WebLOAD User's Guide

Version 8.0

RadView Software

The software supplied with this document is the property of RadView Software and is furnished under a licensing agreement. Neither the software nor this document may be copied or transferred by any means, electronic or mechanical, except as provided in the licensing agreement. The information in this document is subject to change without prior notice and does not represent a commitment by RadView Software or its representatives.

WebLOAD User's Guide

© Copyright 2007 by RadView Software. All rights reserved.

January, 2007, RadView Publication Number WL-0305-UG72

WebLOAD, TestTalk, Authoring Tools, ADL, AppletLoad, WebFT, and WebExam, are trademarks or registered trademarks of RadView Software IBM, and OS/2 are trademarks of International Business Machines Corporation. Microsoft Windows, Microsoft Windows 95, Microsoft Windows NT, Microsoft Word for Windows, Microsoft Internet Explorer, Microsoft Excel for Windows, Microsoft Access for Windows and Microsoft Access Runtime are trademarks or registered trademarks of Microsoft Corporation. SPIDERSESSION is a trademark of NetDynamics. UNIX is a registered trademark of AT&T Bell Laboratories. Solaris, Java and Java-based marks are registered trademarks of Sun Microsystems, Inc. HP-UX is a registered trademark of Hewlett-Packard. SPARC is a registered trademark of SPARC International, Inc. Netscape Navigator and LiveConnect are registered trademarks of Netscape Communications Corporation. Any other trademark name appearing in this book is used for editorial purposes only and to the benefit of the trademark owner with no intention of infringing upon that trademark.

For product assistance or information, contact:

Toll free in the US:	1-888-RadView
Fax:	(781) 238-8875
World Wide Web:	www.RadView.com

North American Headquarters:

RadView Software Inc.
7 New England Executive Park
Burlington, MA 01803
Email: sales@RadView.com
Phone: (781) 238-1111

International Headquarters:

RadView Software Ltd.
14 Hamelacha Street, Park Afek
Rosh Haayin, Israel 48091
Email: sales@RadView.com
Phone: +972-3-915-7060
Fax: +972-3-915-7683

Table of Contents

1. Introduction 15

WebLOAD Documentation 15

Typographical Conventions 16

Where to Get More Information 17

Online Help 17

Technical Support 17

Technical Support Web Site 18

2. Introducing WebLOAD 19

3. Getting Started 21

Welcome to WebLOAD 21

How Does WebLOAD Work? 22

WebLOAD Architecture 23

How are Tests Defined? 25

What is the Product of Your WebLOAD Testing? 26

WebLOAD Offers Two Client Type Modes 26

WebLOAD Features 28

4. WebLOAD Quick Start 35

Quick Start Steps 35

Creating an Agenda 36

Configuring a Load Template using the WebLOAD Wizard 40

Opening the WebLOAD Wizard 40

Selecting an Agenda and a Performance Statistic to Monitor 41
Selecting a Host 42
Scheduling the Test 42
Setting Session Options 43
Running the Test 44
Analyzing Test Results 45
Viewing the Results in Real Time 46
Creating Integrated Reports 46
Viewing Data Drilling Reports 47
Viewing Results with WebLOAD REPORTER 48
5. WebLOAD Installation and Configuration 51
Installing WebLOAD 51
License Registration and Update 52
Standard License Registration 53
Floating License Server 57
Uninstalling WebLOAD 61
Upgrading WebLOAD 62
Communication Between Host Computers 62
SNMP 62
TestTalk for Windows 64
Pinging a Host 67
Troubleshooting Communications 68
Configuration 69
Required Software for each Load Session Component 70
Example of a Test Configuration 71
Sharing an Agenda 72
Working with a Firewall 73
6. WebLOAD Console Features 77
The WebLOAD Console 77
Opening the Console 78
Console Elements 79
Title Bar 79
Menu Bar 80
Context Menus 80
Toolbar 80

Status Bar 81
The Session Tree 81
The Results Window 89
Event Viewer 94
Log Window 94
The WebLOAD Console Shortcut Screen 95
Edit Mode and Run Modes of Operation 96
The Console in Edit Mode 96
The Console in Run Mode 97
The Console Menu Options 98
File Menu 98
Session Control Menu 99
Reports Menu 100
View Menu 101
Tools Menu 104
Window Menu 105
Help Menu 105
The Console Toolbar 105
Toolbar Buttons 106
Extended Toolbar Segments 107
7. Creating Load Templates 111
WebLOAD Options for Creating Load Templates: 111
8. Planning a Test Session 113
Load Session Workflow 113
Forming a Test Plan 115
Planning a Test Program 116
Creating the Test Agendas 116
Examining Your Test Resources 117
Starting WebLOAD 117
9. Connecting to WebRM 121
Getting Started 121
About WebRM 122
How Does WebRM Work 123
Working with WebRM 123
Requesting Resources from WebRM 123
Updating the Number of Requested Resources 124

10. Managing Load Templates and Load Sessions 125

Managing Load Templates 125
Creating a New Load Template 126
Saving a Load Template 127
Opening a Saved Load Template 127
Saving Additional Information with the Load Template 127

Managing Load Sessions 129
Saving a Load Session 129
Saving Additional Information for the Load Session 130

11. Creating Load Templates with the WebLOAD Wizard 133

Using the WebLOAD Wizard to create Load Templates 133

The WebLOAD Wizard Workflow 134

Opening the WebLOAD Wizard 135

Selecting an Agenda or Mix 136

Selecting an Agenda 137

Selecting a Mix 138
Creating a Mix 139

Selecting Host Computers 141
Adding Host Computers 142
Designating a Host Computer as a Load Machine 142
Designating all Host Computers as Load Machines 143
Designating a Host Computer as a Probing Client Machine 143
Replacing a Load Machine or Probing Client Machine 143

Scheduling the Load Session 144
Distributing the Load Equally 145
Manually Scheduling the Load to Generate on Each Load Machine 148
Scheduling Models 150

Completing the WebLOAD Wizard 158

After Completing the WebLOAD Wizard 159
Editing the Load Template 159
Viewing the WebLOAD Wizard Default Report 159
Modifying the Host Selection 160
Modifying the Load Schedule 160

12. Creating Load Templates with Cruise Control 161
Using Cruise Control 161
How does Cruise Control Work? 162
The Cruise Control Wizard Workflow 163
Opening the Cruise Control Wizard..... 163
Selecting an Agenda or Mix 165
Running Single Agendas..... 166
Running a Mix of Agendas 167
Defining a Mix 169
Selecting Host Computers 172
Adding Host Computers to the Host Computers List..... 173
Designating a Host Computer as a Load Machine 173
Designating all Host Computers as Load Machines..... 174
Designating a Host Computer as a Probing Client Machine..... 174
Deleting a Host Computer 174
Defining Performance Goals 175
About the Goal Definition Dialog Box 175
Adding Performance Goals 177
Deleting a Goal 180
Setting and Using Defaults..... 180
Defining the Acceleration Rate 182
Defining WebLOAD's Action Once the Goals are Reached 183
Finishing the Cruise Control Wizard 184
Viewing the Cruise Control Default Report..... 186
13. Configuring a Load Template Manually 189
The Manual Test Configuration Workflow..... 190
Starting a New Template Manually 190
Selecting an Agenda..... 191
Creating Load Templates with One or More Single Agendas 191
Creating Load Templates with an Agenda Mix..... 192
Selecting Hosts 193
Opening the Host Selection Dialog Box 193
Defining Host Computers 194

Defining Load Machines 194
Defining Probing Client Machines 195
Scheduling the Test Session 195
Creating a Load Schedule Manually 196
Creating a Load Schedule with the Load Profiler 197
Completing Your Manual Load Template Configuration 198
14. Defining Test Session Options 201
Session Options 201
15. Setting Global Options 203
How to Set Global Options 203
The General Tab 205
The Export Tab 208
The File Locations Tab 211
The Data Drilling Tab 212
The Advanced Tab 213
The Defect Tracking Tab 215
The Notification Manager 215
16. Setting Agenda Options 221
About Agenda Options 222
Modifying Agenda Options 224
Setting the SSL Options 225
Client Type Settings 225
Selecting HTTP Client Type Mode 227
Selecting Browser Client Type Mode 227
Setting Sleep Time Playback Settings 227
Setting the Browser Cache 229
Configuring Browser Parameter Settings 231
Setting the Browser Type and User-Agent 232
Setting Load Generator and Probing Client Thread Properties 235
Setting Redirection Limits 235
Simulating DNS Cache 236
Simulating SSL Cache 236
Enabling Persistent Connection (Keep-Alive) 236

Setting the HTTP Parameters **237**
Setting Authentication **240**
Connection Speed Settings **242**
Diagnostic Settings **243**
Enabling Syntax Checking 244
Enabling Dynamic Object Recognition Diagnostic Messages 245
Log Report Diagnostics 246
Enabling RadView Support Diagnostic 247
Enabling Save CSJS and CSJS Diagnostic Messages 247
Functional Testing Options **247**
Setting Automatic Data Collection 248
Setting Page Time Verifications 249
Inserting a Text Verification Test 250
Inserting Test Selections 251
Pass / Fail Definition Settings **252**
Configuring Reset Mechanism Settings **254**
Configuring Emulation Settings **256**
Web Services Settings **258**

17. Functional Verification Testing **259**

The Functional Testing Tab **260**
Opening the Functional Testing Tab 260
Selecting the Tests to Perform 262
Functional Verification Test List 264
User-Defined Transactions and Verification **264**
Including a User-Defined Transaction in your Agenda 265
User-Defined Transactions and Verification Syntax 265
User-Defined Transaction Example 265
Verification Function Return Values **266**
Reporting Events **267**
Tracking Reasons for Transaction Failure **267**
Viewing Verification Test Results **269**
Viewing Error Information in the Log Window 269
Viewing Error Information in the Event Viewer 270
Printing the Contents of the Log Window 271
Saving the Contents of the Log Window 271

Data Drilling 271

18. Running Tests and Analyzing Test Results 273

Explanation of Creation and Analysis of Performance Reports 273

19. Running a Load Session 275

Starting a Load Session 275
Running a Load Session from the System Memory 276
Scheduling WebLOAD Tests with TestView Scheduler 276
Launching WebLOAD through the Command Line Interface 277

Establishing Communication 278
Errors in Communication 278

The Console Screen in Session Mode 281
Viewing the Session Tree in Session Mode 282

WebLOAD Operating Commands 282
Starting a Test Session 283
Pausing a Test Session 283
Resuming a Test Session 283
Stopping a Load Session Component 284
Stopping a Test Session 284

Throttle Control - Changing the Load Size Dynamically During Runtime 284
Opening Throttle Control 285
Activating Throttle Control 286
Deactivating Throttle Control 286

20. Viewing the Log Window 287

Setting the Log Window Options 287

Opening the Log Window 288
Viewing the Entire Log Message 289
Sorting Error Messages 289
Opening the Log Window Menu 289
Printing Error Messages Recorded in the Log Window 290
Viewing the Message Status 290

21. WebLOAD Performance Reports 293

Viewing Test Results on the WebLOAD Dashboard 294

Opening Reports **295**
Opening an Existing Report **298**
Editing a Report **298**
Renaming a Report **299**
Deleting a Report **300**
Configuring an Integrated Report **300**
Selecting Measurements for the Integrated Report 302
Changing the Reports Parameters 302
Importing Report Parameters from a Saved Load Template 303
Regression Testing **305**
Configuring an Integrated Report to Include Data from Previously Saved Load Sessions 305
Removing a Previously Saved Load Sessions from the Integrated Report Configuration 306
Integrated Report Chart View **306**
Changing the Scale of a Report for a Selected Measurement 308
Viewing the True Value vs. Relative Value of a Measurement 308
Integrated Report Toolbar in Chart View 309
Integrated Report in Spread(sheet) View **310**

22. WebLOAD Statistics Reports **313**

Statistical Measurements and Their Values **313**
Changing the Length of the Reporting Interval 314
Current Value Computation 315
Per Time-Unit Statistics Definitions 315
Timer Definitions 317
Counter Definitions 320
Statistics for Virtual Clients and Probing Clients 322
Summary Reports **323**
Using the Statistics Report 324
Statistics Report-Summary by Agenda 325
Statistics Report-Summary by Total 327
Statistics Report-Summary by Measurement 329
Statistics Report-History by Measurement and Agenda 331
Statistics Report-History by Measurement and Total 332

23. Data Drilling and Table Compare Reports 335
Data Drilling Reports 335
Opening a Data Drilling Report 336
Viewing the Transaction Grid 337
Viewing the Transaction Reason Failure Grid 337
Viewing the Instance Grid 338
Viewing the Parent Transaction Instance Tree 339
Viewing the Child Transaction Instance Tree 340
Table Compare Reports 341
Opening the Table Compare Viewer 343
24. Exporting Performance Reports 345
Report Export Options 345
Exporting Selective WebLOAD Report Data 347
Exporting Performance Reports to Excel 348
Exporting Performance Reports to a Tab File 350
Exporting Performance Reports to HTML 352
Exporting All Statistical Data 353
Exporting All Statistics to Excel 354
Exporting All Statistics to a Tab File 356
Exporting All Statistics to HTML 357
Customizing the Export File 357
Creating a Summary Report 359
Customizing the Report Fields 360
Report Destination 361
Integrated Report Definition 362
25. Performance Measurements Manager 365
Using the Performance Measurements Manager 365
About the Performance Measurements Manager 366
Opening the Performance Measurements Manager 366
The Performance Measurements Manager Main Window 367
The PMM Configuration Tree Structure 370
Navigating the PMM Configuration Tree 372
Using PMM Configuration Files 373
Creating a New Configuration File 373
Opening / Importing Saved PMM Configuration Files 373

Assigning a PMM Configuration to a WebLOAD Template 374
Saving and Exporting PMM Configurations 374
Configuring the Data Sources, Hosts, and Measurements to Monitor 375
Adding Data Sources 376
Adding Measurements to a Configured Data Source 383
Adding Data Sources to a Configured Host 384
Deleting a Data Source 384
Deleting a Monitored Host 385
Deleting a Measurement 385
Application Server Monitors 385
WebLogic Monitor Configuration Tasks 385
IBM WebSphere Configuration Tasks 387
Database Monitors 387
Oracle Monitor Configuration Tasks on Windows 387
Oracle Monitor Configuration Tasks on UNIX 389
Web Server Monitors 391
Sun One (iPlanet) Configuration Tasks 391
Monitoring Generic SNMP Data Sources 392
Understanding MIB Files 392
The WebLOAD UserMibFilter.ini File Structure 393
Adding SNMP Agents 394
A. WebLOAD Error Messages 397
Message Severity 397
WebLOAD Error Messages 398
B. Reference Materials 403
WebLOAD Files 403
WebLOAD Hot Key Commands 404
Index 407

C H A P T E R 1

Introduction

This section provides a brief introduction to TestView technical support, including both documentation and online support.

In This Chapter

WebLOAD Documentation 15
Typographical Conventions 16
Where to Get More Information 17

WebLOAD Documentation

WebLOAD is supplied with the following documentation:

WebLOAD IDE™ User's Guide

Instructions for recording, editing, and debugging load test agendas to be executed by WebLOAD to test your Web-based applications.

WebLOAD™ User's Guide

A guide to using WebLOAD, RadView's load/scalability testing tool to easily and efficiently test your Web-based applications.

WebLOAD REPORTER™ User's Guide

Instructions on how to use WebLOAD REPORTER to analyze data and create custom, informative reports after running a WebLOAD test session.

WebRM™ User's Guide

Instructions for managing testing resources with the WebLOAD Resource Manager.

TestView™ Suite Programmer's Guide

Complete information on programming and editing JavaScript Agendas for use in WebLOAD and WebFT.

TestView™ Suite JavaScript Reference Manual

Complete reference information on all JavaScript objects, variables, and functions used in WebLOAD and WebFT test Agendas.

TestView™ Suite User's Guide

Instructions for working with the TestView Testing Suite, including the TestView Manager and TestView Scheduler.

The guides are distributed with the WebLOAD software in online help format. The guides are also supplied as Adobe Acrobat files. View and print these files using the Adobe Acrobat Reader. Install the Reader from the Adobe Web site (http://www.adobe.com http://www.adobe.com).

Typographical Conventions

Before you start using this guide, it is important to understand the terms and typographical conventions used in the documentation.

For more information on specialized terms used in the documentation, see the Glossary at the end of this document.

The following kinds of formatting in the text identify special information.

Formatting convention	Type of Information
Triangular Bullet(➢)	Step-by-step procedures. You can follow these instructions to complete a specific task.
Special Bold	Items you must select, such as menu options, command buttons, or items in a list.
Emphasis	Use to emphasize the importance of a point or for variable expressions such as parameters.

Formatting convention	Type of Information
CAPITALS	Names of keys on the keyboard. for example, SHIFT, CTRL, or ALT.
KEY+KEY	Key combinations for which the user must press and hold down one key and then press another, for example, CTRL+P, or ALT+F4.

Where to Get More Information

This section contains information on how to obtain technical support from RadView worldwide, should you encounter any problems.

Online Help

TestView provides a comprehensive on-line help system with step-by-step instructions for common tasks.

You can press the F1 key on any open dialog box for an explanation of the options or select Help | Contents to open the on-line help contents and index.

Technical Support

For technical support in your use of this product, contact:

- North American Headquarters

  e-mail: `support@RadView.com`

  Phone: 1-888-RadView (1-888-723-8439) (Toll Free)

  781-238-1111

  Fax: 781-238-8875

- International Headquarters

  e-mail: `support@RadView.com`

  Phone: +972-3-915-7060

  Fax: +972-3-915-7683

Note: We encourage you to use e-mail for faster and better service.

When contacting technical support please include in your message the full name of the product, as well as the version and build number.

Technical Support Web Site

The technical support pages on our Web site contain:

- FAQ (Frequently Asked / Answered Questions).
- Agenda Center
- Documentation
- RadView's Product Resource Center, where you can find prepared test scripts, product information, and industry related news.
- http://www.RadView.com/support/index.asp (http://www.radview.com/support/index.asp)

C H A P T E R 2

Introducing WebLOAD

The WebLOAD introductory material gives you an overall view of WebLOAD and consists of three chapters:

- Getting Started (on page 21) describes WebLOAD, how WebLOAD works, the WebLOAD components and features.
- WebLOAD Installation and Configuration (on page 51) provides installation and configuration information.
- WebLOAD Console Features *(on page 77)* describes the features available through the Console toolbar and menus.

C H A P T E R 3

Getting Started

Welcome to WebLOAD, the premier performance, scalability and reliability testing solution for Web applications - Internet and intranet.

WebLOAD accurately simulates Internet users behavior and models real-life demands on your Web application to predict capacity requirements, report bottlenecks and weak links in your application before deployment.

This chapter introduces WebLOAD, the WebLOAD test components and an overview of WebLOAD's capabilities and features.

In This Chapter

Welcome to WebLOAD 21
How Does WebLOAD Work? 22
WebLOAD Features 28

Welcome to WebLOAD

WebLOAD delivers the following for maximum testing performance, ease of use and value:

- Full client-side JavaScript support.
- Document Object Model (DOM) access.
- JavaScript-based test scripting.
- Seamless integration with Web application servers.
- Automated operation.
- Single-point management and real-time graphical and statistical reporting.

WebLOAD verifies the scalability and integrity of Web applications by generating a load composed of Virtual Clients that simulate real-world traffic. Probing Clients let you refine the testing process by acting as a single user that measures the performance of targeted activities, and provides individual performance statistics of the Web application under load.

- WebLOAD is powerful. WebLOAD unifies performance, scalability and integrity testing into a single process for accurate and timely verification of your Web application, under load defined conditions. WebLOAD provides you with unmatched flexibility for defining and parameterizing Virtual Clients such as connection speed, browser types, multithreading, and SSL encryption strength, for absolute control while simulating real-world conditions.
- WebLOAD is easy to use. Recording a test Agenda is as easy as using a browser, with WebFT recording each step. WebFT requires no programming and is operated using an intuitive, graphical user interface to create visual Agendas. The visual Agendas you develop in WebFT can be run in WebLOAD to emulate Virtual Clients for load testing, or can be run in WebFT with verification testing capabilities.
- WebLOAD also automatically handles dynamic Session IDs with its Dynamic Object Recognition (DOR), and provides native access to the Document Object Model making advanced functionality only a mouse-click away.
- WebLOAD is efficient. WebLOAD delivers unmatched levels of load generation through its optimized architecture and use of operating systems. You can freely incorporate additional systems for virtually unlimited testing loads.
- WebLOAD is simple. WebLOAD automatically finds the performance threshold of your Web application through Cruise Control. Simply define target performance parameters and WebLOAD generates and increases the number of Virtual Clients accessing the application until performance levels can no longer be sustained.

How Does WebLOAD Work?

WebLOAD tests Web applications by generating Virtual Clients that simulate real-world loads. Virtual Clients emulate the actions of human users by performing typical actions on your Web application. By increasing the number of Virtual Clients, you increase the load on the system. You create visual JavaScript-based test scripts (Agendas) that define the behavior of the Virtual Clients and WebLOAD executes these test scripts monitoring the application response graphically and statistically, and presenting the test results in real time.

WebLOAD incorporates functional verification into the scalability testing process allowing you to accurately verify the scalability and integrity of your Web applications at the per-client, per-

transaction, and per-instance level under defined load conditions. WebLOAD saves the test results including data from the Load Machines (described in the paragraphs that follow) and the hosting hardware's performance monitor. You can view all or part of the data in real time, or after the test session is complete, in tabular format or in graphical format.

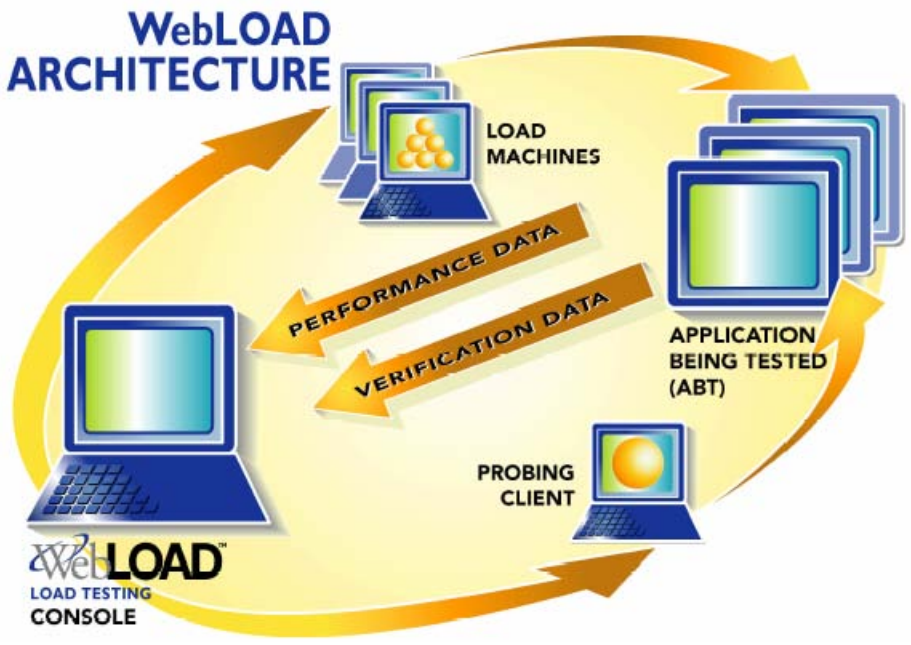
The WebLOAD test process includes:

1. Planning the test
2. Creating Agendas
3. Creating Load Templates
4. Running the test
5. Analyzing the test results

This guide describes steps 1, 3, 4 and 5 in the test creation process in detail. Information on performing step 2, Creating Agendas is available in the *WebFT User's Guide*.

WebLOAD Architecture

The following diagram illustrates the configuration for a typical WebLOAD test.



The Console sets up, runs and controls a test session. At the Console, you can:

- Define the hosts participating in the load test
- Specify the test scripts (Agendas) that the load test executes
- Schedule tests
- View performance reports

Load Machines are hosts, which run Load Generator software simulating multiple Virtual Clients simultaneously accessing your Application Being Tested (ABT). One Load Machine can run multiple Load Generators.

Load Generators execute tests that "bombard" the Application Being Tested with a large load, to enable complete scalability and stress testing.

The tests consist of multiple simultaneous requests made by Virtual Clients (which emulate Web browsers) to Web servers. Load Machines can run multiple threads.

The Application Being Tested (ABT) is where the Web application being tested resides. The ABT does not require that WebLOAD software be installed on it.

The Probing Client Machines are also hosts. They run Probing Client software, which simulates a single Virtual Client accessing the ABT and runs at the same time as Load Machines, to further measure the performance of the ABT under load. WebLOAD returns exact values for Probing Client performance, in contrast to Load Machines, which return average values of Virtual Client performance.

