ming

Figure 5–40 User Defined Profile Manager Dialog Box

16. Select the profile you want to edit and click **Edit**. Oracle Load Testing for Web Applications opens a dialog for editing the sections tree of the profile. The dialog box shows the current sections tree, the available Visual Scripts, and the default synchronization points.

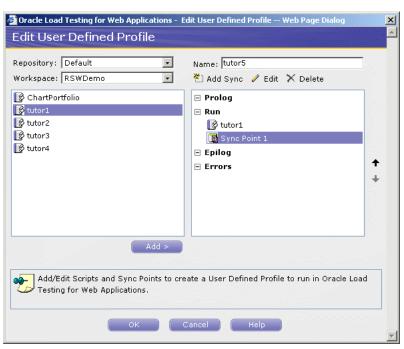


Figure 5–41 Edit User Defined Profile Dialog Box

- **17.** Click the Plus icon to expand the nodes of the tree.
- **18.** Use the arrow buttons as necessary to add items to the sections of the tree. Select an item and click **Delete** to remove it from the sections tree.
- **19.** Click the **OK** button when you finish editing the profile.
- 20. Select Exit to close Oracle Load Testing for Web Applications.
- **21.** Click **No** if asked to save the scenario.

This completes the Oracle Load Testing for Web Applications tutorial. See the *Oracle Load Testing for Web Applications User's Guide* for additional information about load testing and using Oracle Load Testing for Web Applications.

Index

Α

accept a new baseline script, 3-12 Accept Tested Page, 3-12 Active Virtual Users performance statistics, 5-18 Add Binding option, 3-25 Add Mapping option, 3-23 Add per step option, 5-16 Add to Autopilot option, 5-3 Add to Scenario option, 5-3 Add Variable option, 3-22 address line, 2-4 Address node, 3-4 Allow Blocked Content option, 3-9 Auto generate timers for all resources option, 5-15 automatic tests, 3-5 autopilot adding scenarios, 5-15 running scenario profiles, 5-15 setting up, 2-10, 5-4 starting, 5-4 stopping, 5-4 Available Data Series settings, 5-24

В

browser pane, 2-4, 2-5 Build Scenarios overview of, 2-9

С

Caching Emulation setting, 5-13 Clear Results Window option, 3-13, 3-29, 3-36 Collection Interval, 5-13 comma-delimited files, 3-25 configurations adding, 5-5 testing, 5-11 Configure Parameters list, 5-3, 5-13 Content Tests enabling, 3-6 CSV files, 3-25

D

Data Bank Fields list, 3-25

Data Bank file opening, 3-25, 3-32 selecting, 3-24, 3-32 using in load test, 5-13 Data Bank Wizard auto-binding variables, 3-31 auto-mapping variables, 3-30 binding variables, 3-23, 3-31 Fetch Record option, 3-26 going to a specific record, 3-28 mapping variables, 3-21, 3-30 using with forms, 3-20, 3-29 Data Binding tab, 3-23 Data Collector, 5-7 Data Mapping tab, 3-22 Data Sources adding, 5-5 editing, 5-12 Default Profiles list, 5-13 default tests modifying, 3-6 Different HTML Script node, 3-11 Discard Tested Page, 3-13 Duration node, 3-4

Е

email notifications, 4-3 Epilog section in user-defined profiles, 5-29 error notifications, 4-2 Errors section in user-defined profiles, 5-29 expanding script pages, 3-3 exporting reports, 5-25

F

failure markers, 3-10 Failures Only option, 3-34 features functional testing, 1-2, 2-4 Job Scheduler, 1-4, 2-6 load testing, 1-6, 2-9 Fetch Record option, 3-26, 3-33, 3-34 Find Next Failure option, 3-11 Form Element script nodes, 3-17 Form Element tests inserting, 3-16 properties, 3-17 script node, 3-18 Frame node, 3-4

G

Goto Page option, 3-14, 3-18 graphs opening in Microsoft Excel, 5-25 Performance vs. Time report, 5-20 Performance vs. Users report, 5-19 sample session report, 5-25 Statistics vs. Time report, 5-21 Users vs. Time report, 5-20 viewing, 5-17, 5-19