Additional WebLOAD components are:

- WebFT - a tool for recording your activity in a Web browser that automatically translates that activity into a visual Agenda that can be used in WebLOAD for load testing. WebFT is installed on your system with the Console installation. For more information, see the *WebFT User's Guide*.
- TestTalk - a network agent that facilitates communication between the Console and hosts, either Load Machines or Probing Client Machines. TestTalk *must* be installed on both the Console and the hosts. TestTalk runs in the background and does not interfere with any other program running on your system.
- WebLOAD REPORTER - enables you to generate a variety of reports based on performance data and statistics gathered by WebLOAD. Use WebLOAD REPORTER to select a load session and generate predefined or custom reports displaying the results of the test.
- WebRM- distributes and circulates WebLOAD testing resources (Virtual Clients and Probing Clients) amongst users on a "need to use" basis. WebRM is packaged with a maximum number of Virtual Clients, Probing Clients and Connected Workstation ports, as defined by the WebLOAD package. With WebRM, every WebLOAD Console can operate in Standalone Workstation mode or Connected Workstation mode.
- Standalone Workstation - an independent Console that runs WebLOAD and is limited to 20 Virtual Clients with one Load Machine running on the local machine. It is not connected to WebRM.
- Connected Workstation - a Console that runs WebLOAD, is connected to WebRM, and can be connected to other Load Machines running on different machines.

  The Connected Workstation sets up, runs, and controls the test. Using a Connected Workstation, you can request resources (Virtual Clients and Probing Clients) from WebRM, schedule tests and free the resources being used.

How are Tests Defined?

WebLOAD tests are configured by creating load templates, which define the events to occur in the test session. Using the WebLOAD Wizard, you can quickly and easily define basic load templates that include:

- The Agendas (test scripts) to run
- The machines on which to run the test
- The number of Virtual Clients to run
- The test schedule

After creating a basic load template with the WebLOAD Wizard you can modify the template to include advanced features, such as connection speed, browser type and pass/fail definitions, through the WebLOAD Console menus.

What is the Product of Your WebLOAD Testing?

WebLOAD continuously samples the activity while a load session is in progress, and saves a wide variety of measurement data. You can view all or part of the data in real time or after the load session is complete. Data can be displayed in tabular format or in graphical format. At the conclusion of a load session, data can automatically be exported to a spreadsheet program, HTML or as a tab-delimited file to a database application. Reports can also be created using WebLOAD REPORTER.

WebLOAD Offers Two Client Type Modes

Test sessions are not all the same. Sometimes you want to test and scrutinize every single client activity, and verify every single transaction, no matter how trivial it seems or how frequently it occurs. Other times you want to focus on a specific category of problematic activities and can't waste time looking at anything else. You may even decide to generate a huge number of client transactions in as short a time as possible to see how your application copes when pushed to the limit and beyond. And sometimes, you don’t want to take any extra time or resources to measure anything specific, you just want to see how your Web application behaves for a typical user under standard operating conditions.

To provide maximum flexibility and responsiveness to different testing needs, WebLOAD offers two different testing modes. Testers can utilize these different testing modes to design test session tailored to their specific needs, choosing the combination of testing modes and Agendas most suited to their individual requirements. These two testing modes are referred to as two Client Type Modes:

- HTTP Client (on page 26)—Simulates the user scenario at the protocol level. Each Virtual Client sends HTTP methods (and/or other protocol calls) to the server based on the recorded agenda. (Recommended)
- Browser Client (on page 27)—Simulates the user scenario at the browser/application level. Each Virtual Client emulates browser behavior and executes client-side logic, resulting in sending HTTP methods to the server.

This section provides an overview of the features of each of these modes, including a table comparing these features in Client Type Feature Summary (on page 28), to help you choose the combination of modes most applicable to your testing needs. For example, a typical testing scenario may have a comprehensive test session that includes a large number of simple Agendas run under Browser Client mode. A definitive benchmark can then be set using an extremely large load generated by Agendas run under HTTP Client mode.

HTTP Client

The focus on user activities used by the Browser Client mode represents a high-level, conceptual approach to test session design. This user-activity focus is not always appropriate for all testing

situations. Sometimes a tester may prefer to use a low-level, "nuts-and-bolts" approach that focuses on specific internal implementation commands, such as HTTP transactions.

Agendas created for HTTP Client mode focus on HTTP transactions. HTTP Client Agendas are created through the WebLOAD IDE. The WebLOAD IDE is an easy-to-use tool that facilitates the writing of test Agendas for HTTP Client mode. The WebLOAD IDE records your actions as you navigate through a Web application, automatically generating an Agenda that reflects your actions in JavaScript, writing the GET and POST HTTP protocol commands that correspond to your browser activities. You can then use these Agendas to test the performance of your Web application during WebLOAD test sessions.

Working in HTTP Client mode provides users with the most efficient, portable, powerful work mode. HTTP Client mode is portable to many operating systems, including all different flavors of Windows, UNIX, Solaris, and so on. HTTP Client mode runs so efficiently that users can achieve large benchmarks by simulating extremely heavy test loads of thousands of clients. HTTP Client mode also provides the most comprehensive, detailed statistical analysis and test reports.

Working in HTTP Client mode does require a certain level of programming sophistication, since it can require hands-on programming of the JavaScript Agendas with HTTP protocol code, as described in the *TestView JavaScript Reference Manual*. Since the Agendas for HTTP Client mode are created through the WebLOAD IDE, they are not compatible or interchangeable with Agendas created through WebFT, and can not be run on WebFT. HTTP Client mode Agendas are designed to be run only on WebLOAD. HTTP Client mode is also not recommended for web sites with complex User Interfaces or a heavy use of JavaScript. Test sessions that access such web sites should be run in one of the other Client Type modes. WebLOAD provides these different test modes to accommodate a wide range of testing requirements.

Browser Client

During test sessions run in Browser Client mode, Agendas that recreate user browser activities are run through an emulation engine that simulates actual browser activities. WebLOAD offers the option of running test session Agendas through an emulation engine to provide the full set of complex functional verification tests and statistical analyses available through HTTP Client mode.

Browser Client mode is the mid-range work mode for WebLOAD testing and provides a powerful and sophisticated testing option. For example, Agendas running under Browser Client mode can be configured to compensate for slow internet connections, or to filter out certain URL sites, providing more control over the test session environment. Browser Client mode also provides a simpler user interface than HTTP Client mode. Working in Browser Client mode provides users with a work mode that is powerful enough for most work situations, while easy to work with, with a user-friendly GUI that simplifies user input and minimizes any programming requirements. Browser Client mode is portable to most operating systems, including both Windows and UNIX systems. Browser Client mode also handles medium to high test loads of 100-300 clients, and provides clear, detailed statistical analysis and reports for a sophisticated set of HTTP manipulations.

Browser Client mode Agendas can be customized to meet most common testing needs. Browser Client mode is usually not recommended for web sites with complex User Interfaces or a heavy use of JavaScript. Test sessions that access such web sites should be run in one of the other Client Type modes. WebLOAD provides these different test modes to accommodate a wide range of testing requirements.

Client Type Feature Summary

The following table summarizes the features of each Client Type mode:

Feature	Browser Client	HTTP Client
Average Load per Machine*	Up to 300	Up to 1000
*single CPU, 1GB memory		
VB Scripts		
Complex Objects (Flash, ActiveX, etc.)		YES, if using HTTP
Single Script for WebFT & WebLOAD	Yes	
Authoring Tool	YES	YES
Portable to UNIX, etc.	YES	YES
Detailed Statistics	YES	YES
HTTP Manipulations	YES	YES
Backward Compatibility (V5.0)		YES

WebLOAD Features

Using WebLOAD, you have access to all the features described in this section.

Feature	Benefit(s)
HTTP 1.0 and 1.1	Comprehensive support for HTTP 1.0 and 1.1 including cookies, proxies, SSL, client certificates, authentications, persistent connections and chunked transfer coding.
SSL updates	Support for the new SSL standard (TLS 1.0)
SSL Bit Limit	Enables you to set a limit on the maximum number of bits available to Virtual Client for SSL encryption.
WAP support	WAP can now be selected as the browser emulation browser type in the Console user interface. WAP sessions can be recorded into scripts and WAP headers can be edited using WebLOAD IDE for configuration and parameterization. Various hand-sets can be emulated by defining custom user-agents. Binary WAP (WAP1) can also be recorded and re-played during a load session.
Internet Protocol Support	Native support for FTP, SMTP, POP3, IMAP, NNTP, TCP, UDP, and Telnet provides support for non-HTTP protocols.
COM support	WebLOAD Java Script provides direct object access to any component that has a COM wrapping and an Idispatch interface.
Java from JavaScript	WebLOAD supports full Java access from your JavaScript Agendas. Full Java support means that your WebLOAD Agendas not only test access time to an HTML page, they can also invoke and run Java classes used by the Java applications embedded within an HTML page.
ODBC Support	Native support from WebLOAD Java Script for ODBC commands to be integrated in an Agenda.
Throttle Control	Vary the number of Virtual Clients on the fly to create “what-if” scenarios as you change the load conditions.
Data Driven Load Testing	Use real data to emulate realistic scenarios where many different users are accessing your Web application simultaneously. For example, you can use different usernames, and passwords for different Virtual Clients.
Real-Time Server Side Statistics	Using WebLOAD you can monitor a variety of server-side application, database, system and Web-server statistics. You can configure the WebLOAD performance monitor to display real-time performance statistics for services available from the server including percentage of CPU usage, memory usage, and so on.

Feature	Benefit(s)
Supported Server Side Statistics for Open Interoperability	Server side statistics can be collected over generic protocols and interfaces such as SNMP, JMX, Perfmon (Windows) and a Generic Unix performance collector. Specific pre-configured collectors exist for the following servers, using the generic protocols: BEA WebLogic, IBM WebSphere 5, Microsoft Active Server Pages, Microsoft ASP .NET, Oracle, Microsoft SQL Server, Windows NT/2000/XP, UNIX platforms, Windows .NET Framework, Microsoft IIS, Sun One (former iplanet 6), and Microsoft Media Server.
Agenda Authoring (as part of WebLOAD IDE)	Automatically generates test scripts in JavaScript from a standard Web browser.
JavaScript Agenda Definition (as part of WebLOAD IDE)	Delivers the familiarity and scripting power of a full programming language for maximum flexibility, including writing, editing and extending.
Comprehensive Support for Dynamic Content	WebLOAD delivers comprehensive support for dynamic applications.
Capacity planning using Cruise Control	Predictive feature for testing the performance thresholds of Web applications.
Unlimited Load Generation	Optimized, distributed architecture for unlimited load generation with minimal resources.
Support for Keep-Alive	Supports persistent connections with Web applications for accurate simulation of Web browser behaviors.
Supports Multithreading	Provides accurate real-world simulation of Web browser behavior.
Browser Emulation	Emulates Microsoft and Netscape browsers for maximum real-world simulation.
Single-Point Management	Console for managing WebLOAD setup, testing, monitoring and reporting.
Real-time Dashboard	View test execution statistics in real-time for a quick and automated overview of test performance.
Real-Time Analysis	Graphical and statistical analysis of test sessions while the test session is running.
Real-Time Reports and Graphs	WebLOAD's integrated reports let you see the results of your test in real time. Test results can be viewed on the test, Agenda, page and object level. Reports include over 75 metrics including response time, hit per seconds, pages per second, and so on.

Feature	Benefit(s)
Comprehensive Performance Measurements	Over 25 performance metrics for maximum testing flexibility and accuracy.
User-defined Timers	Measure specific activities for identifying performance bottlenecks.
Comprehensive Test Session Reports	Save and compare graphical and statistical test reports for measuring the performance of an application over time.
Microsoft Excel Integration	Automatically exports all test session data to Microsoft Excel for easy storage and reference.
HTML reports	Export WebLOAD's reports to HTML format.
Predefined Default Reports	WebLOAD supplies six predefined reports to provide you with a fast and easy way to view the information of interest to you in real time without having to spend time configuring the measurements to appear in the reports.
WebLOAD REPORTER	WebLOAD REPORTER provides comprehensive reports including failure rates. It automates the process of manually creating reports from WebLOAD performance data. Easily and quickly view these pre-configured reports to gain better insight to performance and scalability issues of your Web applications.
Save Test Scenarios	Saves test scenarios for accurate before/after comparisons and for re-executing test sessions.
Enterprise WebLOAD resource pooling and sharing (using WebRM)	Addresses corporate functional and performance testing efforts by organizing, managing and directing WebLOAD resources (Virtual Clients, Probing Clients and Connected Workstations). WebRM enables multiple users involved in various stages of application development and testing to share testing resources. By distributing WebLOAD testing resources each developer can run a test session to simulate, validate, and pinpoint where performance problems occur at any stage of the development life cycle, thus eliminating design flaws and ensuring product quality.
Variable Connection Speed	Enables you to define a load as coming from different connection speeds (that is, 14400, 28800, and so on).
Functional Testing	WebLOAD incorporates functional testing into the scalability testing process, shortening testing cycles and enabling integrity verification under user-defined load conditions at the per-client, per-transaction, and per-instance level.

Feature	Benefit(s)
Data Drilling	When testing the integrity of your Web application, it is important to verify whether a given transaction succeeded or failed, as well as to determine the reasons for the failure. Data Drilling enables you to display a detailed description of all user-defined and named transactions to the instance level.
Table Support	When viewing Web pages, data is often presented in a tabular format. WebLOAD supports tables in the DOM. It also provides Table Comparison JavaScript utilities and Table Comparison error reporting.
Synchronization Point	Enables you to ensure that all Virtual Clients execute a given command simultaneously.
IncludeFile	Provides modularity by allowing you to reference JavaScript source code in multiple Agendas using the `IncludeFile` command.
CopyFile	Enables you to copy files from the Console to a Load Machine. The `CopyFile` command is important if your test uses an Agenda that references auxiliary files such as input files, output files, or other files, and the Agenda and its auxiliary files reside on the Console.
wlOutputFile	Enables you to save Agenda output for later study and analysis. The `wlOutputFile` object lets you write Agenda output messages to an output file. See the *WebLOAD JavaScript Programming* and *RadView JavaScript Reference* manuals for more information.
File Locations	Supplies a default directory for files used and generated by WebLOAD, including: ◆ Sessions ◆ Templates ◆ Agendas ◆ IncludeFiles ◆ CopyFiles ◆ Scripts
Launch Command	Allows you to initiate testing through a command line interface. You can enter the WebLOAD launch command into either a batch job or an external script. WebLOAD runs directly, without user intervention, using the parameters specified.

Feature	Benefit(s)
Browser Cache	Imitates the Web browser cache activity to provide a realistic simulation of Web activity.
Easy User Interface Settings for Runtime Options for each Agenda	Enables you to customize the runtime options through the Console User Interface.
XML Document Object Model support	WebLOAD provides full support for work with the XML Document Object Model. Using XML DOM objects, WebLOAD Agendas are able to both access XML site information, and generate new XML data to send back to the server for processing, taking advantage of all the additional benefits that XML provides.
Regression Testing	Regression testing enables you to compare Load Sessions while tests are running, or after the Load Session is complete.
Dynamic Object Recognition (DOR)	Dynamic Object Recognition (DOR) enables users to record Agendas against dynamic sites (using dynamic session IDs) with WebFT and be able to run the Agendas immediately without having to modify the Agenda script to handle the dynamic session ID. Dynamic Object Recognition (DOR) identifies the dynamic session ID during recording and automatically inserts code into the Agenda to identify and replace the session ID during playback.
Automatic Transactions	All transactions in the Agenda are automatically turned into Named Transactions, eliminating the need to manually create timers for all transactions.
Automatic named transaction enhancements - Object level and HTTP level	WebLOAD automatically creates sub transactions for all content and automatically breaks down every HTTP command into its elements (open, send, wait, receive).
Diagnostics options	The Diagnostic options can be enabled when developing an Agenda or for tracking problems in existing Agendas.
Global Functional Verification	Global Functional Verification enables you to select verification tests to run on all pages returned from the server from a list of predefined tests.
Global User Variables	Global User Variables enable you to share data between Load Generators and Load Machines.
Performance Threshold	Transactions can be defined as failed by the amount of time they take.

Feature	Benefit(s)
Error path zoom in/out and coloring	The path of the failed transaction is colored and expanded in the Data Drilling reports to emphasize the failed transaction. A zoom in/out feature offers the ability to expand or collapse the Data Drilling tree according to the error path.
WebLOAD Viewer	Graphically see the difference between expected and actual data.
First Byte Statistic	First Byte is the time it takes a Virtual Client to receive the first byte of data.
Agenda Level Options	Unique options can now be set for each Agenda in the test session.
Load Profiler	Create schedules based on pre-defined visual templates for easier and faster scheduling.
Sleep Recording	Enables you to automatically record “think” time in the Agenda to provide a more realistic simulation of users accessing your Web application.
Sleep Time Control Settings	Easily control sleep behavior on the fly to save script development time.
Enhanced Statistic Setting	ESS provides additional statistical information on the WebLOAD metrics.
Pass/Fail Definitions	Set rules to define the success or failure of test sessions.
TestView Scheduler	Automates the process of scheduling WebLOAD tests and test plans.

C H A P T E R 4

WebLOAD Quick Start

This section shows you how you can get started quickly using the RadView Software test site at www.netizenbanking.com (http://www.netizenbanking.com). Then you can refer to the complete WebLOAD documentation set to learn more techniques for using WebLOAD.

In This Chapter

Quick Start Steps 35
Creating an Agenda 36
Configuring a Load Template using the WebLOAD Wizard 40
Running the Test 44
Analyzing Test Results 45

Quick Start Steps

This quick start shows you how you can get started quickly using the RadView Software test site at www.netizenbanking.com (http://www.netizenbanking.com). Then you can refer to the complete WebLOAD documentation set to learn more techniques for using WebLOAD.

In this Quick Start, you perform the following steps:

1. Create an Agenda (see Creating an Agenda (on page 36)).
2. Configure a load template using the WebLOAD Wizard (see Configuring a Load Template using the WebLOAD Wizard (on page 40)).
3. Configure session options (see Setting Session Options (on page 43)).
4. Run a test (see Running the Test (on page 44)).
5. Analyze test results (see Analyzing Test Results (on page 45)).

Note: We recommend that you follow the steps in order. All examples are interrelated and dependent on earlier steps.

Creating an Agenda

The first step in creating an Agenda is to record your actions as you interact with your Web application.

1. Start WebLOAD IDE by selecting Start | All Programs | TestView | WebLOAD IDE.

   WebLOAD IDE opens.

2. Select Create a new project, and click OK.

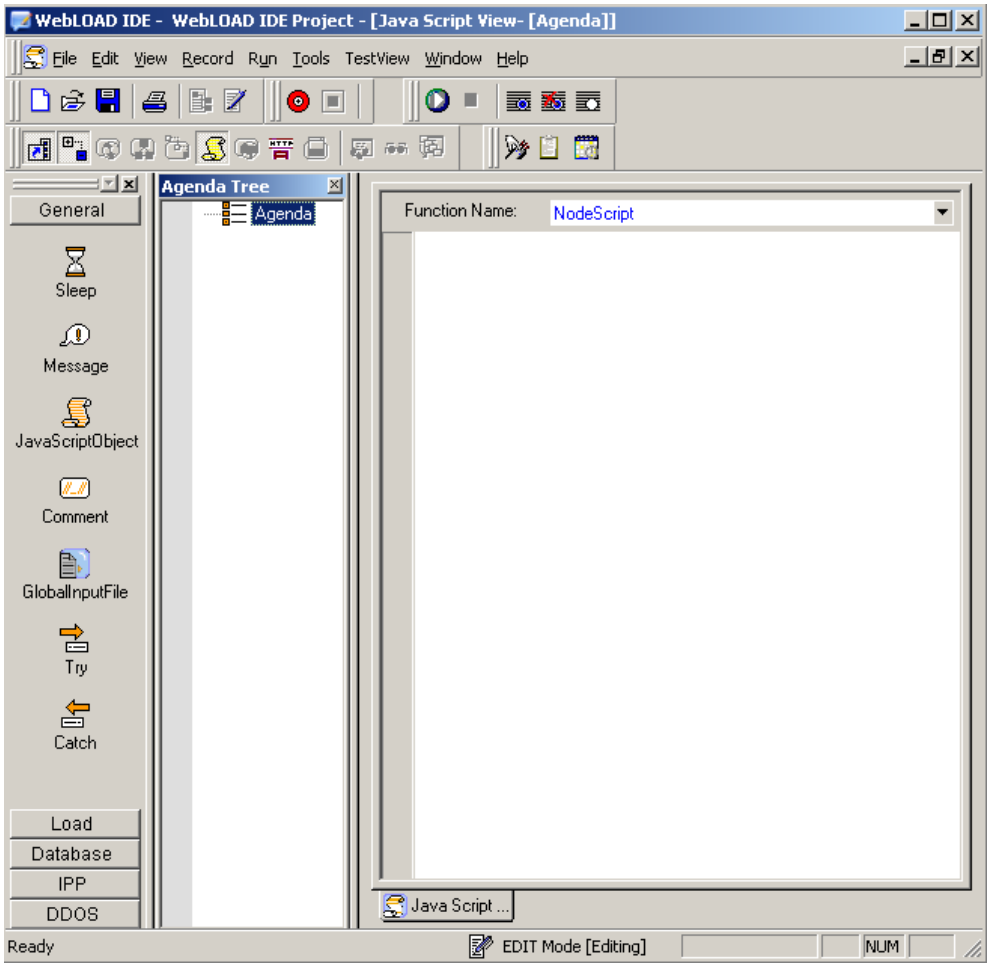
   The WebLOAD IDE main window opens in Editing Mode, for you to begin creating your Agenda.

   When the WebLOAD IDE main window first opens, it opens in Visual Editing Mode. In this mode, there are two active panes: the Agenda Tree and the JavaScript view pane.

   In Visual Editing mode, you can simply record the actions in a browser without programming. Your interactions with your Web application are captured, recorded, and presented graphically in the Agenda Tree.

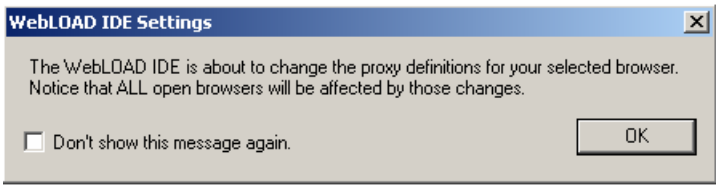
Each node in the Agenda Tree is actually a visual representation of JavaScript code. You can view the contents of the nodes in the JavaScript view pane.

To the left of the Agenda Tree are WebLOAD IDE toolboxes that can be used to edit an Agenda by dragging and dropping items from the WebLOAD IDE toolboxes into the Agenda Tree. This makes programming easier by building the code behind an intuitive drag-and-drop interface.



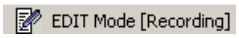
3. In the main window, in Editing Mode, click the Start Record toolbar button to begin recording.

   The following message appears.



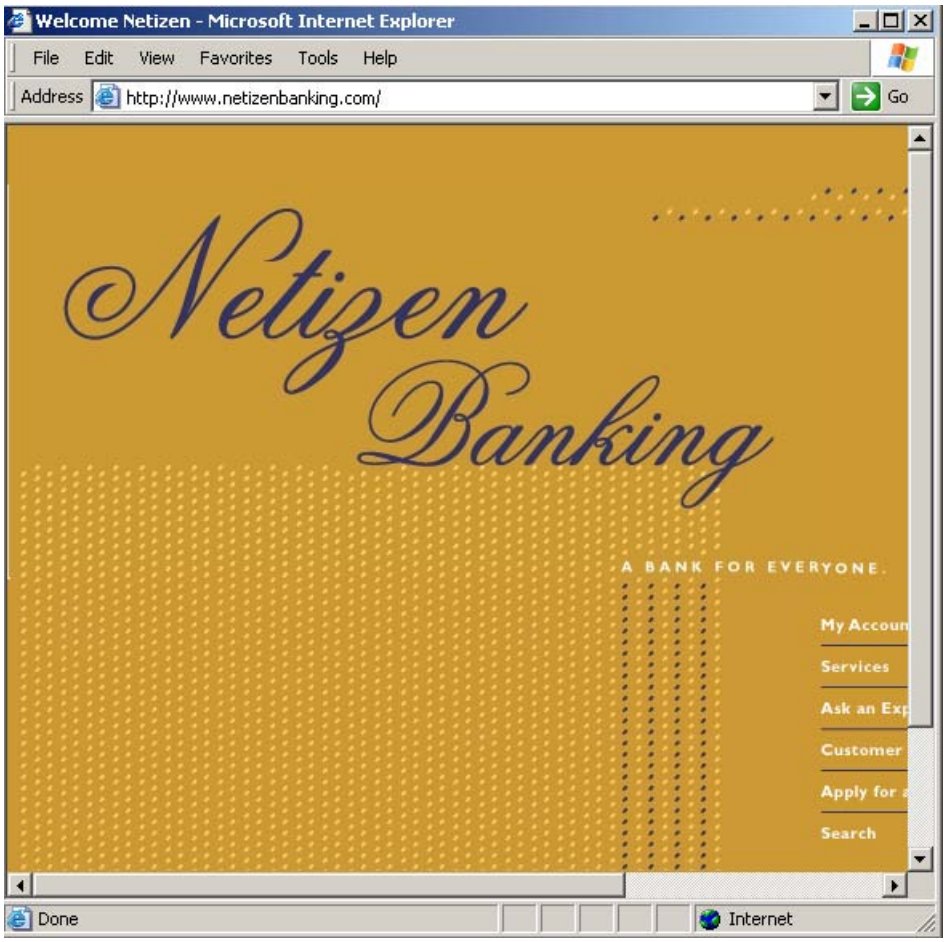
4. Click OK.

   WebLOAD IDE begins recording all of the actions you perform in the browser, as indicated by the recording notification in the WebLOAD IDE status bar.



   A blank browser window opens.

5. In the address bar, enter the Web address www.netizenbanking.com to go to the RadView Software test site.



6. Navigate through the site, performing the actions you want to test.

   For example:

   a. Click My Account on the home page.

   b. Click the Click Here to Login link.

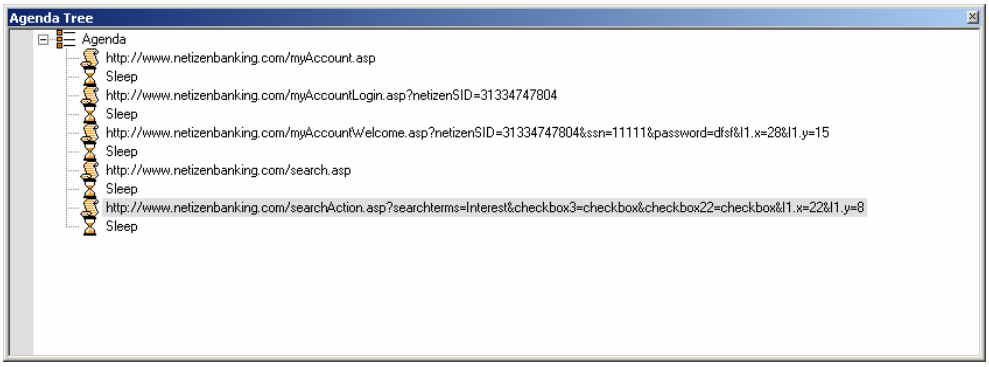
   c. Enter any set of numbers as a fictional social security number, enter a dummy password, and click Submit.

   d. Click the Search link to the right.

   e. Enter a keyword such as Interest to search for in the search field and select the checkboxes Home Mortgage and Business Banking.

   f. Click Search.

Your actions are recorded and appear in the Agenda Tree as you navigate the site. (If you see more nodes in the Agenda Tree with different URLs, this may be traffic generated by browser plug-ins or extensions, for example, third-party toolbars).



7. Click the Stop Record toolbar button in WebLOAD IDE to stop the recording.
8. Click the Save toolbar button or click File | Save As to save your Agenda.
9. Type in Netizen Banking for the name of the Agenda in the Save As dialog box and click Save.

   The Agenda is saved with the extension *.wlp.

You now have a basic Agenda that can be used in a WebLOAD template. For complete information on creating, editing, modifying Agendas, and adding functionality to your Agenda, see the relevant sections of the *WebLOAD IDE User's Guide*.

Configuring a Load Template using the WebLOAD Wizard

From this point on, this Quick Start walks you through the following steps using the WebLOAD Console in configuring a load template with the WebLOAD Wizard:

1. Opening the WebLOAD Wizard.
2. Selecting an Agenda and a Performance Statistic to monitor.
3. Selecting a host on which to run the Load Machine.
4. Scheduling the test, and selecting the number of Virtual Clients to run.
5. Setting session options.

For more details about each of these steps, see the *WebLOAD User's Guide*.

Opening the WebLOAD Wizard

The first step in defining a load template with the WebLOAD Wizard is defining the type of Agenda to run. The Agenda is the test script, which defines the actions that the Virtual Clients

perform during the test. WebLOAD supports both single Agendas (lone test scripts) and Mix Agendas (a group of Agendas simulating groups of users performing different activities on the application at the same time). In this Quick Start, we will create a template running a single Agenda. For information on creating templates with multiple Agendas or Mix Agendas, see Creating Load Templates with the WebLOAD Wizard (on page 133) in the *WebLOAD User's Guide*.

1. Start WebLOAD.

   Select Start | All Programs | TestView | WebLOAD Console.

   The WebLOAD Console opens, and displays the WebLOAD Console dialog box.

2. Select Create a new template using the WebLOAD Wizard, and click OK.

   The WebLOAD Wizard opens.

3. On the Welcome screen, click Next.

Selecting an Agenda and a Performance Statistic to Monitor

1. On the Agenda/Mix Type screen, select Single Agenda, and click Next.
2. On the Agenda/Mix Selection screen, click the [...] button and browse your system to the location of the Netizen Banking.wlp Agenda created earlier using WebLOAD IDE. Select the Agenda and click Open.
3. Click Measurements Manager at the bottom of the screen to open the Performance Measurements Manager and define the statistics to monitor during the test.

   Note: In this simple example, we will monitor the CPU usage on your system during runtime. During a real test, you would probably monitor the CPU performance of your application server.

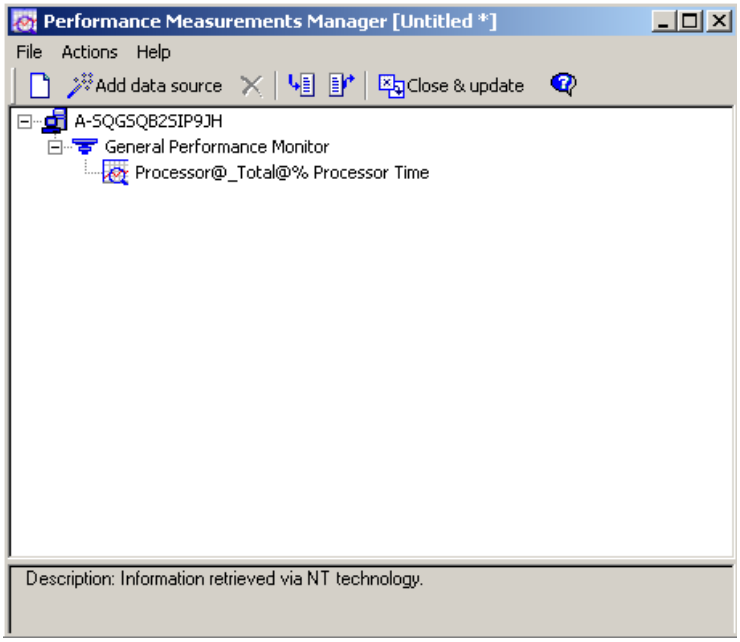
4. Click the Add data source toolbar button.

   The Performance Measurements Manager Wizard opens.

5. On the Welcome screen, click Next.
6. Click the "+" adjacent to Other to expand the tree, select General Performance Monitor, and click Next.
7. Click Browse... to browse available hosts. In the Host selection dialog box, highlight the name of your computer in the list, and click Select.
8. Click Next.

   WebLOAD attempts to connect to the selected host and collect data. A tree of all available measurements to monitor is displayed.

9. Expand the tree to select the measurement you want. Click Processor | _Total, select the checkbox adjacent to % Processor Time, and click Next.
10. Click Finish.



11. Click the Close & update toolbar button to save the configuration with the template and return to the WebLOAD Wizard.
12. On the Agenda / Mix Selection screen, click Next.

Selecting a Host

For the steps in this Quick Start, you use your own machine as the host load machine to generate Virtual Clients. The number of Virtual Clients that can be generated depends on the power of the host machine. For this test, you do not select a Probing Client.

1. On the Host Selection screen, note that your computer is automatically listed as the default Load Machine.
2. Click Next to accept the default selection.

Scheduling the Test

When you schedule the test, you define the load to be generated throughout the test and the total testing time. The Load Profiler provides a quick and easy way to select different testing patterns.

To schedule the test:

1. On the Schedule screen, select Manual, and click Next.

   The wizard progresses to the Schedule Manually screen.

2. Click on the name of your computer.

   The Load Profiler buttons becomes enabled.

3. Click Load Profiler.

   The Load Profiler opens.

4. Select Linear from the Select scheduling model drop-down list.

5. Accept the defaults for the Total time in minutes and Starting Load Size fields. Enter 10 in the Concluding Load Size field.

   The test will run for 10 minutes, starting with a load of 5 Virtual Clients and increasing to a maximum of 10.

6. Click Replace.

   The Linear Schedule model now appears in the row below your host computer name in the Schedule Manually screen.

7. Click Next.

8. Clear the option to run the load template immediately, and click Finish.

You have now completed creating a load template using the WebLOAD Wizard.

Setting Session Options

Before you run the template you just created, you can select options for verifications to perform during runtime. You can set WebLOAD to:

- Collect data at the page level.
- Verify the time required to receive a page (default 20 seconds).
- Verify that the links in your Web application act as expected.

When you select page verification, WebLOAD collects transaction data under the Title name (in this case www.netizenbanking.com (http://www.netizenbanking.com)). You can observe statistics, such as the response times for this page, as the load increases. You can also gather performance data about the objects on a page by setting WebLOAD to collect object level data. As a result, you can determine whether pictures or other embedded objects are taking a long time to download.

To set options for these verification tests:

1. In the WebLOAD Console, select Tools | Current Session Options.
2. Select the Functional Testing tab.
3. Under Automatic Data Collection, select the Pages checkbox.
4. Under Page Time, select the Verification checkbox, enter 20 in the Max Time field, and select Minor Error from the Severity Level drop-down list.

   If the time required to receive a page exceeds 20 seconds, a Minor Error message appears in the Log Window and the event is available in the WebLOAD data drilling report.
5. Under Verification, select the Verification checkbox, and enter 5 in the % field.
6. Select the Advanced Emulation Verifications checkbox, and click the [...] button to open the Emulation Verifications dialog box.
7. Select the Compare Links checkbox, and click OK.

   Five percent of the Virtual Clients will perform the selected verification tasks. This percentage lets you balance data gathering with the need to maintain performance during the test.
8. Click OK.

Running the Test

You are now ready to start the test. As you run the test, you can view the data gathered by WebLOAD in real time.

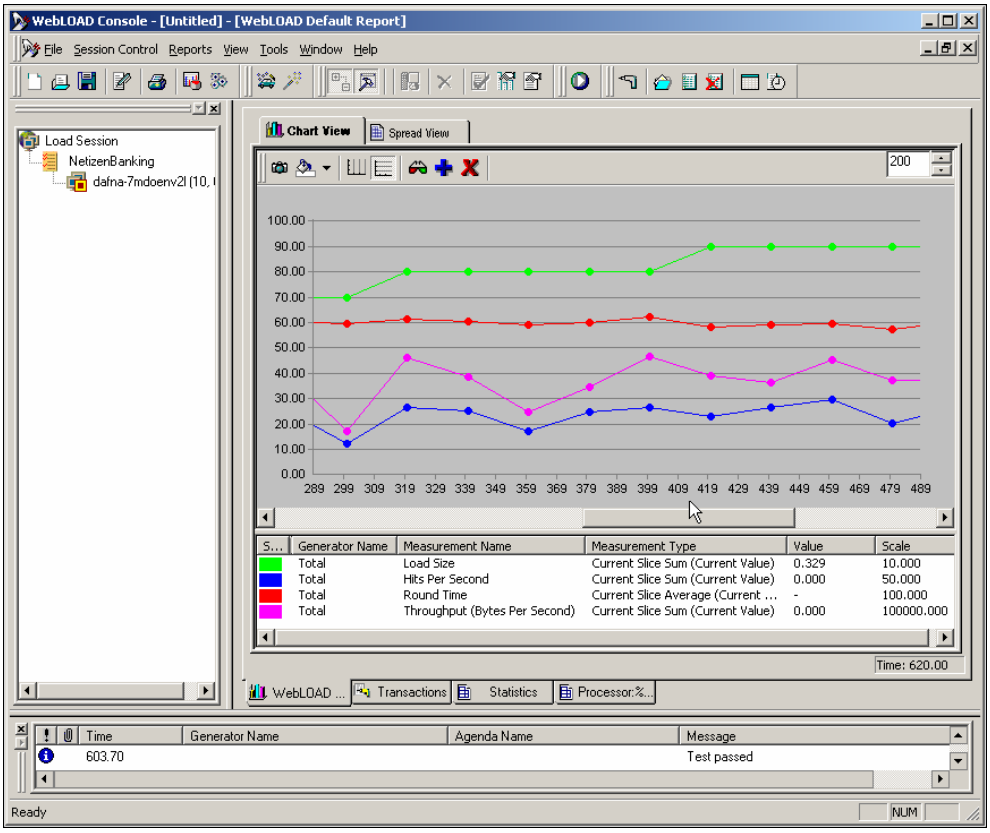
1. Click the Start Session icon on the Console toolbar.
2. At the prompt, click Yes to save the template. Type the filename NetizenBanking.tpl, and click Save.

   You can now reuse this template, with its Agenda, schedule and options at a later time.

   The test begins.

Analyzing Test Results

You can view test results in real time while the test is running or after the test session has completed. Once the test begins, the WebLOAD Wizard default report displays in the Results window.



Viewing the Results in Real Time

- Data on load size, hits per second, round time (one run through the Agenda) and throughput (bytes per second) appear in the Chart View window, with a color legend underneath the chart. Each data point represents a 20 second slice. Data is gathered every 20 seconds and then either averaged (for hits, round time and throughout) or summed (for load size). The graphs are all normalized to scale so they can be viewed in a single Chart View window.
- Right-click on any data point to see actual values.
- Select the Spread View tab at the top of the Results window to see the data in numerical format.
- Select the Dashboard icon on the Console toolbar to view a quick overview of the status of the test including the following:
  - Elapsed time
  - Current load size (number of running Virtual Clients)
  - Rounds executed
  - Failed rounds
  - Failed hits
  - Hits per second
  - Pages per second
  - Throughput
- Select the Open Statistics icon on the Console toolbar. A Statistics tab appears in the results window. Scroll through the data and double-click a cell to drill-down to further detail.

Each of these views opens in a new tab window. Please close all windows except the Default Report tab.

Creating Integrated Reports

Create an Integrated report to view the results for additional metrics gathered by WebLOAD. This can be done while the current session is still running.

To create an Integrated Report:

1. In the WebLOAD Console, select Reports | Report Manager.

   The Integrated Report dialog box appears.

2. In the Integrated Report dialog box, click New.
3. In the tree, expand the Netizen Banking::Total branch.
4. Click the "+" sign adjacent to Hit Time to expand the branch and then expand Current Slice. Select the checkbox adjacent to Average (Current Value).

   Checkmarks appear next to Hit Time, Current Slice, and Average (Current Value).
5. Expand Successful Hits and then expand Current Slice. Select the checkbox next to Sum (Current Value).
6. Expand Successful Connections and then expand Current Slice. Select the checkbox next to Sum (Current Value).
7. Scroll to the PM@your-system-name branch at the bottom of the tree (PM stands for performance monitor). Click the "+" sign adjacent to it to expand the branch. Expand Processor:%Processor Time:_Time and then expand Current Slice. Select the checkbox next to Average (Current Value).

   This choice displays the statistics on CPU processor usage that were collected during the load test.
8. Click OK.

   The Save dialog box appears.
9. Type in MyReport for the name of the report, click Predefined Report, and click OK.

   The new report is added to the list of Predefined Reports.
10. To open the newly created report, select MyReport in the list of Predefined Reports, and click Open.

    In the WebLOAD Console Results window, the MyReport tab appears displaying the report data. The name of the tab is the name of the report you selected.
11. If the test is still running, select Session Control | Stop All.
12. Select File | Save Load Session to save the data from the test along with the template information (Agenda, options and schedule). Specify a filename (NetizenBanking), and location and click Save.

Viewing Data Drilling Reports

The Data Drilling report shows all transactions for each web page tested, categorized by the following:

- Total
- Successful
- Failed
- Marked counts (indicates how many children of the transaction failed.)

Data Drilling provides both a global and detailed account of hit successes and failures, letting you verify the functional integrity of your Web application at the per-client, per-transaction, and per-instance level.

To view Data Drilling reports:

1. Click on the Data Drilling icon on the Console toolbar.

   A Transactions tab appears in the Results window.

2. On the Transaction tab, transaction failures are noted in the Failed Count column. For any transaction that has an entry in this column, click the page name in the Transaction Name column to see the reasons for the transaction failure.

   A reason failure row will appear in the bottom half of the screen.

3. To discover which transactions failed, double-click the reason failure row to open the Instance Grid. This grid lists the individual instances of transaction failure.

Note: You can continue exploring the reasons for the transaction failure by double-clicking grid rows. For example, double-clicking a row in the Instance Grid opens a Parent Transaction Instance Tree, which provides a detailed breakdown of all sub-events that occurred as part of the selected transaction.

You can return to these test results to track progress as you tune performance and reopen the session at a later point to perform more analysis on the data gathered. You can also export the results.

Viewing Results with WebLOAD REPORTER

WebLOAD REPORTER creates analysis reports based on your session data. Even after you complete a session, you can use this data to create both predefined reports and custom reports of your own. You can create portfolios, which are collections of custom reports to run on other sessions. You can also export reports to PDF, Excel, RTF, HTML, or TXT formats to share with others, store with test plans, or mail to team members.

To view results with WebLOAD REPORTER:

1. Click the WebLOAD REPORTER icon on the Console toolbar or select Test View | Open WebLOAD Reporter from the Console menu.

   Data for the last session saved is automatically selected.

2. Double-click the Ten Slowest Transactions icon at the far left or the Ten Slowest Transactions link in the Report List tree or select the Ten Slowest Transactions item from the Reports menu.

   The report will be generated in the main window.

   Ten Slowest Transactions is an example of the many standard reports available. Reports are displayed in separate windows. Use the Windows menu to move back and forth among reports that you create.

3. Create a custom report by selecting Reports | Custom Session Report... or Reports | Custom Agenda Report..., name the report, click the Save checkbox to save the report, select an axis metric and statistics (from the tree), and click OK.

   You can create both session and Agenda reports. Session reports display a higher level of information for the entire session, including all the Agendas that were run. Agenda reports display detailed data pertaining to one Agenda. In the steps outlined in this Quick Start, you ran one Agenda so the only difference in these reports is the level of detail.

   Each generated report can be either printed or exported to one of the formats that appear on the Export toolbar.

4. Close WebLOAD REPORTER.

5. Close the WebLOAD Console.

This concludes our Quick Start guide for WebLOAD. For additional information please refer to the WebLOAD *User's Guide* or to RadView's web site at http://www.RadView.com (http://www.radview.com/support).

C H A P T E R 5

WebLOAD Installation and Configuration

This chapter contains the instructions for installing WebLOAD on a PC running Windows 2000, Windows 2003, or Windows XP systems.

Communication between hosts in a test session is a critical aspect of WebLOAD testing. This chapter also describes communications between systems. It discusses TestTalk, the WebLOAD network agent, and gives you tips on troubleshooting network problems.

In This Chapter

Installing WebLOAD 51
License Registration and Update 52
Uninstalling WebLOAD 61
Upgrading WebLOAD 62
Communication Between Host Computers 62
Configuration 69
Working with a Firewall 73

Installing WebLOAD

You must have at least 260 MB of available disk space for the complete installation of WebLOAD.

When you install WebLOAD on your computer, the installation program asks you for the components to install.

Install the WebLOAD components as follows:

- Install the full WebLOAD product (with or without WebLOAD REPORTER) on computers that run the WebLOAD Console.
- Install just the Probing Client or the Load Machine software on the network host computers. Do not install the full product on the network host computers.

Notes:

WebLOAD components in each major version will work with all other components of the same version, no matter what the minor revision number. WebLOAD components are not compatible with other components of a different version. For example, all components of WebLOAD Version 7.x will work with each other. Version 7.0, 7.1, etc, are all compatible with each other. However, components of Version 7.x are not compatible with components of Version 6.x or Version 5.x. Verify that the WebLOAD components have been upgraded to Version 7.x on all participating hosts. Running WebLOAD Console Version 7.x with an earlier version of Load Generator (running on different Hosts) may cause unexpected behavior.

The WebLOAD license file is limited to the computer system (machine) on which WebLOAD is initially installed. Before installing WebLOAD, make sure you are installing on the machine with which you intend to work.

When installing WebLOAD with Unicode support on a system running an English MS Windows, Japanese must be configured as the default Windows language.

To install WebLOAD on your system:

1. Browse to the location of the WebLOAD executable (.exe) installation file.
2. Double-click the file.

   The WebLOAD Installation Wizard appears.
3. Follow the instructions on your screen.
4. When the Installation Wizard prompts for the component to install, select the desired component. For Load Machines and Probing Clients, select Load Generator / Probing Client.

When WebLOAD installation is completed, the WebLOAD License dialog box automatically opens to complete the registration process. License registration is discussed in License Registration and Update (on page 52).

License Registration and Update

Your WebLOAD license must be registered before you can start working with WebLOAD. WebLOAD licenses that have expired must also be renewed or updated before you can continue working with WebLOAD. All license registration and updating is accomplished through the Update License dialog box.

Standard License Registration

The Update License dialog box can be opened in one of two ways:

- At the time of WebLOAD installation, when the WebLOAD Installation Wizard is finished, the License dialog box opens automatically,

  Note: After a WebLOAD installation, you may be prompted to restart your computer before the License dialog box appears. In this case, the License dialog box will appear automatically after the computer restarts.

  -Or-

  At any other time, from the Windows desktop, click on Start | All Programs | TestView | Tools | Update License.

The License dialog box includes three tabs:

- Host ID
- Select License
- License Information

These tabs are used to register and update your WebLOAD authorization license information.

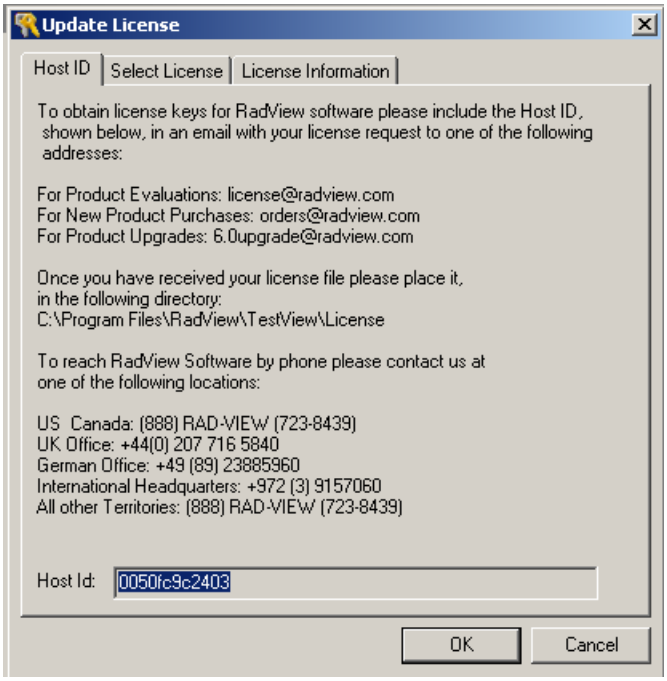
To complete or update your license registration:

1. If you are registering a new license after an initial WebLOAD installation, go to the Update License dialog box. The Update License dialog box is opened automatically after a successful WebLOAD installation is completed,

   -Or-

   If you are updating your license registration after your current license has expired, from the Windows desktop, click on Start | All Programs | TestView | Tools | Update License.

2. Select the Host ID tab.



3. Create an email message with the following information:

   a. Copy the Host ID displayed in the text box into the email.

   b. Add your name, company, address, and phone number to the email message.

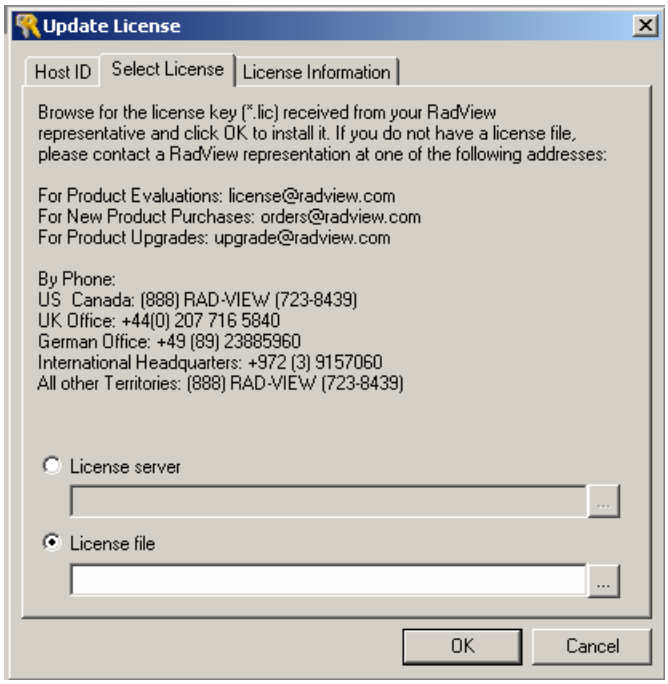
4. Send the email to license@radview.com.

   A WebLOAD license file (`*.lic`) will be sent to you.

5. After receiving the file, save it to your hard drive in the following directory:

   `Program Files\RadView\WebLOAD\License`

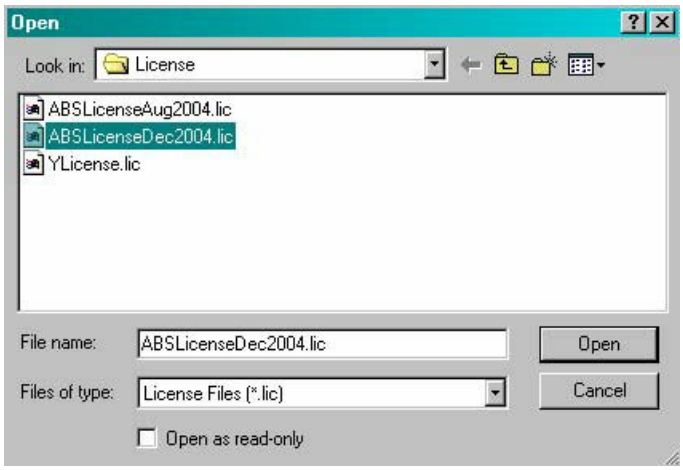
6. Select **Start | All Programs | TestView | Tools | Update License** to open the **Update License** dialog box and install the license key.
7. Select the **Select License** tab.