Η

Hits Per Second performance statistics, 5-18 Hits performance statistics, 5-19 HTML node, 3-4

Ignore This Failure option, 3-11 Images node, 3-4 Insert Table Test option, 3-18 installation, 2-1 introduction, 1-1 iterations playing back with, 3-27

J

Job pane, 2-7 Job results pane, 2-7 Job Scheduler activating schedules, 4-5 deactivating schedules, 4-6 editing jobs, 4-6 editing schedules, 4-7 overview of, 2-6 playing back jobs, 4-5 saving schedules, 4-5 selecting scripts, 4-1 specifying notifications, 4-2 specifying schedule time, 4-4 starting, 4-1 tutorial, 4-1 using schedules, 2-8, 4-4 viewing results, 2-7, 4-5 Job Scheduler Wizard, 2-8, 4-2 starting, 4-1

Κ

Kilobytes Per Second performance statistics, 5-18 Kilobytes performance statistics, 5-19

L

Links node, 3-4 load scenarios building, 2-9, 5-2 setting up autopilot, 2-10, 5-4 Log Message Format notification, 4-2

Μ

Mail Server configuring, 4-3 Maps node, 3-4 Master node, 3-11 Maximum Time Allowed for Playback setting, 3-16 Minimum Time setting, 3-16 modifying default tests, 3-6 Monitored System, 5-7 monitors adding, 5-6 editing, 5-12

Ν

New Links script node, 3-11 New Script option, 3-14 Next Difference option, 3-12 notifications specifying for email, 4-3 specifying for results log, 4-2

0

Oracle Functional Testing for Web Applications tutorial, 3-1 Oracle Load Testing for Web Applications tutorial, 5-1

Ρ

Page Content Tests Manager, 3-6 Pages Per Second performance statistics, 5-18 Pages performance statistics, 5-18 Parameters, 3-21, 3-26, 3-29 Data Bank script node, 3-27 script node, 3-21 Pass when property, 3-15, 3-27, 3-29 Perfmon (Windows Performance Monitor), 5-7 selecting counters, 5-8 specifying instances, 5-9 Performance object list, 5-8 performance statistics categories of, 5-18 viewing, 5-17 playback analyzing failures, 3-10, 3-28, 3-35 playing with iterations, 3-27, 3-34 starting, 3-8 starting jobs, 4-5 using all records, 3-28 using Data Bank, 3-28, 3-35

viewing results, 2-6, 3-10 playing back Visual Scripts, 3-7 Profile performance statistics, 5-19 profiles adding scripts, 5-29 adding synchronization points, 5-30 creating user-defined, 5-28 editing, 5-33 moving scripts, 5-31 Prolog section in user-defined profiles, 5-29

R

recording Visual Scripts, 3-1 starting, 3-2, 3-21, 3-29 stopping, 3-3, 3-21, 3-29 reporting saving data, 5-13, 5-23 setting options, 5-14 reports creating, 2-12 exporting, 5-25 generating, 5-23 Performance vs. Time report, 5-20 Performance vs. Users report, 5-19 sample session report, 5-25 Statistics vs. Time report, 5-21 test results, 3-9 Users vs. Time report, 5-20 viewing scenario reports, 5-27 viewing session reports, 5-27 resizing Visual Scripts, 3-3 **Resource Validation** overview of, 3-8 running automatically, 3-7 Results Log appending to, 3-7 disabling, 3-7 enabling, 3-7 saving, 3-29, 3-36 **Results Report** overview of, 3-9 running automatically, 3-7 run graphs viewing, 2-11, 5-17 Run section in user-defined profiles, 5-29 Run Test button, 5-4, 5-16