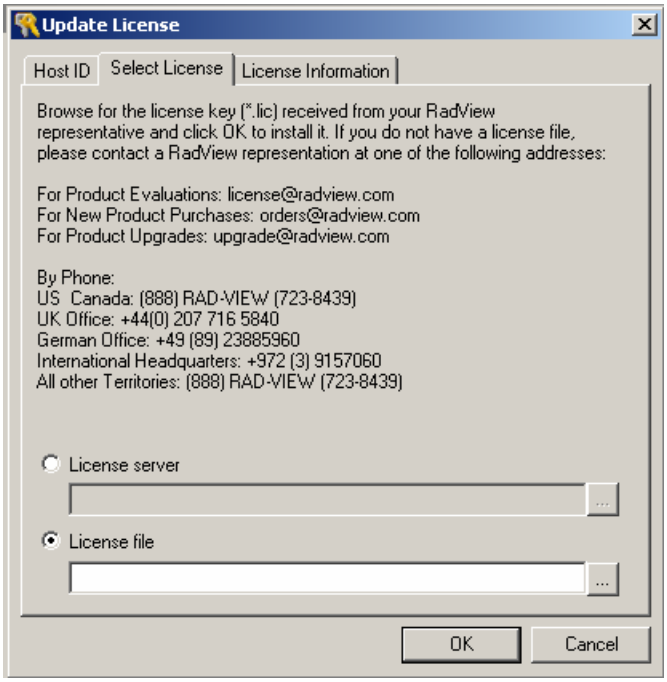
8. By default, the **License file** radio button is selected. This assumes that your license file is located on your local computer system.

   For information on installing a floating license, or connecting to a license server, see Floating License Server (on page 57).

9. Click on the Browse (...) button next to the License file text box to browse to the location of the license (`*.lic`) file sent to you by a RadView representative.



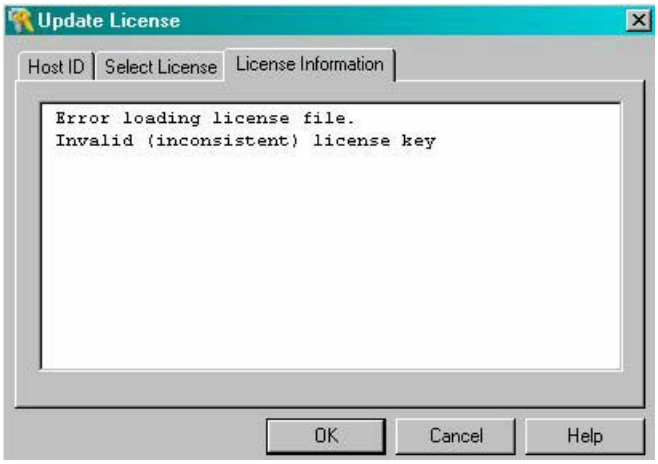
10. Select the correct license file and click Open to return to the Update License dialog box.



11. Click OK to load the new license.
12. If the license registration was successful, a message box appears indicating a successful license update.

You can now begin working with WebLOAD.

13. If the license registration was not successful, an error message appears in the License Information tab.



If you are not able to successfully register a valid license file, contact RadView Support for assistance.

Floating License Server

The Floating License Server is a stand-alone tool that manages all resource distribution to WebFT, FTRunner, TestView Manager, and RM Workstations, which is the machine used to run load in a floating license environment. It is the method that controls the number of users allowed to access the product concurrently.

A floating license for virtual clients is available and defined as a pool of virtual (probing) clients that can concurrently be used by members of a testing team and monitored by the license server.

During installation of the license server, you are prompted for the floating license file that defines:

- The number of concurrent applications for each product (RM Workstation, WebFT, FTRunner, and TestView Manager).
- The number of concurrent Virtual and Probing Clients.

Installation also creates a FlexLM Service that controls the operation.

The license server displays a list of currently connected RM Workstations and the number of resources (Virtual and Probing Clients) each is using. A Floating License Server enables a testing team to share resources.

This section describes how to connect to the Floating License Server and how to view License details. For more information about running WebRM, WebFT, and FTRunner in Floating License Mode, updating Floating License Resources, revoking Resources, or working with the Floating License Server Console, see the TestView User's Guide.

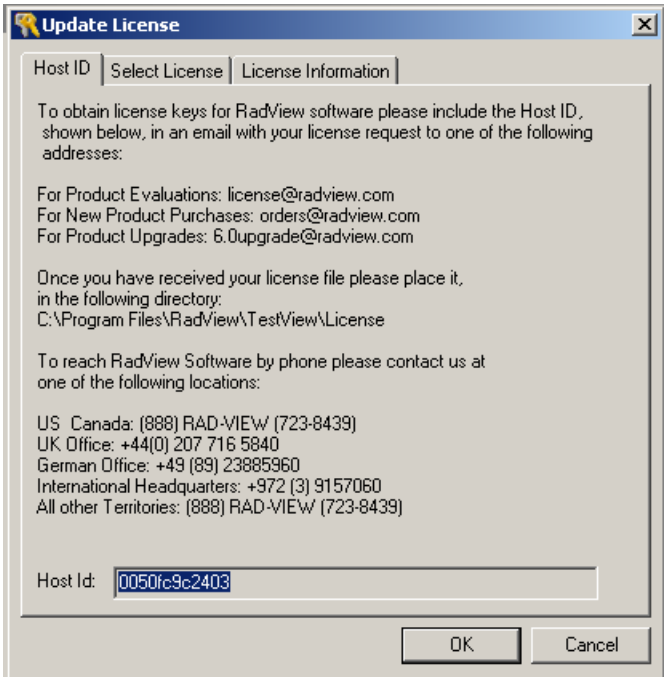
Connecting to the Floating License Server

To connect to the Floating License Server:

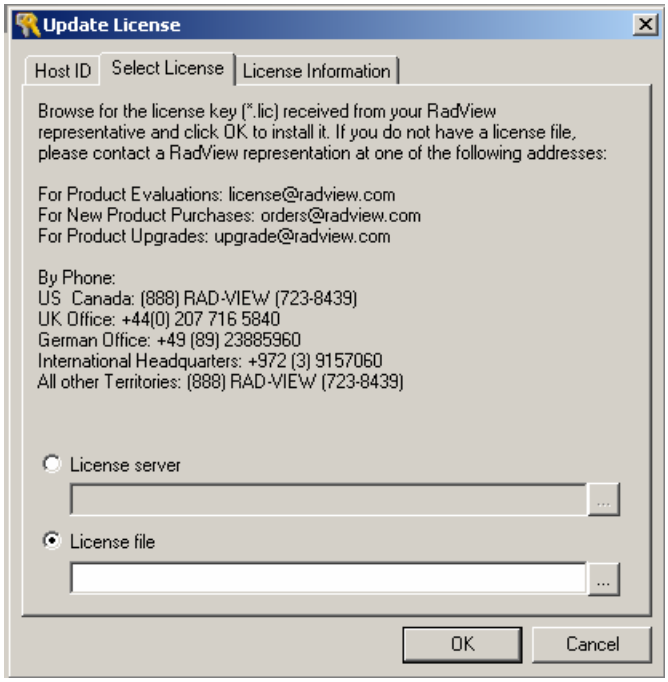
1. After installing TestView, the Update License dialog box opens, enabling you to select a valid license key,

   -Or-

   To open the Update License dialog box manually, select Start | All Programs | TestView | Tools | Update License.

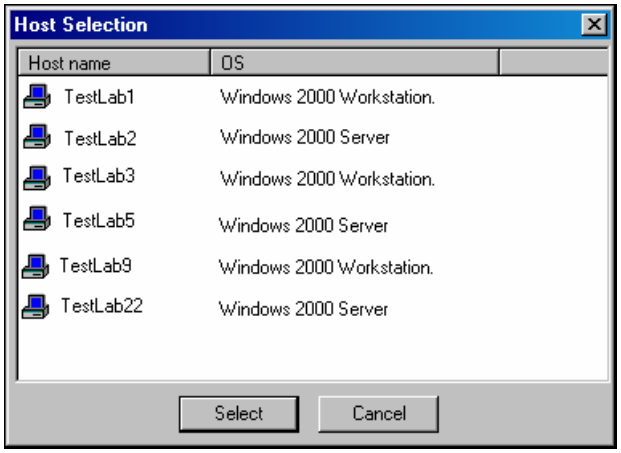


2. Select the Select License tab.



3. Select the License Server radio button, and press the browse button to select the license server.

   The Host Selection dialog box opens.

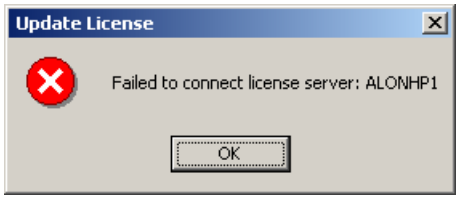


4. Select the license server host and click Select to return to the Update License dialog box.
5. Click OK.

   A connection to the selected license server is attempted.

If the connection is successful the following message displays:

If the connection is not successful, the following message displays:



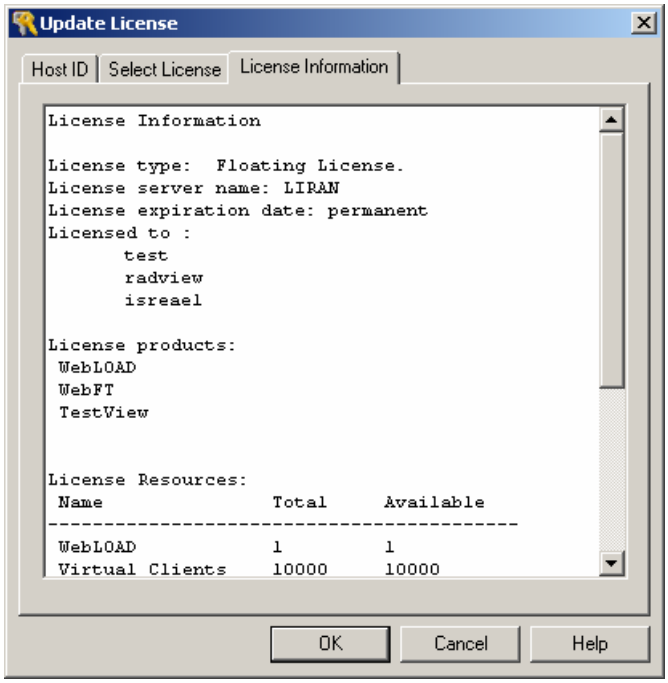
Viewing License Details

After selecting the license server, you can view the current license details.

To view license details:

1. Select Start | All Programs | TestView | Tools | Update License.
2. Select the License Information tab.

   All of the current license information is displayed, including free resources, license features and user details.



For more information about running WebRM, WebFT, and FTRunner in Floating License Mode, updating Floating License Resources, revoking Resources, or working with the Floating License Server Console, see the TestView User's Guide.

Uninstalling WebLOAD

To uninstall WebLOAD:

1. Click Start | Settings | Control Panel.
2. Double-click the Add/Remove Programs icon.
3. Click the Install/Uninstall tab.
4. Click WebLOAD, and then click Add/Remove.

   The Uninstall Wizard appears.

Note: It is only necessary to delete the WebLOAD program files and not the user files. WebLOAD Load Templates, Load Sessions, and Agenda files can be used with all WebLOAD upgrades.

Caution: Close TestTalk before uninstalling WebLOAD. Failure to close TestTalk results in outdated files remaining on your disk, which could adversely affect future installations of the program.

Upgrading WebLOAD

To upgrade WebLOAD:

1. Close TestTalk.
2. Install the new version of WebLOAD.

   For installation instructions, see Installing WebLOAD (on page 51).

Note: WebLOAD components in each major version will work with all other components of the same version, no matter what the minor revision number. WebLOAD components are not compatible with other components of a different version. For example, all components of WebLOAD Version 7.x will work with each other. Version 7.0, 7.1, etc, are all compatible with each other. However, components of Version 7.x are not compatible with components of Version 6.x. This applies to executable components only. All WebLOAD data files, including WebLOAD Load Templates, Load Sessions, and Agenda files, can be used with all WebLOAD upgrades.

Communication Between Host Computers

The Console must be able to communicate with all of the hosts in your test session. All of the hosts must be on a common Local Area Network (LAN) or Wide Area Network (WAN). In addition to the physical network, all of the systems must be running the TestTalk network agent, which is installed with your WebLOAD program.

SNMP

Simple Network Management Protocol (SNMP) enables communication between systems in your Load Session. If SNMP service is installed on your system, it runs automatically in the background.

Installing SNMP Service on a Windows 2000 Platform

To install SNMP service on Windows 2000:

1. Select Start | Settings | Control Panel | Add/Remove Programs.

   The Add/Remove Programs dialog box displays.

2. Click Add/Remove Windows Components.

   The Windows Components Wizard dialog box displays.

3. Click the Management and Monitoring Tools check box and click Next.

   Windows asks for the location of the Windows installation files.

4. Specify the location of the Windows installation files and click OK.

   SNMP service is installed.

Installing SNMP for WebLogic

The WebLogic SNMP agent will not work as a subagent for the OS master agent. The workaround is to make WebLogic act as master (on port 161) and delegate to the native OS agent.

1. Modify winnt\system32\drivers\etc\services file from:

   …

   `snmp 161/udp`

   …

   to:

   `snmp 1161/udp (any free port will do here)`

2. Modify the `startsnmpagent.cmd` file in the WebLogic root. For example:

   ```
   %JAVA% weblogic.SNMPAgent -password mypassword -serverURLs
   t3://localhost:7001 -snmpPort 161 -trapDestinations
   localhost:162 -otherAgents 1161:.1.3.6.1.2.1.
   ```

   The `-snmpPort 161` flag makes WebLogic the master agent.

   The `-otherAgents 1161:.1.3.6.1.2.1.` flag tells this agent to delegate the 1.3.6.1.2.1 subtree to port 1161 (where the native agent is running).

3. Run this file and ensure the agent is restarted on the 1161 port.

4. Use system command `netstat -a -n -p udp` to view the active udp ports.

   Both udp 161 and udp 1161 should appear in the resulting list. The WebLogic appserver should be running.

For more information see:
http://www.weblogic.com/docs51/admindocs/snmpagent.html#use_agent
(http://www.weblogic.com/docs51/admindocs/snmpagent.html#use_agent)

Installing SNMP for ORACLE

The ORACLE SNMP agent will not work as a subagent for the OS master agent. The workaround is to make ORACLE act as master (on port 161) and delegate to the native OS agent.

1. Modify winnt\system32\drivers\etc\services file from:

   …

   `snmp 161/udp`

   …

   to:

   `snmp 1161/udp (any free port will do here)`

2. On the ORACLE machine perform following steps:

   a. Modify the `oracle\ora81\network\admin\encaps.cfg` file by adding the following line:
   `AGENT AT PORT 1161 WITH COMMUNITY PUBLIC`

   a. Modify the `oracle\ora81\network\admin\master.cfg` file, specifying that ORACLE runs on port 161.

   b. Start following services:
   `OracleSNMPPeerMasterAgent`
   `OraclePeerEncapsulator`

TestTalk for Windows

TestTalk is the WebLOAD program that enables communication between the systems in your Load Session. TestTalk is installed with the Console and Load Generator software. All of the hosts in the test session must be running TestTalk in the background for the test to operate.

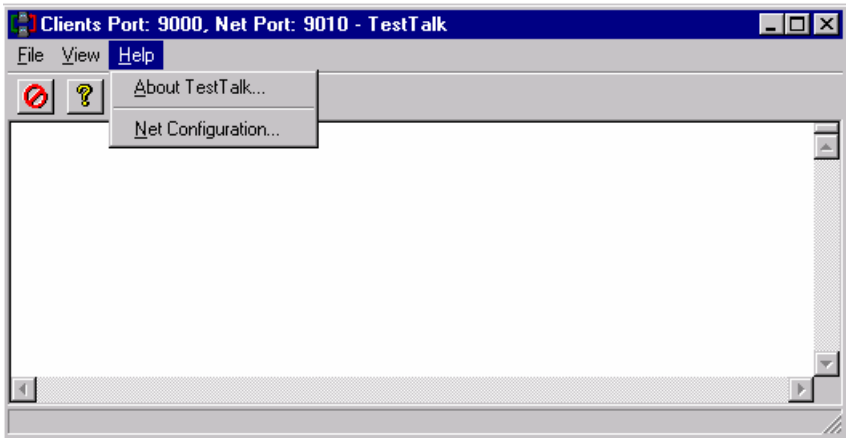
Opening TestTalk

TestTalk is opened automatically when you open the Console.

To open TestTalk on Load Machines not running the Console:

- Select Start | Program | TestView | Tools | TestTalk.

  The TestTalk window appears.



Caution: Do not close TestTalk in the middle of a test session. This can cause unexpected behavior. When shutting down the system, be sure to shut down the Console first and only then shut down TestTalk.

The TestTalk window contains the following menus:

Menu	Function
File	Contains the Exit option. Use this option to Close the TestTalk program. Clicking the Windows Close button does not close TestTalk. This is a safety feature that prevents you from accidentally closing TestTalk when the Console is active.
View	Contains the TestTalk display configuration options. The first two options in this menu are: ◆ Toolbar ◆ Status Bar Toggle these options on/off to display the Toolbar or the Status Bar The Clear View Window option clears the display window.
Help	Contains the options for viewing the program information and information about your system. The Help menu contains two options: ◆ Select About TestTalk to display the information about this program. ◆ Select Net Configuration to display your system configuration information.

Viewing the Network Configuration

To view the network configuration:

1. Select Help | Net Configuration.

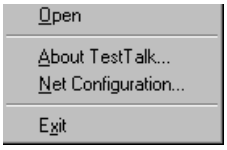
   The following TestTalk message box appears.

   The information displayed in the TestTalk message box includes:

Field	Description
Computer's name	The name of your computer. You must know the name of each of the hosts to configure your test session. If you are not familiar with the name of your host, open TestTalk at that host and view this name in this message box.
Computer's address	The IP address. You need this information to ping a host if you want to verify that it is connected to the network.

   While TestTalk is running, the TestTalk icon is displayed on the Status bar in the lower right hand corner of the screen.

2. Right click on the TestTalk icon to display the following menu:



Menu Item	Description
Open	Opens the TestTalk window.
About TestTalk	Displays the program version information.
Net Configuration	Displays the Server Configuration (as shown above).
Exit	Closes TestTalk.

The Required Connections

WebLOAD can only run when the hosts participating in the test session are properly connected. Each participating host must have two types of connections. If either of these connections is not complete, you may experience problems running WebLOAD.

- Physical connection

  The hosts in the test session must all be physically connected to the network. They must be turned on and the connection must be live.

- Connected by network agent

  All of the hosts in the test session must be running TestTalk as a background process. Only the host controlling the tests must be running the Console.

Testing Your Connection

To test your connection do the following:

1. Test the host's physical connection to the network by pinging the appropriate host. This is described in Pinging a Host *(on page 67)*. If the host is not connected properly to the network, connect it and try to run WebLOAD again.
2. If the ping is properly acknowledged, verify that the TestTalk program is up and running properly on the host. This is done by checking the host physically to verify that the TestTalk program is functioning. If TestTalk is not running, open TestTalk and retry the program.

   TestTalk runs in the background. Error messages are displayed on the program window. Check this window for any error messages that may explain the problem.
3. If there is still a problem with the host, you may be experiencing problems not related to TestTalk. Restart the system and rerun WebLOAD.

Pinging a Host

You can test a host's physical connection to the Console by pinging the remote host. A ping is a signal that is sent from the Console to a remote host. The response that the Console receives from the ping indicates the status of the connection to the remote computer.

- From the operating system, type ping systemname.

  Example:

  ```
  c:> ping systemname
  ```

Acknowledgments from a Ping

You can receive several types of acknowledgments from a ping:

- Ping `systemname` with 32 bytes of data.

  ```
  reply from 192.114.35.1: bytes = 32 time = 1ms TTL = 255
  reply from 192.114.35.1: bytes = 32 time = 1ms TTL = 255
  reply from 192.114.35.1: bytes = 32 time = 1ms TTL = 255
  reply from 192.114.35.1: bytes = 32 time = 10ms TTL = 255
  ```

  This acknowledgment indicates that the host is connected properly. If you receive this response, and WebLOAD continues to return an error, test the TestTalk network agent.

- Bad IP address `systemname`.

  Indicates that the remote host is not connected properly. The host must be physically connected to the network, and configured properly.

- Request timed out.

  Indicates that the IP address is known to the Console, however the host is not currently working. The host must be turned on and tested again.

Troubleshooting Communications

On occasion, you may have trouble with a WebLOAD host. Refer to the following table for possible solutions.

Problem	Solution
At the beginning of a test session, the host prompts: All hosts are stopped. Indicating that none of the hosts are accessible and the Domain Name Server (DNS) cannot recognize a host name.	Test your connection, as described in Testing Your Connection (on page 67).
At the beginning of a test session, a host prompts: Load Session Setup -timed out, not completed. Indicating that at least one host is not accessible and the Domain Name Service (DNS) cannot recognize a host name.	Test your connection, as described in Testing Your Connection (on page 67).

Problem	Solution
Your system starts, but the test session times out during operation.	Test your connection, as described in Testing Your Connection (on page 67). Check the names of the computers in the TestTalk Help menu, under Net Configuration. Verify that there is complete consistency between the names of the computers and the names used in WebLOAD.
WebLOAD reports unexpected results from a test.	Access the Application Being Tested (ABT). If you have a good connection, the test results are accurate.
The system issues the message: Error loading Agenda...	The Console must specify the complete path of the Agenda that is used by the remote host. For example, if the Agenda is on a shared disk, the name of the disk in the path specified by the Console, must be the same name used by the remote host.

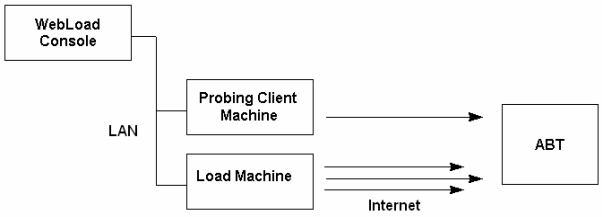
Configuration

A Load Session includes the following components:

Component	Description
Console	The Console controls set up, manages a Load Session, and displays the Load Session results. It is recommended that the Console reside on its own separate system and not be installed on systems running Load Generators or Probing Clients.
ABT (Application Being Tested)	The Web application to be tested. WebLOAD can test any Web application that can be accessed through a Web browser, on the Internet or intranet. The ABT requires no special software.
Load Machine	The host(s) that runs Load Generators. The Load Generators bombards the ABT with user-defined transactions for testing scalability and integrity of the ABT.

Component	Description
Probing Client Machine (optional)	The host(s) that run Probing Clients. A Probing Client runs at the same time as load testing. WebLOAD generates exact values for the Probing Client performance as opposed to averages for Load Generator performance.
WebRM	WebRM distributes and circulates WebLOAD testing resources (Virtual Clients and Probing Clients) amongst users on a "need to use" basis. With WebRM, every WebLOAD Console can operate in Standalone Workstation mode or Connected Workstation mode. For more information, see Connecting to WebRM *(on page 121)*.

The following diagram shows a simple WebLOAD test configuration. The Load Machine and the Probing Client Machine both access the ABT. The Console controls the test from a separate system. All three computers are networked.

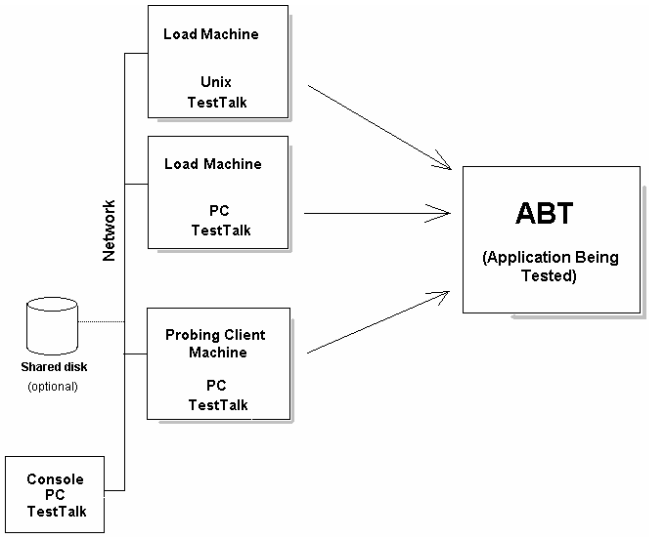


Required Software for each Load Session Component

Test Session Component	Required Software for Running Test Session Component
Console	◆ TestTalk ◆ Console ◆ WebFT (optional)
ABT (Application Being Tested)	None
Load Generator	TestTalk
Probing Client (optional)	TestTalk
Shared disk (optional)	None

Example of a Test Configuration

The following diagram illustrates a typical WebLOAD test configuration.



This test session has three participating host computers. Two hosts act as Load Machines, and one host is a Probing Client Machine. In this configuration, the Console is run on a separate system. All systems are on a network with a shared disk that contains the Agenda.

The Console

In the diagram above, the Console is a node on the network along with the host computers in the test session, and the shared disk. The Console communicates with the hosts that are testing the ABT through the TestTalk network agent. The Console controls the test setup, management and reporting.

Note: Running a Load Machine on the same system running the Console software is not recommended.

The Load Machine

Web application tests can be run from either one Load Machine, or from several hosts simultaneously. The test load can be greater if the load is divided between multiple systems. For example, if one host can simulate 100 simultaneous Virtual Clients, a second host with the same capacity can simulate an equal number of Virtual Clients. When the two hosts run simultaneously, they can simulate double that number.

Probing Client

The Probing Client resides on the network with the other hosts in the test session. The Probing Client tests the performance of the ABT while it is being bombarded by the Load Machines. In

this test session there is one Probing Client, however WebLOAD can support multiple Probing Clients in a test session. WebLOAD generates exact values for the Probing Client performance as opposed to averages for Load Machine performance.

Sharing an Agenda

You can increase the size of the test load by distributing the load generation tasks over multiple Load Machines. You may want all of these Load Machines to test the ABT using the same Agenda. There are several ways to share an Agenda over multiple systems.

Shared disk:

You can configure a shared disk that can be accessed by all the systems on the network, as shown in the previous diagram. The Agenda can then reside on the shared disk, and can be used by all systems for testing. Thus there is only one copy of the Agenda, and all the relevant hosts run it from the shared disk.

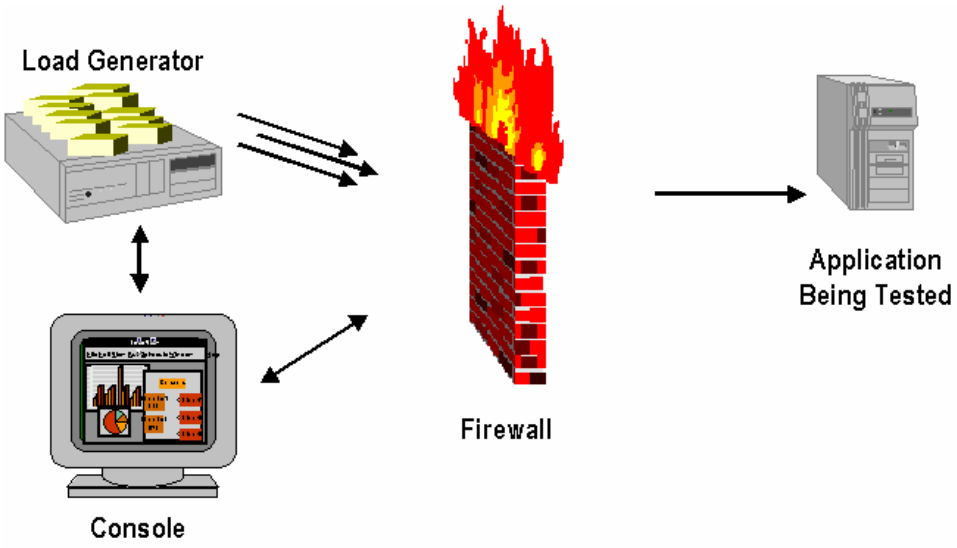
In this configuration, you must make sure that in the Agenda configuration, all hosts refer to the shared disk and are mapped to the same logical name. On one host the logical name of the shared disk may be “K” while on another host the logical name may be “L.” Each host must configure the Agenda using the correct logical name.

Copying the test program to each host:

Copy the Agenda to each host participating in the test. Verify that the most updated copy of the Agenda resides on each system.

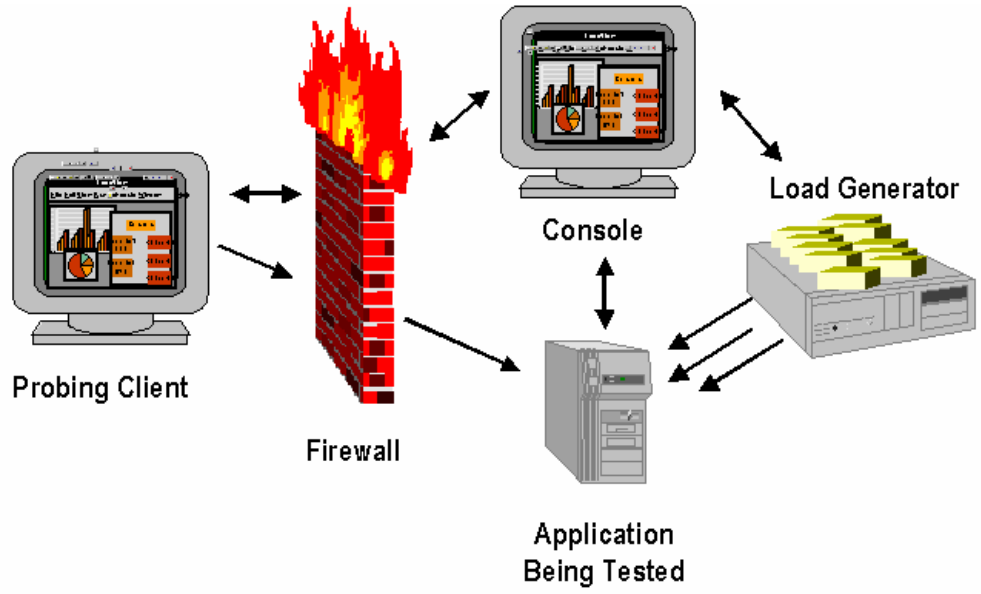
Working with a Firewall

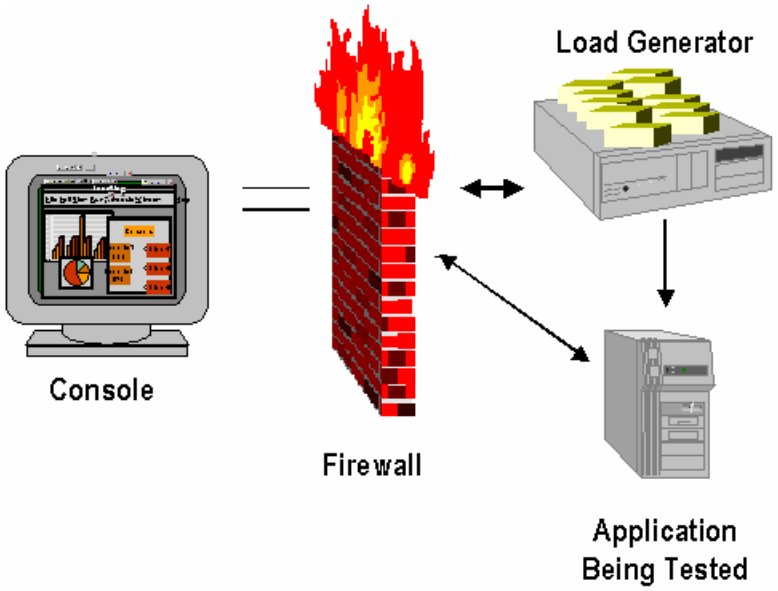
If your machine configuration is similar to the one illustrated below, no special WebLOAD configuration is necessary.



In some cases you may want to use machine resources in the test session that reside on different sides of a firewall. In this case, you must change your initial settings to enable WebLOAD to accommodate this unique configuration.

The following diagrams illustrate situations where the Load Machine and the Console are on opposite sides of the firewall.





The following configuration is required to enable data transfer through the firewall. Generally, the rules that need to be added to the firewall are:

- Allow outgoing TCP from the Console all ports to Probing Client port 9000.
- Allow incoming TCP from Probing Client port 9000 to Console all ports (options=ackonly).
- Allow outgoing UDP from Console port 137 to Probing Client port 137.
- Allow incoming UDP from Probing Client port 137 to Console port 137.

C H A P T E R 6

WebLOAD Console Features

The WebLOAD Console is where you perform the majority of your load / performance testing activities.

In This Chapter

The WebLOAD Console 77
Console Elements 79
Edit Mode and Run Modes of Operation 96
The Console Menu Options 98
The Console Toolbar 105

The WebLOAD Console

The Console consists of the following:

- Menu bar
- Toolbar
- Session Tree
- Results window
- Status bar

The Console manages the load session. It also:

- Configures load session hosts and Agendas.
- Schedules load session Agendas.
- Monitors the application's performance and integrity under the generated load.
- Manages the load session as it is running, allowing you to pause, stop and continue load session components as needed.
- Displays the current performance of the ABT and provides a final performance report for Virtual Clients and Probing Clients.
- Manages exporting of test reports.

The WebLOAD Console component controls your test session.

From the Console, you can:

- Specify the program (Agenda) that the test session runs.
- Select the number of Virtual Clients to run.
- Define the hosts participating in the load session.
- Schedule tests.
- View performance reports.

Opening the Console

To open the Console:

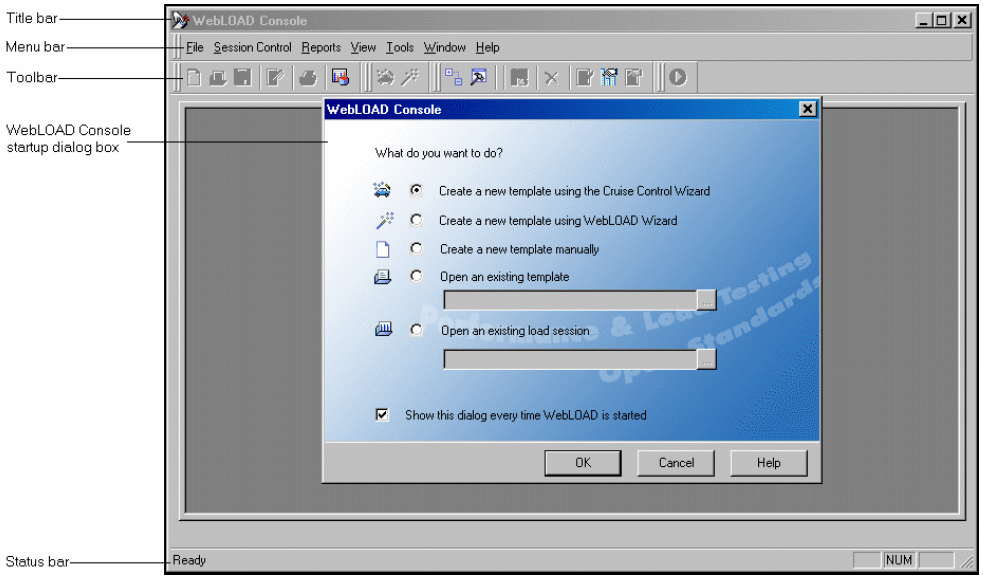
- Select Start | All Programs | TestView | WebLOAD Console,

  -Or-

  From your desktop, click the Console icon.

  The WebLOAD Console shortcut screen appears displaying a variety of shortcuts to the various WebLOAD components.

The Console screen displays as follows:



The Console is operated through its toolbar and menus at the top of the screen. Tests are configured through a series of dialog boxes.

Console Elements

When you first start WebLOAD, you see the Console interface in Edit mode. The Console window displays different components in the two different modes:

- Edit Mode where you create your templates.
- Run Mode where you run your sessions and see the results.

The Console interface consists of the following elements.

Title Bar

Displays the name and path of the current template or session.

Menu Bar

Displays the commands you use to work with WebLOAD. Beside the standard File, View, Window and Help menus, menus are provided to access functions specific to template creation and session management such as Session Control and Reports. For more information on the menus, see Edit Mode and Run Modes of Operation (on page 96).

Context Menus

Contain shortcuts to frequently performed actions.

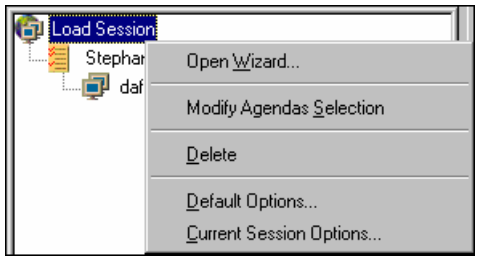
To open a context menu:

- Right-click on the object you are using.

The specific list of shortcuts available depends on the area where you click the right mouse button.

For example, the context menus displayed when you right click on the load session icon in the session tree in edit mode enables you to:

- Open the WebLOAD Wizard.
- Modify Agenda selections.
- Delete the load template.
- Access the default and current session options.



Toolbar

Provides quick access to commonly used commands. You click a button on the toolbar once to carry out the actions represented by that button.

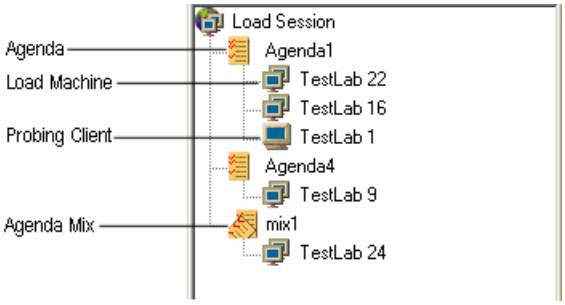
For more information on the Console toolbar, see The Console Toolbar (on page 105).

Status Bar

At the bottom of the Console window, the status bar displays information about operations in progress.

The Session Tree

The session tree displays in the left pane of the Console screen, and gives you a complete graphical overview of the test session including the Agendas run, and the hosts running each Agenda. The icons adjacent to the tree items enable you to view your test activity at a glance.



Opening the Session Tree

The session tree displays when you are configuring a test session or running a test. If the session tree is not open, you can open the session tree as follows:

- Select Session Control | Open Session Tree,

  -Or-

  Click on the Session Tree icon in the Console toolbar.

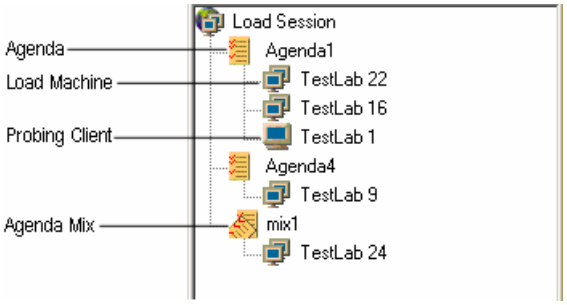
During the test configuration phase, the session tree is a convenient interface for editing the test session. Double-clicking on the load session icon starts the WebLOAD Wizard. Double-clicking on a tree node opens the dialog box for configuring that tree item. Right-clicking on a tree item prompts a menu that enables you to perform a number of functions on the test session. As you change your configuration through the Console configuration dialog boxes, the changes that you make are reflected in the session tree.

Note: You can also access the configuration dialog boxes through the toolbar and the menus.

While your test session is running, the session tree displays its progress. The session tree displays the Agendas configured for running, the participants in the test, and their status. For more information on the status of the test, see Status Icons (on page 82).

The Session Tree Structure

- The “root” of the tree, in the upper-most line, displays the load session icon.
- The first level branches display the Agendas to be run during the test session. The Agendas may be single Agendas or a Mix of Agendas. The icons displayed next to the Agenda name indicate its status. For more information on the status of session components, see Status Icons (on page 82).
- The second level branches present the hosts participating in the test session. These hosts can be Load Machines or Probing Clients. A different icon represents each host type. The icons displayed adjacent to the hosts indicate their status. Several hosts can run the same Agenda.



Session Tree Icons

WebLOAD displays icons on each line of the session tree, making it easy to view your test elements and activity.

The session tree icons are described in the following table:

Icon	Session Component
	Load Session - The root of the session tree.
	Agenda - A single test script.
	Mix of Agendas - A combination of test scripts.
	Load Machine - A machine used for generating Virtual Clients.
	Probing Client Machine- A machine used for generating Probing Clients.

Status Icons

During runtime, status icons appear adjacent to each load session item (hosts, Agendas) in the session tree, indicating the item's status.

The status icons are described in the following table.

Icon	Description
	No current activity.
	Initialization completed successfully. Ready to run.
	Currently running.
	Temporarily paused. Remainder of load session continues running as usual.
	Currently stopped.
	An error has occurred.

Navigating the Session Tree

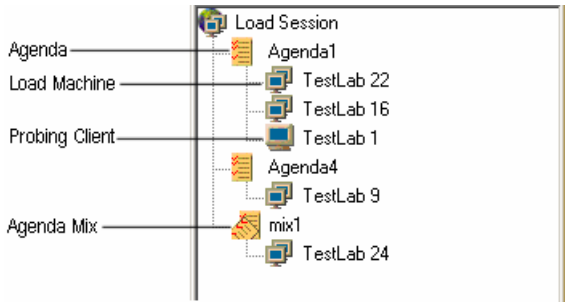
You can move through the session tree by expanding and collapsing nodes.

The following table lists the keystrokes you can use to move in the session tree:

Key	Result
Up arrow	Moves the selection up one item in the session tree.
Down arrow	Moves the selection down one item in the session tree.
Right arrow	Expands the selected item. If the selected item does not contain hidden items, behaves like down arrow.
Left arrow	Collapses the selected item. If the selected item doesn't contain exposed items, behaves like up arrow.
Page Up / Home	Moves the selection to the first item in the session tree.
Page Down	Moves the selection to the bottom item visible in the pane.
End	Moves the selection to the last item in the session tree.
Plus sign “+” on the numeric keypad	Expands the selected node.
Minus sign “-” on the numeric keypad	Collapses the selected node.
Asterisk “*” on the numeric keypad	Expands the entire tree below the selected node.

The Session Tree in Edit Mode

In edit mode, the session tree provides a complete graphical overview of the test session.



The following is a line-by-line explanation of the above Session Tree:

Display		Explanation
	Load Session	Identification icon representing the Session Tree.
	Agenda1	Describes one of the Agenda files (*.pft / *.js) used for running this test session.
	TestLab 22	Describes one of the Load Machines running the Agenda, Agenda1. The Load Machine icon is followed by the host name, in this case “TestLab 22”. A test session may include multiple Load Machines.
	TestLab 16	Describes one of the Load Machines running the Agenda, Agenda1. The Load Machine icon is followed by the host name, in this case “TestLab 16”. A test session may include multiple Load Machines.
	TestLab 1	Describes one of the Probing Client Machines used in this test session. The Probing Client Machine icon is followed by the host name, in this case “TestLab 1”. A test session may include multiple Probing Client Machine hosts
	Agenda4	Describes the one of the Agenda files (*.js) used for running this test session.
	TestLab 9	Describes one of the Load Machines running the Agenda, Agenda4. The Load Machine icon is followed by the host name, in this case “TestLab 9”. A test session may include multiple Load Machines.
	Mix1	The Mix file (`*.mix`) to be run during the load session. The icon is followed by the name of the Mix, in this case “Mix1”.
	TestLab 24	Describes one of the Load Machines running the Mixed Agenda, Mix1. The Load Machine icon is followed by the host name, in this case “TestLab 24”. A test session may include multiple Load Machines.

Right-Click Options in Edit Mode

Using the session tree you can define and modify load templates.

The following options can be accessed by right clicking on a session tree component in edit mode:

Tree Item	Options	Description
Load Session	Open Wizard	Open the WebLOAD Wizard.
	Modify Agendas Selection	Open the Agenda / Mix Selection dialog box to modify the selections.
	Delete	Delete the selected WebLOAD component and all sub-components.
	Default Options	Open the Default Options dialog box and set new default options for each Agenda added to a load session.
	Current Session Options	Open the Default Options dialog box and set options for each Agenda currently defined in the load session.
	Notification Manager	Open the Notification Manager dialog box for defining the conditions under which you want to receive a notification and a notification action.
Agenda/Mix	Open Wizard	Open the WebLOAD Wizard.
	Modify Load Machine Selection	Open the Load Machine Selection dialog box and modify the Load Machines defined for the session.
	Modify Load Machine Schedule	Open the Manual Schedule dialog box and modify the test schedule.
	Open	Open the Agenda in WebFT for editing.
	Delete	Delete the Agenda.
	Rename	Rename the Agenda.
	View JavaScript	Open a window displaying the JavaScript for the Agenda.
	Copy Load Machines and Schedule	Available for Agenda only. Copy the Load Machines and the load schedule defined for this Agenda to the clipboard.
	Paste Load Machines and Schedule	Available for Agenda only. Paste the Load Machines and the load schedule from the clipboard.

Tree Item	Options	Description
	Agenda Options	Available for Agenda only. Open the Agenda Options dialog box. For more information on these settings, see Setting Agenda Options *(on page* 221*)*.
	Mix Options	Available for Mix only. Open the Mix Creation dialog box, enabling you to select an Agenda from the mix for which to change the options.
Load Machine and Probing Client Machine	Modify Schedule	Open the Load Machine Schedule dialog box enabling you to modify the scheduling parameters.
	Replace	Replace the host (from the host list)
	Delete	Delete the host
	Copy Schedule	Copy the schedule defined for the Load Machine to the clipboard.
	Paste Schedule	Paste the schedule from the clipboard.

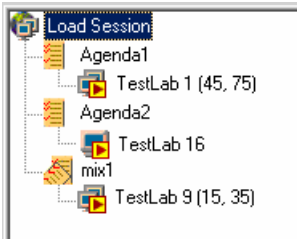
The Session Tree in Run Mode

The session tree displays the complete configuration of the current load session. Status icons reflect the status of your load session.

Through the session tree you can see:

- The Agendas running in the test session
- The hosts running each Agenda
- Each Agenda and host's current operating status

Icons are displayed on each line of the tree, making it easy to view the test activity.



The following is a line-by-line explanation of the above session tree:

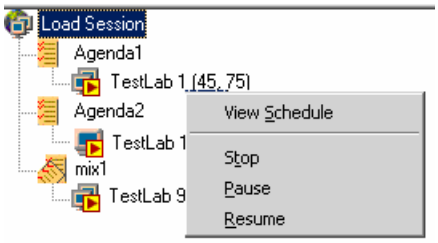
Line Num	Icon		Explanation
1		Load Session	Indicates a session tree.
2		Agenda1	The Agenda file to be run during the load session
3		TestLab 1 (45,75) [CONNECTED]	Describes the Load Machine for this test session. The Load Machine icon, along with its status (currently running), is followed by the host name (TestLab 1). A test session may include multiple Load Machines. Two numbers follow each Load Machine entry: ◆ The first number is the current load being generated. ◆ The second number is the maximum load that Agenda schedules it to generate over the course of a test session. In this example, TestLab 1 is running under Agenda1, and has a maximum scheduled load of 75 Virtual Clients and is currently generating a load of 45 Virtual Clients.
4		Agenda2	The Agenda file to be run during the load session.
5		TestLab 16	Describes the Probing Client Machine host for this test session. The Probing Client icon, along with its status (currently running), is followed by the Host name (TestLab 16). A test session may include multiple Probing Client hosts.
6		Mix 1	The Mix file to be run during the load session
7		TestLab 9 (15,35)	Describes the Load Machine for this test session. The Load Machine icon, along with its status (currently running), is followed by the host name (TestLab 9). Two numbers follow each LoadMachine entry: ◆ The first number is the current load being generated. ◆ The second number is the maximum load that Mix schedules it to generate over the course of a test session. In this example, TestLab 9 is running under Mix1, and has a maximum scheduled load of .35 Virtual Clients and is currently generating a load of 15 Virtual Clients

To the right of each Load Machine two numbers are displayed in parentheses, for example TestLab 1(45, 75). These display the load size currently being generated; followed by the maximum load size (number of Virtual Clients) the Agenda is scheduled to generate. The current load size number will change according to the load schedule defined in the Schedule dialog box. Before a test session begins, the 'current' load size being generated is 0. In most

modes of operation, the current load size will never exceed the maximum load size scheduled for that Agenda listed to the right of the current value.

Note: Using Throttle Control, you can increase the current load size to exceed the maximum load size - but not while running a Cruise Control session.

In run mode you can right-click on an item in the session tree to view the test parameters defined in the session.



Right-Click Options in Run Mode

While in run mode, the following options can be accessed through the right mouse button:

Tree Item	Options	Description
Load Session	View Agendas Selection	Open the Agenda / Mix Selection dialog box.
	Stop All / Pause / Resume	Control the test in progress.
	View Goals	Open the View Goals message box (available only during Cruise Control Sessions) to display the goals defined in the test session.
	Throttle Control	Open the Throttle Control dialog box, enabling you to dynamically change the load generated on the fly.
	Default Options	Open the Default Options dialog box enabling you to view the default options for each Agenda added to a load session.
	Current Session Options	Open the Default Options dialog box enabling you to view the options for each Agenda currently defined in the load session.
	Notification Manager	Open the Notification Manager dialog box for defining the conditions under which you want to receive a notification and a notification action.
Agenda / Mix	View Load Machine Selection	Open the Load Machine Selection window.

Tree Item	Options	Description
	View Load Machine Schedule	Open the Load Machine Schedule window displaying the schedule defined for the Agenda.
	Stop / Pause / Resume	Controls the test in progress, enabling you to pause, stop and resume running of the selected Agenda.
	Throttle Control	Open the Throttle Control dialog box, enabling you to dynamically change the load generated on the fly.
	Open	Open WebFT and display the selected Agenda for editing.
	View JavaScript	Open a window displaying the JavaScript for the Agenda.
	Agenda Options	Opens Agenda Options dialog box enabling you to view the options for the selected Agenda.
Load Machine	View Schedule	Open the Load Machine Schedule dialog box displaying the schedule defined for the Load Machine.
	Stop / Pause / Resume	Control the test in progress, enabling you to pause, stop and resume running of the selected Load Machine.
	Throttle Control	Open the Throttle Control dialog box, enabling you to dynamically change the load generated on the fly (available only when Load Machine Pool is selected).