S

Save data for reporting option, 5-14 Save Output Log As option, 3-13 Save Schedule As option, 4-5 saving Results logs, 3-29 scenarios adding Virtual User profiles, 5-13 running multiple profiles, 5-15 running tests, 5-4, 5-16 saving, 5-15 setting up autopilot, 5-4

specifying profiles, 5-2 viewing reports, 5-27 schedule window, 2-8, 4-5 schedules activating, 4-5 creating, 4-1 deactivating, 4-6 editing, 4-7 saving, 4-5 specifying, 4-4 Script Content Tests options, 3-6 Scripts node, 3-4 Select Property setting, 3-17 Select Table Element, 3-18 Select Test Criteria setting, 3-17 Selected text is present setting, 3-15, 3-27, 3-29 Send HTML Summary option, 4-3 Send Text Log option, 4-3 Server Response tests inserting, 3-15 properties, 3-16 script node, 3-16 ServerStats adding monitors, 5-6 configuring, 5-5 editing configurations, 5-12 overview of, 2-13 testing configurations, 5-11 using configurations, 5-16 sessions saving data, 5-16 starting and stopping options, 5-14, 5-16 viewing reports, 5-27 setting script options, 3-6 Specify Expression setting, 3-17 status line, 2-4 submit my entry button, 3-14 Submitted Scenario Profiles list, 5-3 synchronization points adding to user-defined profiles, 5-30 moving in user-defined profiles, 5-31

Т

Table Tests creating, 3-19 inserting, 3-18 script node, 3-20 selecting elements, 3-19 selecting fields, 3-20 Terminate all agents at end of session option, 5-14 test cases adding to Visual Scripts, 3-13 inserting Form Element tests, 3-16 inserting Server Response tests, 3-15 inserting Table Tests, 3-18 inserting Text Matching tests, 3-14, 3-27 test failures, 3-10 analyzing, 3-10 finding different images, 3-13

ignoring, 3-11 rejecting, 3-12 Tested node, 3-11 Text Matching tests inserting, 3-14, 3-27, 3-29 properties, 3-15 script node, 3-15 Timer performance statistics, 5-19 Transactions Per Second performance statistics, 5-18 Transactions performance statistics, 5-18 Transactions with Errors performance statistics, 5-18 tutorial initializing, 3-1 Job Scheduler, 4-1 Oracle Functional Testing for Web Applications, 3-1 Oracle Load Testing for Web Applications, 5-1

U

URL dropdown, 2-4 Use Databanks setting, 5-13 User Mode setting, 5-13 user-defined profiles adding scripts, 5-29 adding synchronization points, 5-30 creating, 5-28 editing, 5-33 moving scripts, 5-31 using Data Banks, 3-20, 3-29

V

value, 5-13 variables auto-binding in Data Bank Wizard, 3-31 auto-mapping in Data Bank Wizard, 3-30 binding in Data Bank Wizard, 3-23, 3-31 mapping in Data Bank Wizard, 3-21, 3-30 View All Responses option, 5-15 viewing failure markers, 3-10 viewing in scripts, 3-21, 3-26, 3-29 Virtual User (VU) Ramp-up options, 5-16, 5-17 Virtual User Display, 5-22 Virtual User Grid, 2-11 Virtual User Pacing setting, 5-13 virtual users aborting all, 5-23 aborting individual, 5-23 adding per step, 5-16 controlling, 5-21 creating user-defined profiles, 5-28 editing user-defined profiles, 5-33 modifying run attributes, 5-21 navigating, 5-22 ramping up, 5-4 setting number of, 5-13 stopping all, 5-23 stopping individual, 5-23 viewing, 2-11

viewing status, 5-5 viewing user actions, 5-22 watching, 5-22 watching currently running, 2-11 Virtual Users with Errors performance statistics, 5-18 Visual Script Parameters viewing, 3-21, 3-26, 3-29 Visual Scripts adding test cases, 3-13 analyzing playback failures, 3-35 collapsing, 2-5 displaying properties, 3-4 expanding, 2-5 expanding pages, 3-3 playing back, 3-7 recording, 3-1 rejecting problems, 3-12 resizing, 2-5, 3-3 saving, 3-3, 3-29 test flags, 2-5 viewing, 2-4 VUs performance statistics, 5-19

W

Watch Virtual User Grid tab, 2-11