Note: In edit mode, the items in the pop-up menus are used to edit the test session configuration. See The Session Tree in Edit Mode (on page 84).

The Results Window

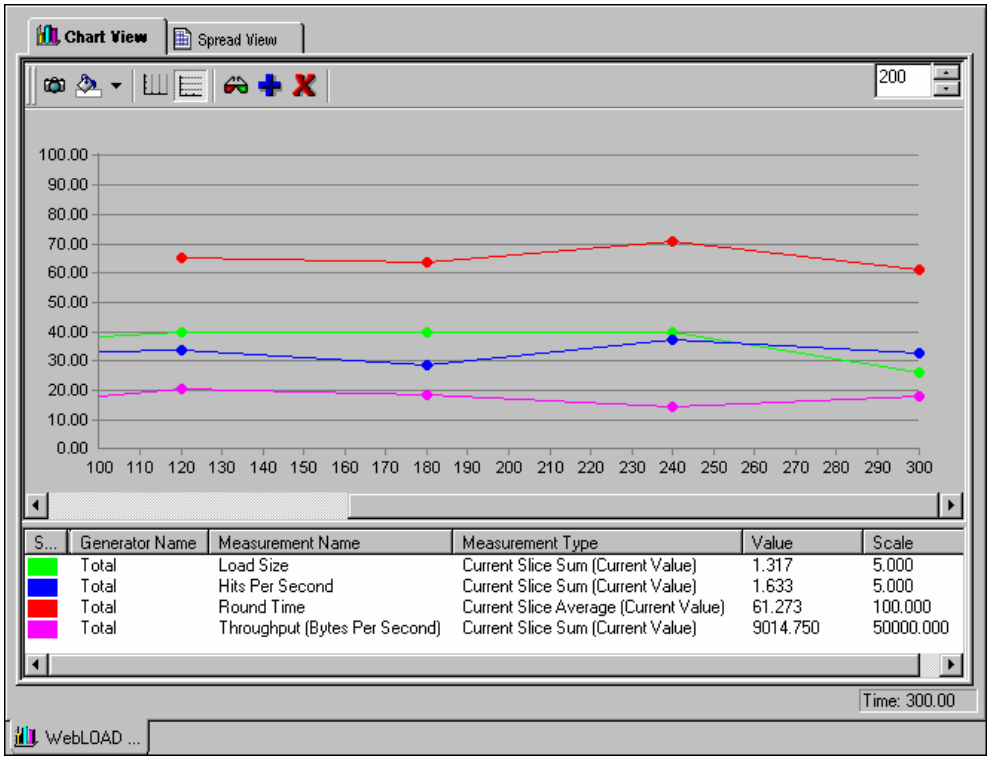
The Results window displays all of the reports opened during a test session. Use the tabs located at the top and the bottom of the window to view different reports. The following reports are available in the Results window.

Report	Description
Integrated Reports	A report that can be configured to combine Performance data from the Load Machines and Probing Clients, Performance data on the ABT, gathered from Performance Manager and existing performance data from previously saved load sessions.

Report	Description
Statistic Reports	A real-time monitor that displays the results of your test while you run it that can be saved and exported.
WebLOAD Dashboard	Provides a summary of the performance statistics generated during the test session.
Data Drilling Reports	Provides a detailed description of all user-defined and named transactions to the instance level, including all hit failures and the reasons for them.
Viewer	Provides a graphic presentation of event failures displaying both the expected and actual events.

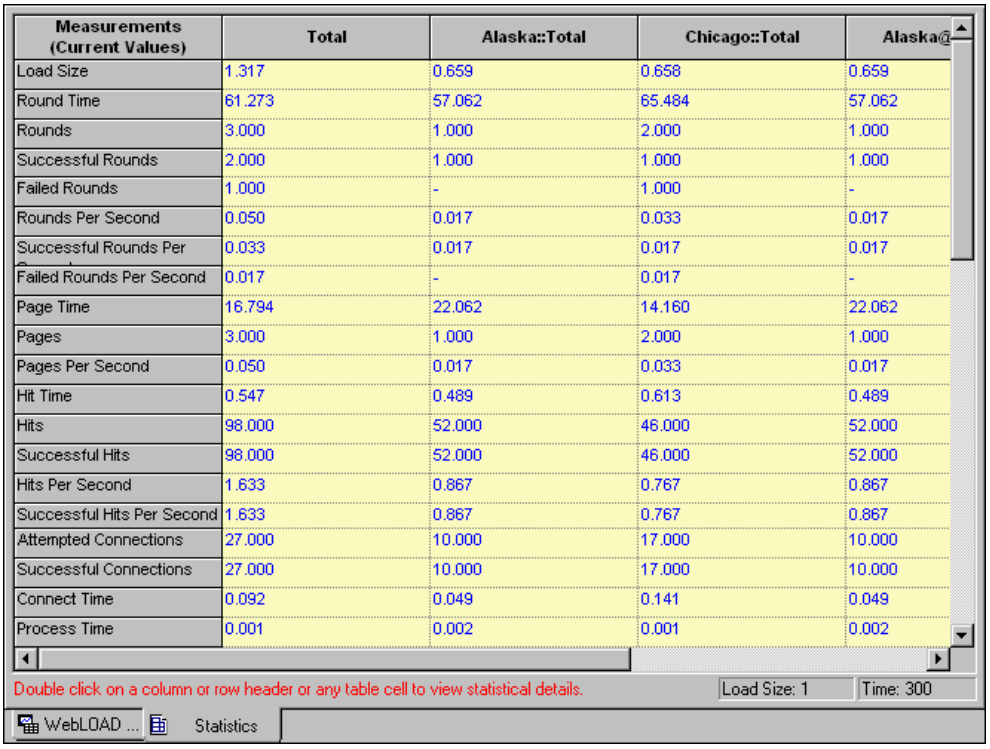
Integrated Report

WebLOAD Integrated Reports provide both a graphical and statistical view of the performance of your application as it is being tested. Integrated Reports can be viewed while the test is in progress or saved for later analysis.



Statistics Report

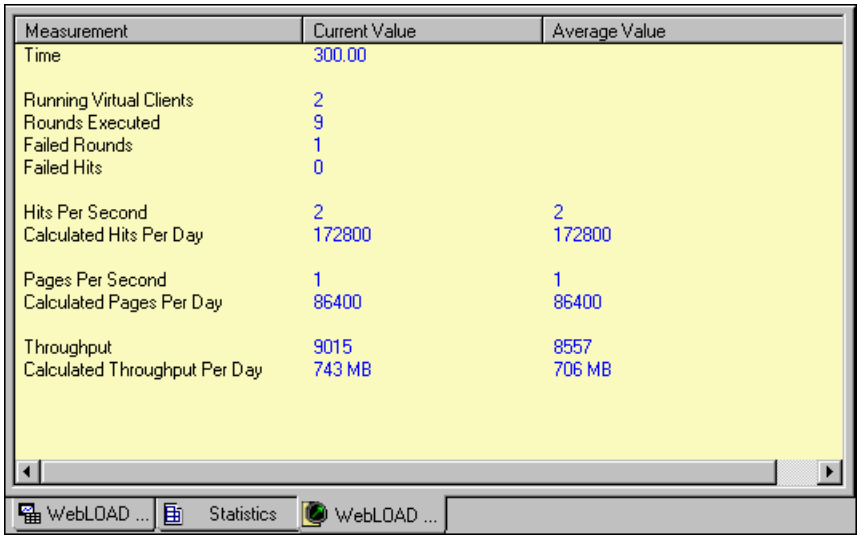
WebLOAD collects approximately 35 different statistics during a test. The Statistics Report displays the values for all of them.



Measurements (Current Values)	Total	Alaska::Total	Chicago::Total	Alaska@
Load Size	1.317	0.659	0.658	0.659
Round Time	61.273	57.062	65.484	57.062
Rounds	3.000	1.000	2.000	1.000
Successful Rounds	2.000	1.000	1.000	1.000
Failed Rounds	1.000	-	1.000	-
Rounds Per Second	0.050	0.017	0.033	0.017
Successful Rounds Per	0.033	0.017	0.017	0.017
Failed Rounds Per Second	0.017	-	0.017	-
Page Time	16.794	22.062	14.160	22.062
Pages	3.000	1.000	2.000	1.000
Pages Per Second	0.050	0.017	0.033	0.017
Hit Time	0.547	0.489	0.613	0.489
Hits	98.000	52.000	46.000	52.000
Successful Hits	98.000	52.000	46.000	52.000
Hits Per Second	1.633	0.867	0.767	0.867
Successful Hits Per Second	1.633	0.867	0.767	0.867
Attempted Connections	27.000	10.000	17.000	10.000
Successful Connections	27.000	10.000	17.000	10.000
Connect Time	0.092	0.049	0.141	0.049
Process Time	0.001	0.002	0.001	0.002

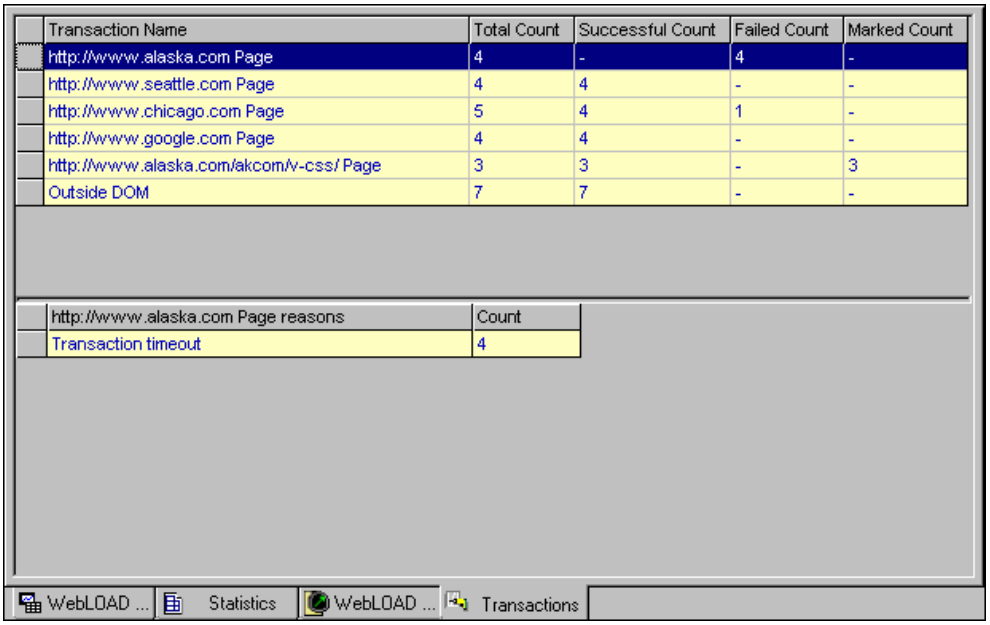
Dashboard

The Dashboard displays real-time statistical information about the test session including the number of Virtual Clients running, hits per day, pages per day and throughput.



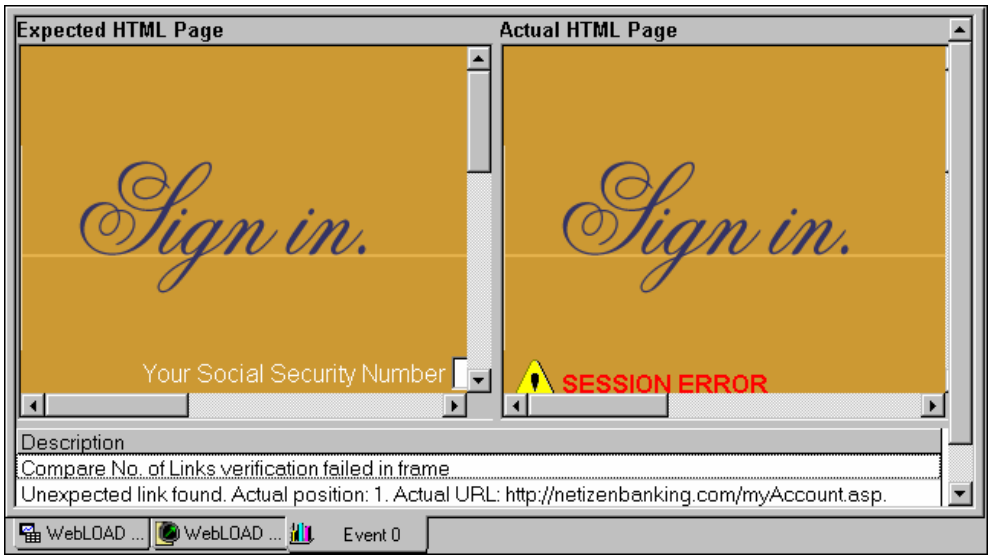
Data Drilling Report

Data Drilling provides both a global and detailed account of hit successes and failures allowing you to verify the functional integrity of your Web application at the per-client, per-transactions and per-instance level. The Data Drilling reports provide an extremely detailed yet easily accessible summary of all the statistical, timing and performance information collected over the course of the test session.



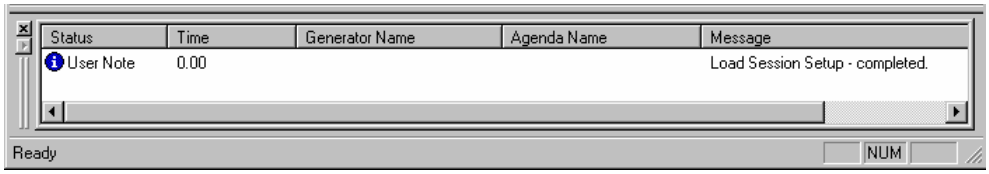
Event Viewer

The Event Viewer displays an illustration of each event triggered during the test.



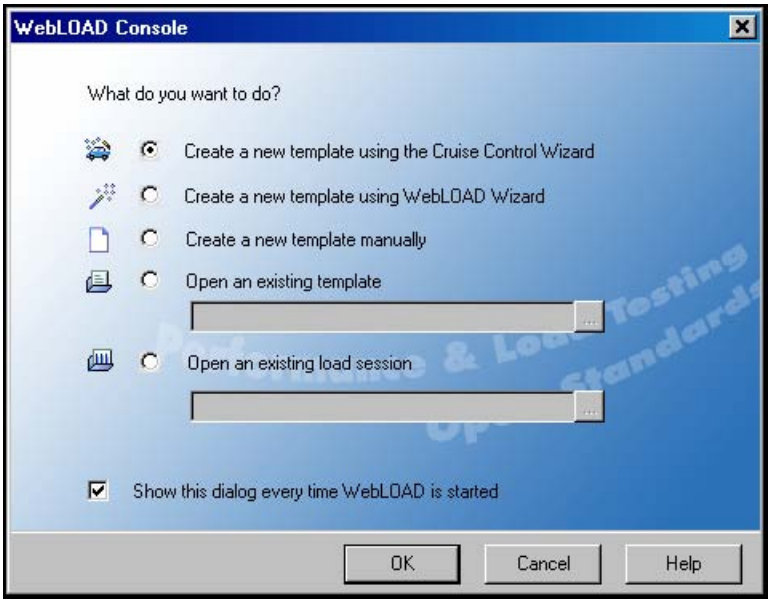
Log Window

The Log Window displays a summary of the test including all log messages detected by WebLOAD in run time that are generated by the Console, the JavaScript compiler and any user messages programmed in the test script.



The WebLOAD Console Shortcut Screen

When you start WebLOAD, the WebLOAD Console shortcut screen displays at the center of the screen, providing you with direct access to the WebLOAD wizards, load templates and previously run load sessions.



Item	Description
Cruise Control Wizard	Opens the Cruise Control Wizard for creating a new load template.
WebLOAD Wizard	Opens the WebLOAD Wizard for creating a new load template.
Manual	Opens a new blank session tree for creating a new template.
Load Template	Opens a saved load template (`*.tpl`). Enter the name of the saved load template in the edit box.
Load Session	Opens a saved load session (`*.ls`) including template data from a completed test. Enter the name of the saved load session in the edit box.

Use either the Cruise Control Wizard to develop a goal-seeking performance test or the WebLOAD Wizard to develop a basic load template. Both of these wizards enable you to save your test parameters as a template that you can later edit manually using the Console toolbar and menus.

After developing a test with either of the wizards, you can save the test parameters as a load template. After running the test you can save the test parameters and results as a load session. You can then edit the test parameters by editing the load template or view the load session test results. For more information on load sessions and load templates, see Managing Load Templates and Load Sessions (on page 125).

Edit Mode and Run Modes of Operation

When you open a new test, WebLOAD is in Edit mode. In edit mode, you can edit the parameters of your test.

When you run your test, WebLOAD switches to Run mode. When in run mode, you cannot edit the parameters of the test.

If you are in Run mode and you wish to edit the test parameters, press the Edit Template toolbar button to switch to edit mode.

The Console in Edit Mode

In edit mode you can edit the session tree parameters. In run mode WebLOAD displays the configuration dialog boxes. Change from run mode to edit mode using the Edit toolbar button.

To edit test session parameters:

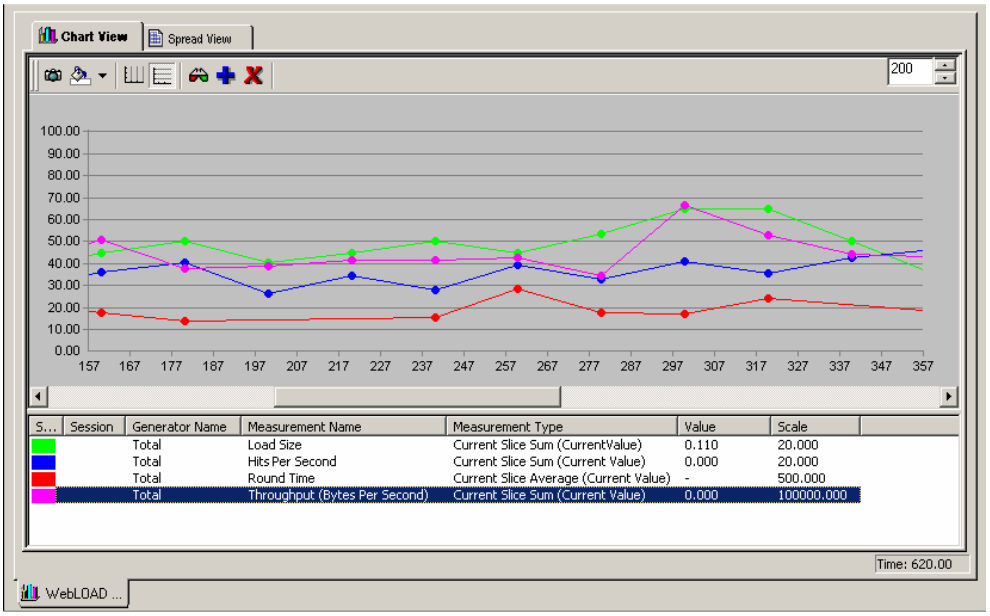
- Place the cursor on a node in the Session Tree and click the right mouse button,

  -Or-

  Double-click on an item in the session tree.

The Console in Run Mode

When you start running your load session, the Results window appears.



The following table describes the segments of the Console in Run mode:

Segment	Function
Session Run Toolbar	Displays only during a test run. It contains buttons for controlling the test session.
Session Tree	Presents a graphic display of your test session.
Results Window	Displays all of the reports opened during a test session. Use the tabs located at the top and the bottom of the window to view different reports.
Log Window	Displays all of the error messages recorded during a test session. You can toggle the Log Window display on/off through the View menu.
Status Bar	Indicates the program status. You can toggle the Status Bar display on/off through the View menu. When WebRM is running, the Standalone Workstation icon or the Connected Workstation icon displays.

The Console Menu Options

The Console menu bar is located at the top of the screen.

This section discusses each of the WebLOAD menus, their options and their related toolbar buttons.

File Menu

The File menu includes the following functions for WebLOAD file management. The Console toolbar has an icon corresponding to menu options.

Menu Entry	Icon	Description
New		Starts a new test session, creating a new WebLOAD load template (*.tpl) file. When you start a new load template an empty session tree is displayed on the screen.
Open Load Session	(none)	Opens a saved load session. A load session is saved as a file of type `*.ls`. The Console displays a session tree window initialized to the settings saved in the file.
Save Load Session, Save As Load Session		Saves the current load session configuration and results to a file of type `*.ls`. The following information is saved: ◆ Selected hosts, both Probing Clients and Load Machines ◆ Assigned Agendas for each host ◆ Test session schedule ◆ Report configuration ◆ Test session results (Performance Report data)
Cruise Control Wizard		Opens the Cruise Control wizard. The Cruise Control wizard enables you to run goal-seeking performance testing.
WebLOAD Wizard		Opens the WebLOAD wizard. This wizard walks you through the WebLOAD configuration process.
Open Load Template		Opens a load template file of the type `*.tpl`. A load template file contains test configuration, the assigned Agendas, the testing schedule, and the report configuration, without the test session results.

Menu Entry	Icon	Description
Save, Save As Load Template	(none)	Saves the current load configuration as a load template file. The following information is saved: ◆ Selected hosts, both Probing Clients and Load Machines ◆ Assigned Agendas for each host ◆ Test session schedule ◆ Report configuration
Edit Template		Transfers from Run mode to Edit mode.
Print		Prints the report displayed with all contents and formats.
WebFT		Opens WebFT.
WebLOAD REPORTER		Opens WebLOAD REPORTER.
Create Mix	(none)	Opens the Mix Creation dialog box, enabling you to create an Agenda Mix.
Additional Information	(none)	Displays additional information for the active template / session.
Recent Files	(none)	Lists the template / session files you've most recently opened in the WebLOAD. To quickly reopen one of these files, click it.
Exit	(none)	Closes the Console.

Session Control Menu

This menu is used to control load sessions. The menu items change from Edit to Run mode. The menu items in the upper section of this menu enable editing the test configuration dialog boxes. These menu items vary, when in Run mode, to enable only viewing the dialog boxes.

The Session Control menu includes the following functions for load session management. The Console toolbar has an icon corresponding to each menu option.

Menu Entry	Icon	Description
Modify Hosts Selection		Opens the Host Selection dialog box. In Run mode, you can view the contents of the Host Selection dialog box. In Edit mode, you can define the hosts you wish to designate as Probing Client Machines or Load Machines.
Open Session Tree		Opens a graphic representation of the current load session configuration. Double-clicking on components before the test session begins lets you modify the selected component's configuration.

Menu Entry	Icon	Description
Modify Agenda Selection	(none)	Opens the Agenda / Mix Selection dialog box. In Run mode you can view the selected Agenda and Mix files. In Edit mode you can add or delete Agenda and Mix files to the session.
Performance Statistics Manager	(none)	Opens the Performance Statistics Manager dialog box for configuring the statistics parameters to be monitored.
Modify Schedule	(none)	Opens the Modify Schedule dialog box that corresponds to the currently selected host or Agenda. In Run mode you can view the contents of the Modify Schedule dialog box. In Edit mode you can change the schedule.
Throttle Control		Opens the Throttle Control dialog box. Use Throttle Control to dynamically change the load while the session is in progress.
Delete		Deletes the currently selected load session component. A component may be an Agenda, or a host. A message box pops up, asking for confirmation.
Start Session		Starts executing the current load session.
Stop All		Stops all load session activities. A message box pops up, asking for confirmation.
Stop	(none)	Stops the currently selected load session components. If the root is currently selected, all components are stopped. Selecting stop during a Cruise Control test session stops all components.
Pause		Temporarily stop execution of the load session component. If the root is selected, all components are paused
Resume		Resumes execution of a paused load session component. Note: Pausing and continuing does not automatically reschedule load session Agendas. Scheduling continues in relation to the start of the session.
Update floating license resources		Open the Update Resources dialog box.

Reports Menu

The Reports menu provides options for creating real-time and summary reports showing the performance data gathered by WebLOAD. The following is a brief description of each report. A full description of all WebLOAD reports can be found in Running a Load Session (on page 275).

Category	Report	Icon	Description
Integrated Reports	New Report		Define a new report. Configure the measurements that are displayed in the integrated report.
	Delete Report		Delete an existing report from the reports list.
	Open Report		Open an existing report from the reports list.
Statistics Report	Statistics Report		Displays real-time updates of the statistics measurements for Load Machines, Probing Clients and the performance measurements.
Dashboard	Real-time dashboard		Displays real-time updates of statistic measurements in a dashboard like display.
Data Drilling	Summary Grid		Open the Transaction Summary Grid and display information on all user-defined and named transactions.
Export	Export to Excel		Export the chart or grid displayed to Microsoft Excel.
	Export to Tab File		Export the chart or grid displayed to a Tab file.
	Export to HTML		Export the chart or grid displayed to HTML.
Export All	Export to Excel		Export all statistics to Microsoft Excel.
	Export to Tab File		Export all statistics to a Tab file.
	Export to HTML		Export all statistics to HTML.
Create Summary Report	Summary Report		Open the Summary Report Wizard.

View Menu

Menu Entry	Icon	Description
Toolbar	(none)	Opens the Toolbars window. Select toolbars to be displayed on the Console toolbar.
Gauges	(none)	Toggle to display/not display the graphic gauges.

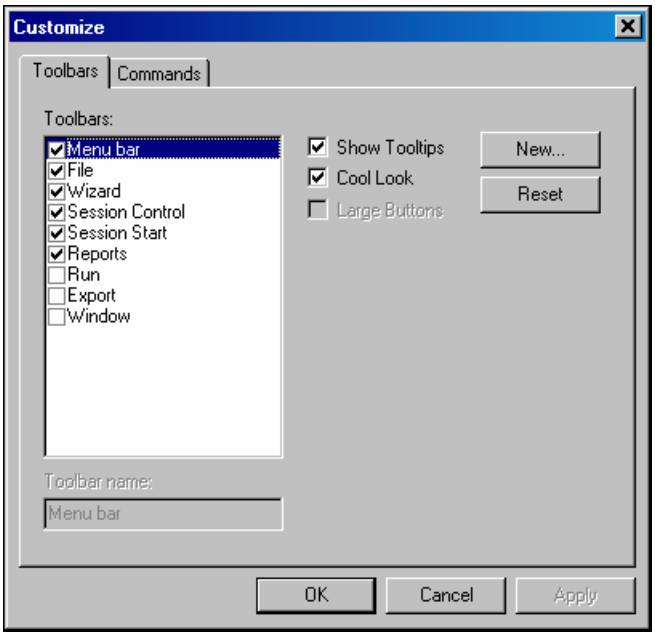
Menu Entry	Icon	Description
Toolbar	(none)	Opens the Toolbars window. Select toolbars to be displayed on the Console toolbar.
Session Tree Window		Toggle to display/not display the session tree pane.
InfoBar	(none)	Toggle to display/not display the Infobar.
Log Window		Toggle to display/not display the Log window. The log window is both floating and dockable, and can be placed in any part of the screen.
Clear Log Window	(none)	Clear the Log window.
Workbook	(none)	Toggle to display/not display the report tabs at the bottom of the Results window.
Customize	(none)	Open the Customize dialog box.

The Customize option contains two tab dialog boxes:

Tab	Description
Toolbars	Configure the toolbars displayed.
Commands	Customize toolbar buttons and menu commands.

Toolbars Dialog Box

The Console toolbar has several topic-specific segments, separated by a set of bars. The Toolbars dialog box lists the various toolbar segments. To display a toolbar in the Console, select the checkbox adjacent to the toolbar name. To hide a toolbar, clear the checkbox.

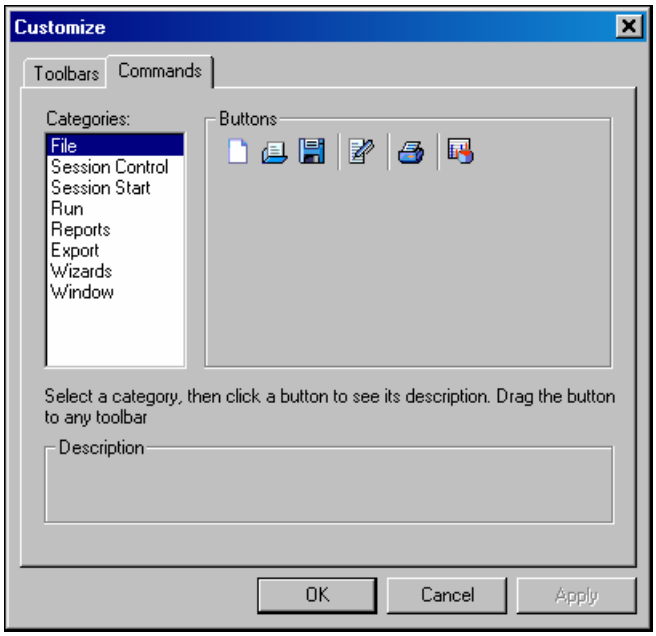


Example: You can remove all the file options from the toolbar by disabling the file entry in the toolbar dialog box. The file-specific toolbar segment is not displayed.

The first five toolbar segments are checked by default. The Export toolbar is not displayed by default. When using the WebLOAD export features, check this option to use the convenient Export toolbar buttons.

Commands Dialog Box

Use the Commands dialog box to configure the buttons on each toolbar. In this dialog box, you can remove toolbar buttons or move them to a different toolbar segment.



Toolbar buttons are moved or removed by dragging and dropping them where desired. To remove a toolbar button drag and drop it from the Console toolbar to this dialog box.

Tools Menu

Menu Entry	Icon	Description
Global Options		Opens the Global Options dialog box. Use the global options to customize WebLOAD to your specifications, see Setting Global Options (on page 203).
Default Options		Opens the Default Options dialog box. Use the default settings to customize the Agenda settings to your specifications. When building a new template or session, or editing an existing one, each new single Agenda or Agenda participating in a Mix is automatically assigned these defaults, see Setting Agenda Options (on page 221).
Current Session Options	(none)	Opens the Current Session Options dialog box. Use the session settings to customize the current session to your specifications. This option will update the options for all Agendas in the current session, see Setting Agenda Options (on page 221).

Menu Entry	Icon	Description
Agenda Options		Opens the Agenda Options dialog box. Use the Session settings to customize the current Agenda to your specifications, see Setting Agenda Options (on page 221).

Window Menu

Use the Window menu to configure the look of the Results window in the Console.

Help Menu

Use the Console Help menu to access:

- Online help
- License information
- WebLOAD Wizard
- Copyright and version information about the Console

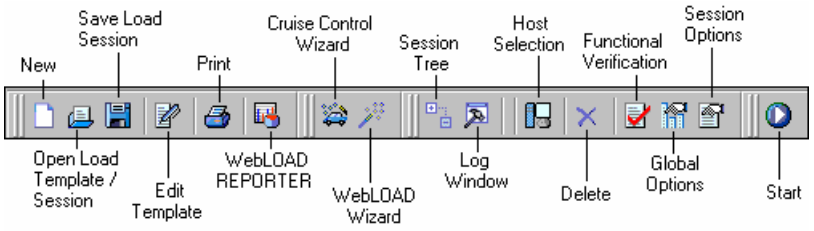
The Console Toolbar

The Console toolbar is located below the menu bar. The standard toolbar contains four types of options, arranged as follows:

Options	Description
File Management Options	Access to standard Windows file management commands. These options are: ◆ New ◆ Open ◆ Save ◆ Edit ◆ Print
WebLOAD Wizards	Access the Cruise Control Wizard and the WebLOAD Wizard.
Session Control Options	Commands used to call up a WebLOAD configuration dialog box. These commands are used to manually configure a test, or edit a load template.
Run Commands	While the test session is in progress, you can pause the test, stop the test, restart the test and dynamically change the load size using these toolbar buttons.

Toolbar Buttons

The following figure details each button on the Console toolbar. (Some items, such as Throttle Control, Stop, Pause and Go display when you start a test or select them from Tools | Customize.)

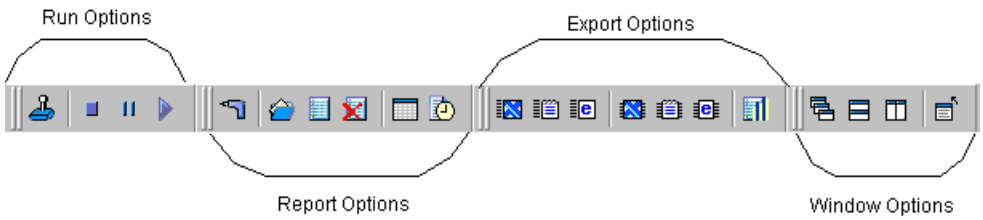


Button	Description
New	Starts a new load template. A new empty session tree displays.
Open Load Template	Opens an existing load template.
Save Load Session	Saves a load session. This includes the load template and the data accumulated running the test.
Edit Template	Transfers from Run mode to Edit mode.
Print	Prints the active chart or grid displayed in the Results window, with all contents and formats.

Button	Description
WebFT	Opens WebFT.
WebLOAD REPORTER	Opens WebLOAD REPORTER with the current load session (if available) loaded by default.
Cruise Control Wizard	Opens the Cruise Control wizard.
WebLOAD Wizard	Opens the WebLOAD wizard.
Session Tree	Opens the session tree. The tree is a graphical representation of the current load session configuration. The various components of the test can be edited by either double-clicking or using the right mouse button on various branches of the tree. It is recommended that you view the session tree as you work.
Log Window	Toggles the log window on and off.
Host Selection	Opens the Host Selection dialog box for selecting the hosts, and defining the roles of the hosts participating in the test.
Delete	Deletes a component from the session tree.
Functional Verification	Opens the Functional Testing tab for configuring your test to perform functional verification.
Global Options	Opens the Global Options dialog box.
Session Options	Opens the Session Options dialog box.
Start	Starts a test session. WebLOAD verifies the configuration and the test program and then run the test.

Extended Toolbar Segments

The following toolbar buttons are optional and can be added to the toolbar through the Toolbars dialog box:

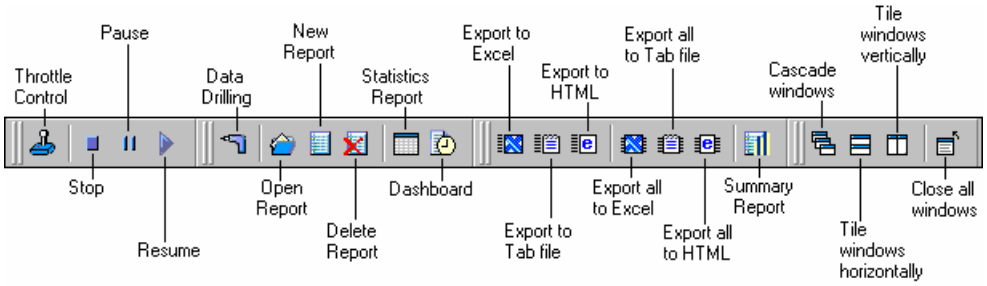


Options	Description
Run Options	Commands used to control the session in progress.

Options	Description
Reports Options	Commands used to open, delete and create new reports.
Export Options	Commands used to export Integrated and Statistic reports to Excel, HTML and tab files, and to create and export a summary report.
Window Options	Commands used to define multiple window presentations. Windows can be displayed tiled, cascaded, or adjacent to one another. This toolbar can also be used to close windows.

Extended Toolbar Buttons

The following illustration details each button on the Console optional toolbars:



Button	Description
Throttle Control	Opens the Throttle Control dialog box.
Stop All	Stops a test that is running.
Pause	Pauses a test that is running.
Resume	Restarts running a paused test.
Data Drilling	Opens the Transaction grid.
Open Report	Opens an existing WebLOAD report.
New Report	Creates a new Integrated report.
Delete Report	Deletes an existing report.
Open Statistics	Opens the Statistics report.

Button	Description
Dashboard	Displays statistical information about the test session running including: ◆ Number of Virtual Clients running ◆ Number of failures ◆ Hits per day ◆ Pages per day ◆ Throughput per day
Export to Excel	Exports a WebLOAD report to Excel.
Export to Tab File	Exports a WebLOAD report to a tab-delimited file.
Export to HTML	Exports a WebLOAD report to HTML.
Export All to Excel	Exports all WebLOAD statistics to Excel.
Export All to Tab File	Exports all WebLOAD statistics to a Tab file.
Export All to HTML	Exports all WebLOAD statistics to HTML.
Summary Report	Creates and exports a summary report.
Cascade Windows	Cascades the windows displayed in the WebLOAD Results window.
Tile Horizontally	Tiles the windows displayed in the WebLOAD Results window horizontally.
Tile Vertically	Tiles the windows displayed in the WebLOAD Results window vertically.
Close Window	Closes the windows displayed in the WebLOAD Results window.

C H A P T E R 7

Creating Load Templates

Using WebLOAD, you create load templates to define the events to occur in the load session. This part describes how to quickly set up a load template. Load templates define the participants (Clients and Servers) in the load session, the Agendas to run, the number of Virtual Clients to emulate and the test schedule.

In This Chapter

WebLOAD Options for Creating Load Templates: 111

WebLOAD Options for Creating Load Templates:

WebLOAD provides three options for creating load templates:

- Using the WebLOAD Wizard to build a basic load template. You define the Agendas to run, the Load Machines on which to run the Virtual Clients and the test schedule. After creating a basic template with the wizard, you can change aspects of the configuration through the Console toolbar and menus.
- Using the Cruise Control Wizard to create goal-seeking performance tests. Rather than running multiple tests to measure the performance at different loads, you can run Cruise Control to measure the performance goal that you desire. You define the Agendas to run and the goals to achieve and WebLOAD automatically schedules the test session and increases the number of Virtual Clients running until your goals are achieved.
- Manually using the Console toolbars and menus.

The load template material is divided into five chapters:

- Planning a Test Session (on page 113) describes the stages involved in planning and executing a test session.
- Connecting to WebRM (on page 121) describes the process of connecting to WebRM and requesting testing resources.
- Managing Load Templates and Load Sessions (on page 125) provides information about working with Load Templates and Load Sessions.
- Creating Load Templates with the WebLOAD Wizard (on page 133) guides you through a basic template configuration using the WebLOAD Wizard.
- Creating Load Templates with Cruise Control *(on page* 161*)* teaches you to configure a goal-seeking performance test.
- Configuring a Load Template Manually (on page 189) provides instructions for configuring a test template manually.

After completing these chapters, you will have the basic skills necessary to create and run your own test.

C H A P T E R 8

Planning a Test Session

Planning your test is imperative for developing tests that accomplish your load testing objectives. Planning your test helps you:

- Plan the resources required for testing.
- Create templates that accurately emulate your users typical working environment.
- Define test success criteria.

This chapter describes the stages involved in planning and executing a test session. It shows you where to find details in this guide about each concept so you can quickly find information of particular interest to your needs.

In This Chapter

Load Session Workflow 113
Forming a Test Plan 115
Starting WebLOAD 117

Load Session Workflow

A load session is performed in stages. This chapter explains the planning stage in detail. The rest of the stages are detailed in subsequent chapters.

- Planning a Load Session

  Planning a load session involves considering:

- What you want to test - what application are you going to test?
- What functionality do you want to test - what actions will the users perform?
- How many Virtual Clients you want to simulate?
- How long your test will run?
- What are acceptable results? Acceptable results are defined by your test objective. For example, you can verify:
  - Acceptable user response times
  - Reliability by running stress tests
  - Performance degradation after updates
- What resources are required for performing the test?

♦ Creating Visual Test Agendas

Use WebFT to record the activity you want to test in a Web browser and save it as an Agenda. If you plan to test users performing different activities, you need to record separate visual Agendas for each activity. See the *WebFT User's Guide.*

♦ Creating Load Templates

Creating load templates involves defining the participants (Clients and Servers) in the load session. The definition includes selecting the Agendas to run, the host computers and number of Virtual Clients participating in the load session, and scheduling the test. After you create a template it is saved in a *.tpl file. Use the commands in the Session Control menu to create, open and save template files.

WebLOAD provides three ways for creating load templates:

- Using the WebLOAD Wizard to build a basic load template. You define the Agendas to run, the Load Machines on which to generate the Virtual Clients and the test schedule. After creating a basic template with the wizard, you can change aspects of the configuration through the Console toolbar and menus. For information on creating a template with the WebLOAD Wizard, see Creating Load Templates with the WebLOAD Wizard (on page 133).
- Using the Cruise Control Wizard to create goal-seeking performance tests. Rather than running multiple tests to measure the performance at different loads, you can run Cruise Control to measure the performance goal that you desire. You define the Agendas to run and the goals to achieve, and WebLOAD automatically schedules a test and increases the load until your goals are achieved. For information on creating a template with the Cruise Control Wizard, see Creating Load Templates with Cruise Control (on page 161).
- Manually using the Console toolbars and menus. See Configuring a Load Template Manually (on page 189) for information on defining a template manually.

♦ Running a Load Template

After you create the load template, you run it. As your test runs, you receive various reports and messages from the various participating hosts. The activity in your test

session is displayed on the Console screen. This screen is updated continuously in real time as you work. See Running a Load Session (on page 275).

- Analyzing the Test Results

  WebLOAD displays the results of your load session in tabular format and in easy to read graphs.

Forming a Test Plan

Every test must begin with a test plan. When you plan a test session, the first step is to analyze your application to ensure that your test will accurately reflect the working environment of your users. You want to consider the goals of the test and your available resources. Consider the following:

1. What application are you testing?

   When planning a test, the first thing you should consider is the target of your test. You must define what Web application you plan to test. Your test target is your ABT (Application Being Tested).

   Before planning further, call up a browser and access the site to be tested, to make sure that the connection to the site on the Web is valid.

2. What criteria are you testing?

   You must know what functionality of the Web application you want to test. Usually you want to test the amount of time required to perform a certain function on the Web site. You may want to test the time it takes to access a page, the time it takes to submit a form, or the number of clients that can simultaneously access your site.

   Using Cruise Control, you can test the number of clients that can perform a function, such as accessing a page or submitting a form, at the performance level that you have specified.

   When defining the criteria for the test you should consider:

   - How many users do you want to simulate? How many users are anticipated to connect to your application.
   - How many different user activities do you want to simulate? If you plan to test users performing different activities, you need to create separate Agendas for each activity.
   - What type of users do you want to simulate? First time users respond differently to an application than returning users. Using the playback sleep time settings you can set WebLOAD Virtual Clients to run with the recorded sleep time delays, remove all delays to stress the application, or create random delays simulating different users.
   - What type of response time is acceptable? Using Cruise Control, you can create test sessions that increase the load and notify you at what load the response time exceeds acceptable levels.
   - What is an acceptable failure rate? Should your test be stopped if a set number or percentage of errors is surpassed?

- What type of connections speeds do you want to simulate? Using the connection speed settings available with WebLOAD you can simulate users connecting through different connections.

3. What types of systems are testing these aspects?

   Your test can include Load Machines and Probing Clients. A Load Machine bombards the ABT with a load of Virtual Clients. Load Machine testing gives you an indication of the way your program behaves under the stress of a heavy load.

   Probing Client testing is usually performed simultaneously with Load Machine testing. Using a Probing Client, you can test the system performance with other functionality, while the system is undergoing stress testing. For example, you can measure the time required to submit a form, while the ABT is being bombarded by the Load Machine simulating multiple clients calling up a page.

   WebLOAD generates exact values for Probing Client performance as opposed to averages for Load Machine performance.

Planning a Test Program

The test you devise must be coded in an Agenda file. Agendas define the actions WebLOAD Virtual Clients perform. At this stage, you must plan the test program that you will develop. Your Agendas should define the typical tasks you expect users to perform with your Web application.

For example, for a book store application you could create Agendas that perform typical shopping activities such as searching for a book, adding books to the shopping cart and checking out. You should also consider your test objectives and define transactions to measure the response times for typical user activities such as searching for a book to ensure your Web application meets your requirements. You can also plan synchronization points to emulate peek server loads and ensure your requirements are met even under heavy load. Remember that if your plan indicates that your Load Machine and Probing Client perform different tasks, you need an Agenda for each host.

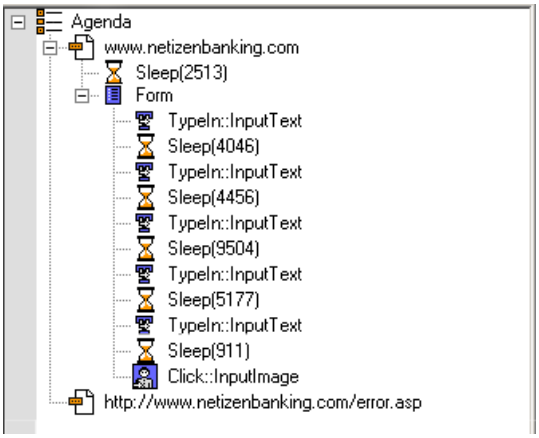
WebLOAD Agendas are created in WebFT. Refer to the *WebFT User's Guide*, the *TestView Programmer's Guide*, and the *TestView JavaScript Reference Manual* for further information on creating Agendas.

Creating the Test Agendas

The simplest way is to create an Agenda is using WebFT that enables you to create a visual Agenda by recording your activity in the browser.

For example, if you want to test the speed at which a certain Web application displays a page, and the speed required to submit a form, you can open WebFT and go to that application through a browser. With WebFT recording your activity, you can display the page desired and submit the form desired. WebFT saves your browser activity as a visual Agenda. After recording

the basic activity, you can add transaction statements to measure the time it takes to display the page.



You can tailor the recorded Agenda, by adding commands and timers available in WebFT. For more information on creating an Agenda with WebFT, see the *WebFT User's Guide*.

Examining Your Test Resources

After you know what you are going to test, you must examine the resources that are available to you for running your test. This helps you allocate the resources for your test load. If the computers available to you cannot support the desired load, consider distributing the load over several computers.

Starting WebLOAD

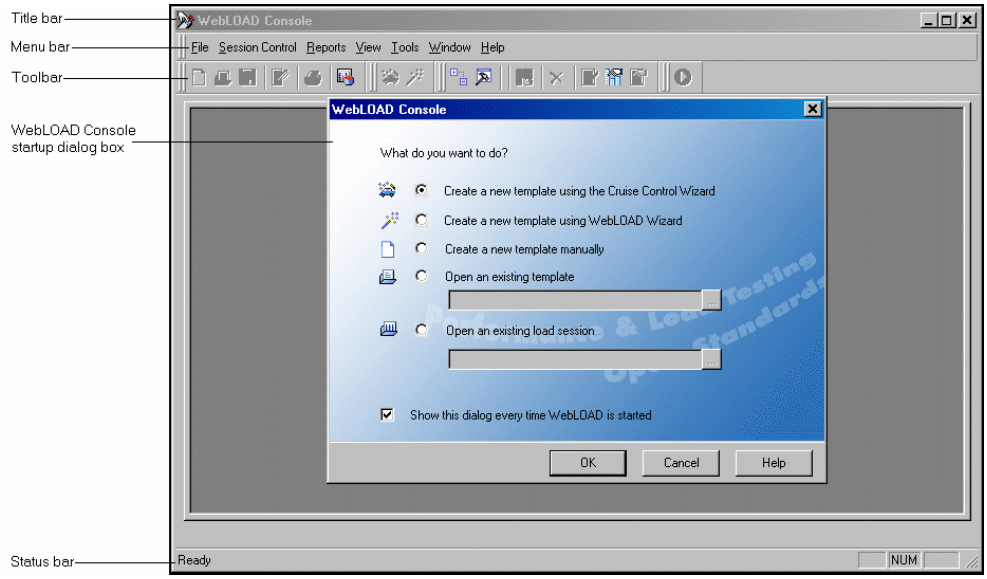
When your test plan is complete and your Agendas are ready, you can open WebLOAD and begin test configuration process.

To open WebLOAD:

1. Verify that all the participating Host computers are properly connected. See Pinging a Host (on page 67).
2. Select Start | All Programs | TestView | WebLOAD Console.

   The Console calls up TestTalk automatically, on its system.

The Console opens with the Console shortcut screen displaying a variety of shortcuts to the various WebLOAD components.



To create a new WebLOAD template, select one of the three methods available on the shortcut screen.

- Manual - create the load template by yourself, defining the Agendas to run, the number of Virtual Clients to run, the Load Machines on which to generate the load and build a schedule for running the test.
- WebLOAD Wizard - the wizard helps you through the process of configuring the load template.
- Cruise Control wizard - you define the Agendas to run, the Load Machines to run the Virtual Clients on and the test goals to achieve and WebLOAD automatically builds a schedule based on your goals.

The easiest way to configure a basic test is through the WebLOAD Wizard. Save your work in the Wizard as a load template. After you have closed the WebLOAD Wizard, you can use the Console menus to add functionality to your load template (*.tpl file).

Creating Load Templates with the WebLOAD Wizard (on page 133) discusses the WebLOAD Wizard. Setting Global Options (on page 203) provides instructions for using the Console features to add functionality to the load template created in the WebLOAD Wizard. Creating Load Templates with Cruise Control (on page 161) provides instructions for configuring a test session running the Cruise Control feature. Connecting to WebRM (on page 121) provides instructions for connecting the Console to WebRM using the WebRM Wizard. The Console can then request resources from WebRM so it can work in Connected Workstation mode.

3. Click the TestTalk icon on each host computer participating in the test. (TestTalk is automatically enabled only on the host running the Console.)

   If you are using a Solaris or Linux system as a Load Machine, verify that the TestTalk software is running on the Solaris or Linux system. If you have closed TestTalk after the installation, run TestTalk again.

   All of the systems participating in the test session, including the system running the Console, the Load Machines, and the Probing Clients must be running the network agent, TestTalk. This establishes communication between participating clients and servers. The ABT does not require TestTalk or any other WebLOAD-specific software.

Note: Do not close TestTalk in the middle of a test session. This can cause unexpected results. When shutting down the system, be sure to shut down the Console first. Then shut down TestTalk.

C H A P T E R 9

Connecting to WebRM

WebRM addresses corporate functional and performance testing efforts by organizing, managing and directing WebLOAD resources (Virtual Clients, Probing Clients and Connected Workstations) for powerful collaborative testing and analysis, improved communications and faster issue resolution.

WebRM enables multiple users involved in various stages of application development and testing to share testing resources. By distributing WebLOAD testing resources each developer can run a test session to simulate, validate, and pinpoint where performance problems occur at any stage of the development life cycle, thus eliminating design flaws and ensuring product quality. Using WebRM, WebLOAD resources can be used optimally to fulfill an organizations goals and priorities.

WebRM can run on a separate machine from the other WebLOAD Consoles/Load Machines or can run on a machine that runs WebLOAD Consoles/Load Machines.

In This Chapter

Getting Started 121
Working with WebRM 123

Getting Started

WebRM addresses corporate functional and performance testing efforts by managing WebLOAD Virtual Clients, Probing Clients, and Connected Workstations for powerful collaborative testing and analysis, improved communications and faster issue resolution.

WebRM enables multiple users involved in various stages of application development and testing to share testing resources. By distributing WebLOAD testing resources each developer can run a test session to simulate, validate, and pinpoint where performance problems occur at any stage of the development life cycle, thus eliminating design flaws and ensuring product quality. Using WebRM, WebLOAD resources can be used optimally to fulfill an organizations goals and priorities.

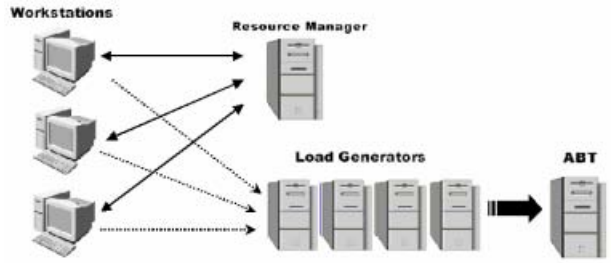
WebRM can either run on a separate machine from the other WebLOAD Consoles/Load Machines or it can run on a machine that is also running WebLOAD Consoles/Load Machines.

About WebRM

WebRM enables developers and QA teams to perform functional and performance testing throughout the development cycle, to ensure that applications respond quickly in the way they were designed to work.

Using WebRM, WebLOAD testing resources (Virtual Clients and Probing Clients) can be distributed and circulated amongst users on a “need to use” basis. WebRM is packaged with a maximum number of Virtual Clients, Probing Clients and Connected Workstation ports, as defined by the WebLOAD package.

The following diagram illustrates a typical WebRM configuration:



WebRM controls the following WebLOAD resources:

- The number of concurrent Connected Workstations (connected ports).
- The number of Virtual Clients running on all Connected Workstations simultaneously.
- The number of Probing Clients running on all Connected Workstations simultaneously.

How Does WebRM Work

When you install WebRM, you apply your license file. The license file contains information regarding the total number of Virtual Clients, Probing Clients, and Concurrent Connected Workstations for which you are licensed.

When installing your WebLOAD workstation, select the location where you installed your WebRM server. WebLOAD will communicate with your WebRM server in order to request and grant testing resources for you.

When you open your WebLOAD console you are prompted to request the number of Virtual Clients and Probing Clients you desire. These resources are granted to you as long as your WebLOAD Console is open, allowing you to complete your test session utilizing the allocated number of Virtual and Probing Clients.

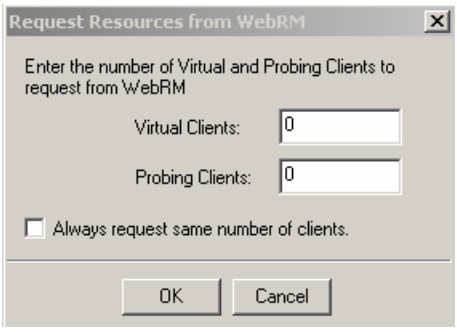
Working with WebRM

WebRM is very straightforward and easy to use. Much of your interaction with WebRM is behind the scenes. Once you have selected the Resources you wish to use, you can configure WebRM to automatically grant you these same resources each time you open your WebLOAD Console.

Requesting Resources from WebRM

Following WebLOAD Console Workstation installation, configure your workstation connection settings by browsing to the machine where the WebRM Server resides. (This is described in WebRM Installation.) When opening your WebLOAD Console, the connection to WebRM occurs automatically.

1. Open your Console by going to Start | Programs | TestView | WebLOAD Console.
2. The Request Resources dialog appears:



3. Enter the number of Virtual Clients and Probing Clients you will need for your test. If you request more than the number of resources allowed for your license, or more than are currently available, you will be prompted with a dialog stating that the resources you requested are unavailable. This dialog will also list what resources are currently available so that you can resubmit your request.
4. If you know that you will require the same number of resources each time you connect to WebRM, you can enable the checkbox to Always request same number of clients.
5. Click OK.

Updating the Number of Requested Resources

You can easily change the number of resources granted to you.

1. In the WebLOAD Console, go to Session Control | Update Floating License Resources.
2. The Request Resources dialog appears, displaying the resources currently granted to you.
3. Edit the Virtual Client and Probing Client settings to reflect the new total number of Virtual and Probing clients you need.

C H A P T E R 1 0

Managing Load Templates and Load Sessions

Load templates define the events that occur in the test including:

- The Agendas to run
- The hosts participating in a test session
- The test schedule
- The number of Virtual Clients to run

Load templates save you the time of reconfiguring the test when you are repeating tests. Load sessions include the test definitions in the load template along with test results.

In This Chapter

Managing Load Templates 125
Managing Load Sessions 129

Managing Load Templates

Load templates define the events that occur in the test. You create test definitions manually or using one of the wizards (WebLOAD or Cruise Control) available with WebLOAD. After creating a test definition, WebLOAD prompts you to save the test configuration as load template. Load templates contain complete test definitions including:

- Agendas
- Information on participating hosts
- Host schedules
- Virtual Clients to run
- Run time configuration options
- Report configurations

Load templates include all test information except for the test results. Load template files are saved as `*.tpl` files. Load templates can be edited using the Console toolbar and menus. You can then save the load template for future test iterations. For example, if you build a test in the WebLOAD Wizard you can call it up in the Console and edit the load template manually, changing perhaps the hosts, the number of Virtual Clients and even the Agenda. Use the commands in the File and Session Control menus to create, open, save and close template files.

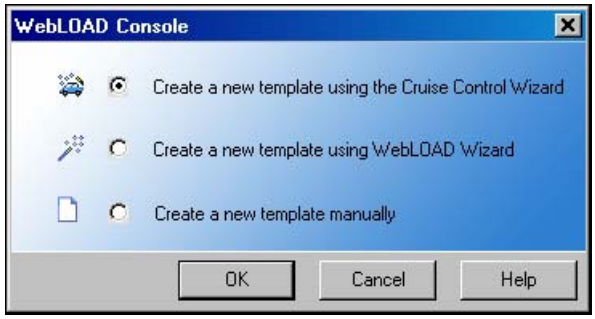
Creating a New Load Template

When you create a new load template, you define the events to occur in the test. A new blank session tree displays to begin configuring your test.

To create a new Load Template:

1. Select File | New.

   The Console shortcut screen displays.



2. Select a method for creating your template and click OK.

   Follow the directions for creating a template using the selected method. For information on creating templates with the WebLOAD Wizard, see Creating Load Templates with the WebLOAD Wizard (on page 133). For information on creating templates with the Cruise Control Wizard, see Creating Load Templates with Cruise Control (on page 161).

Saving a Load Template

To save a Load Template:

1. Select File | Save Load Template,

   -Or-

   Select File | Save As Load Template.
2. In the File Name box, type a name for the load template.
3. Click Save.

   The file is saved with the extension `*.tpl` to the specified location and the Save dialog box closes.

Opening a Saved Load Template

To open a saved Load Template:

1. Click the Open Load Template toolbar button,

   -Or-

   Select File | Open Load Template and browse your directory structure for the location of the saved load template (*.tpl)file,

   -Or-

   Select the Open an Existing Template button from the WebLOAD Console dialog box at system start-up and browse your directory structure for the location of the saved load template `(*.tpl)`file.
2. Double-click on the template you want to open.

   The file opens and the test configuration displays in the session tree.

Saving Additional Information with the Load Template

The Additional Information dialog box provides details about the load template that help identify it. For example:

- Descriptive title
- Author name
- Subject of the test
- Application being tested
- Other important information about the load template

Use the Additional Information dialog box to display information about the load template.

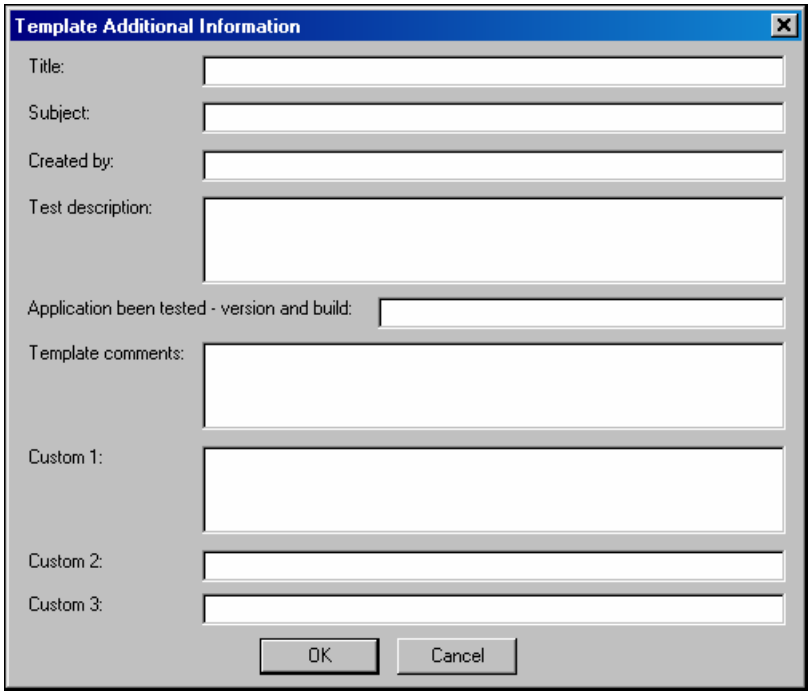
Setting Additional Information Properties

You can set additional information properties for the active Load Template.

To set additional information properties:

1. Select File | Additional information.

   The Template Additional Information dialog box opens.



2. Complete the fields to save additional information, useful for later reference, with the load template.

   The following fields are available:

Field	Description
Title	Provides a space for you to type a title for this load template. The title can be different then the template file name.
Subject	Provides a space for you to type a description of the subject of the load template. Use this property to group similar load templates together.
Created by	Provides a space for you to type the name of the person who authored this load template.
Test Description	Provides a space for you to type a description of the test objectives and what the load template is designed to test.
Application being tested - version and build	Provides a space for you to type the name, version and build number of the application being tested.
Template comments	Provides a space for you to type any comments regarding the load template.
Custom	Provides a space for you to type any comments you want saved with this load template.

3. Click OK.

Managing Load Sessions

A Load session consists of the load template and the data accumulated during the test run. Load sessions cannot be edited.

A saved load session includes:

- The test definition including configuration information on Agendas, participating hosts and their schedules - as illustrated in the session tree.
- The report configuration.
- All test session results.

Load sessions are saved in `*.ls` (load session) files. Load session results are saved in a `*.dat` (Data) file. Saved load sessions are useful for viewing results of previously run test sessions.

Saving a Load Session

To save a Load Session:

1. Select File | Save Load Session,

   -Or-

   Select File | Save As Load Session,

-Or-

Click the Save toolbar button.

2. Browse to the location to save the file and in the File Name dialog box, type a name for the load session.
3. Click Save.

   The file is saved with the extension `*.ls`.

Saving Additional Information for the Load Session

The Additional Information dialog box provides details about the load session that help identify it. For example:

- Descriptive title
- Author name
- Subject of the test
- Application being tested
- Other important information about the load session

Use the Additional Information dialog box to display information about the load session.

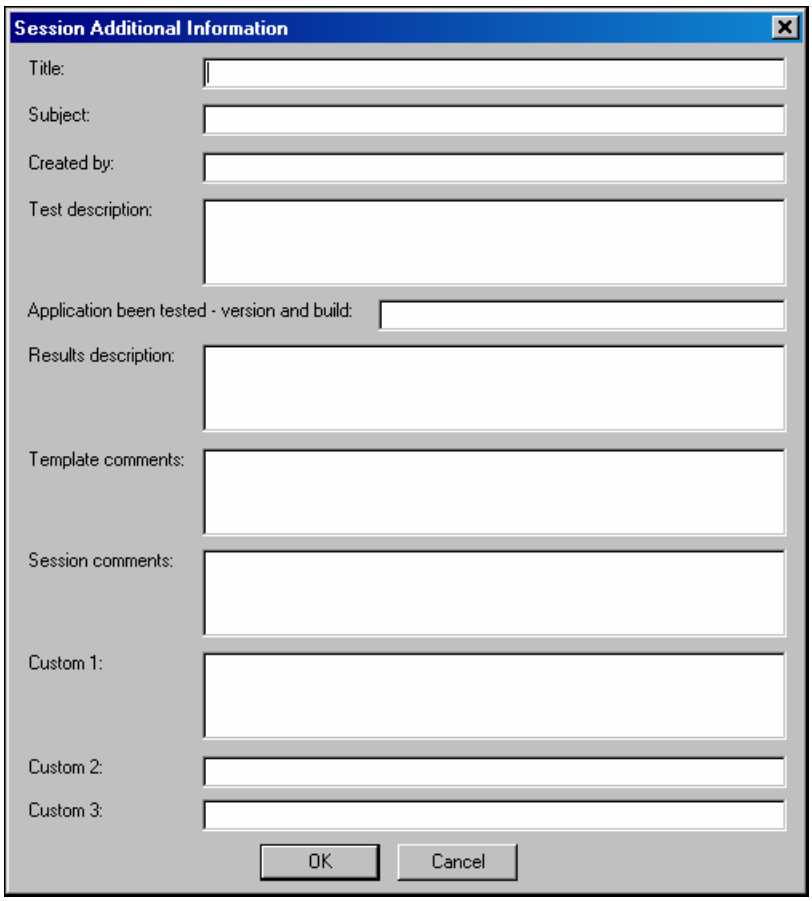
Setting Additional Information Properties

You can set additional information properties for the active load session.

To set additional information properties:

1. Select File | Additional information.

   The Session Additional Information dialog box opens.



2. Complete the fields to save additional information, useful for later reference, with the load session. Some of the fields, such as title, subject and application being tested, may already be completed if the information was saved with the load template.

   The following fields are available:

Field	Description
Title	Provides a space for you to type a title for this load session. The title can be different from the template file name.
Subject	Provides a space for you to type a description of the subject of the load session. Use this property to group similar load sessions together.

Field	Description
Created by	Provides a space for you to type the name of the person who authored this load session.
Test Description	Provides a space for you to type a description of the test objectives and what the load session is designed to test.
Application being tested - version and build	Provides a space for you to type the name, version and build number of the application being tested.
Results Description	Provides a space for you to type a description of the test results.
Template comments	Provides a space for you to type any comments regarding the load session.
Session comments	Provides a space for you to type any comments regarding the load session.
Custom	Provides a space for you to type any comments you want saved with this load session.

3. Click OK.

Opening Saved Load Sessions Files

To open a saved Load Session file:

1. Select File | Open Load Session and browse your directory structure for the location of the saved (`*.ls` file),

   -Or-

   Select the Open an Existing Load Session button from the WebLOAD Console dialog box at system start-up and browse your directory structure for the location of the saved (`*.ls`) file.

2. Double-click on the session you want to open.

   The file is opened, the test configuration is displayed in the session tree and the WebLOAD Default Report is displayed in the Results window.

C H A P T E R 1 1

Creating Load Templates with the WebLOAD Wizard

The easiest way to configure a load template is using the WebLOAD wizard.

In This Chapter

Using the WebLOAD Wizard to create Load Templates 133
The WebLOAD Wizard Workflow 134
Opening the WebLOAD Wizard 135
Selecting an Agenda or Mix 136
Selecting an Agenda 137
Selecting a Mix 138
Selecting Host Computers 141
Scheduling the Load Session 144
Completing the WebLOAD Wizard 158
After Completing the WebLOAD Wizard 159

Using the WebLOAD Wizard to create Load Templates

Using the wizard to create a load template, you define the following:

- Agenda(s) to run
- Load machines on which the load is generated
- Number of Virtual Clients to run
- Load schedule

You can also specify Agenda options including the type of browser to emulate connection speed, and playback sleep time options.

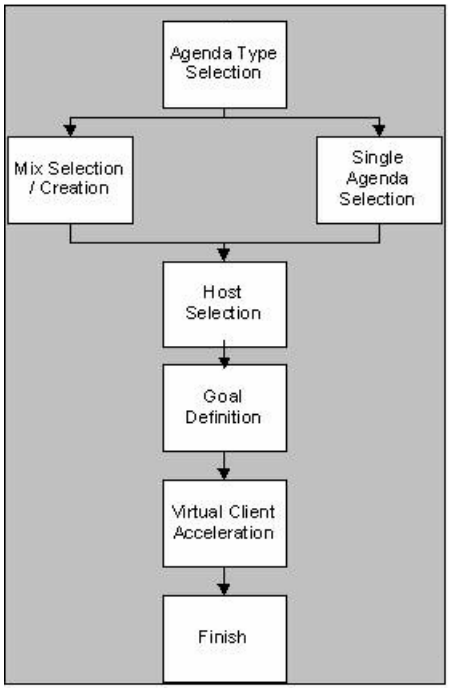
The WebLOAD wizard walks you through the configuration process step-by-step. Each screen of the WebLOAD wizard contains text explaining the configuration process.

After creating a load template with the WebLOAD Wizard, you can add functionality, not available in the wizard, through the Console menus. The Console allows you access to any part of the configuration process at any time. For details about the Console menus, see The Console Menu Options (on page 98).

Note: You cannot use the WebLOAD wizard if you have not yet recorded (or otherwise created) any Agendas. If you do not have any Agendas to work with, you must exit the wizard and create an Agenda using one of the authoring tools, such as WebLOAD IDE.

The WebLOAD Wizard Workflow

The following diagram illustrates the WebLOAD Wizard workflow:



Opening the WebLOAD Wizard

To open the WebLOAD Wizard:

1. Select WebLOAD wizard from the WebLOAD Console shortcut screen,

   -Or-

   Select File | WebLOAD Wizard,

   -Or-

   Click the WebLOAD wizard icon on the Console toolbar.

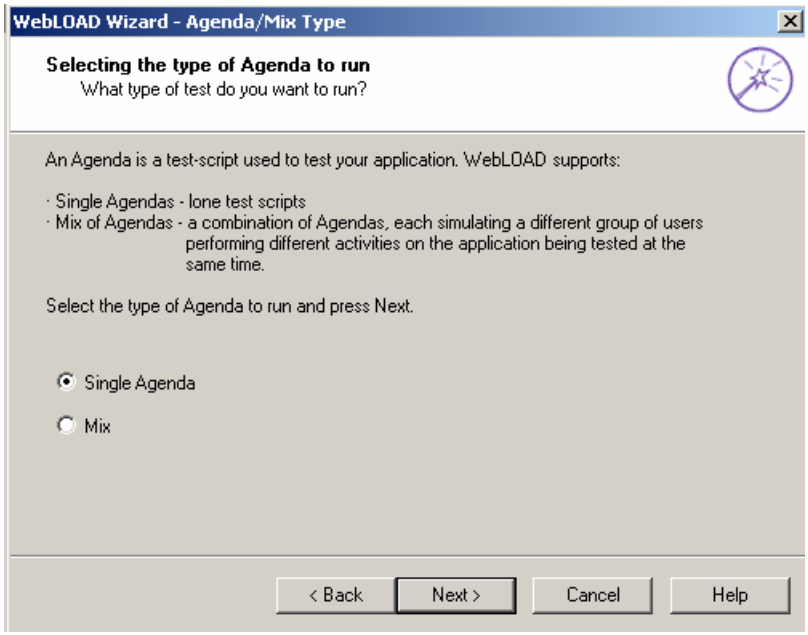
   The WebLOAD Wizard Welcome dialog box opens.

2. Click Next.

   The WebLOAD wizard progresses to the Agenda/Mix Type dialog box.

Selecting an Agenda or Mix

The next step in a WebLOAD session is to select the type of Agenda you want to run during your test session. An Agenda is the test script used to test your application.



WebLOAD supports two types of Agendas:

- Single Agendas - lone test-scripts.
- Mix of Agendas, - a combination of Agendas. Use a mix to simulate different groups of users performing different activities on the (ABT) at the same time.

To run a single Agenda:

1. Select Single Agenda.
2. Click Next.

   WebLOAD progresses to the Agenda Selection dialog box. See Selecting an Agenda (on page 137).

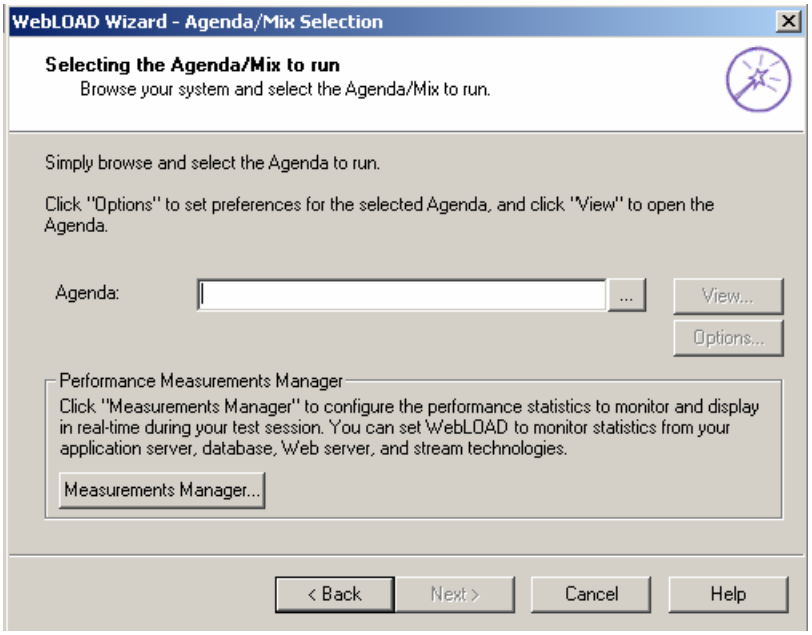
To run a Mix of Agendas:

1. Select Mix.
2. Click Next.

   WebLOAD progresses to the Mix Selection dialog box see Selecting a Mix (on page 138).

Selecting an Agenda

The WebLOAD wizard enables you to create a test using previously recorded Agendas. If you do not have any Agendas to work with, you must exit the wizard, and create an Agenda using one of the authoring tools, such as WebLOAD IDE.



To select an Agenda:

1. Click the browse button to select the source of the Agenda to be used for the test.
2. (Optional) To open and view or edit the selected Agenda, press View.

   WebFT opens displaying the selected Agenda.
3. (Optional) To configure runtime options exclusive to this Agenda, press Options.

   The Agenda Options dialog box opens enabling you to define the runtime options for the selected Agenda. For more information on Agenda options, see Setting Agenda Options (on page 221).
4. (Optional) Open the Performance Measurements Manager and configure WebLOAD to collect performance statistics from your application server, database, Web server and stream technologies while the test is running. Click Measurements Manager to configure the performance statistics to monitor and display in real-time during the test session.

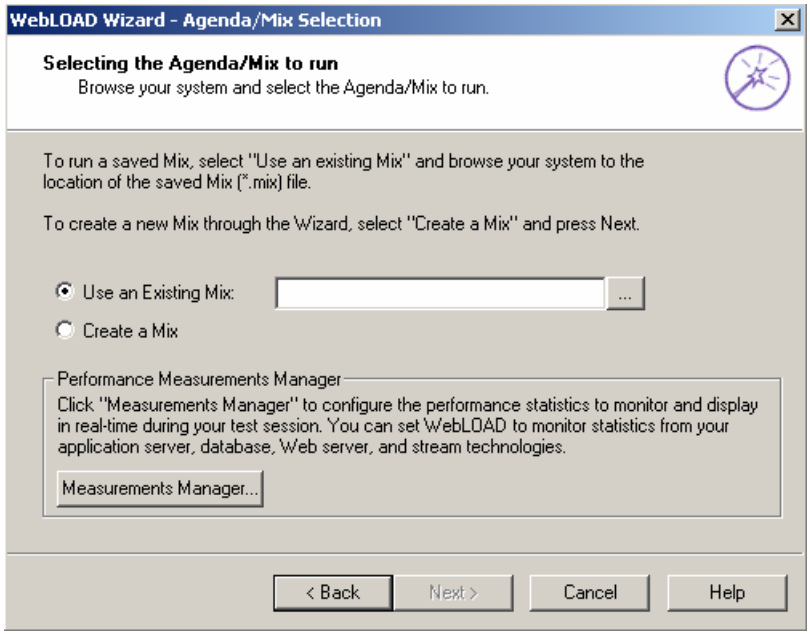
   The Performance Measurements Manager opens. For more information on configuring the statistics to monitor, see Performance Measurements Manager (on page 365).

5. Click Next.

   The Wizard progresses to the Host Selection dialog box. See Selecting Host Computers (on page 141).

Selecting a Mix

The WebLOAD wizard enables you to run your test using a saved Mix or to create a new Mix of Agendas through the wizard. If you do not have any Agendas to work with, to create a Mix, you must exit the wizard, and create an Agenda using one of the authoring tools, such as WebLOAD IDE.



To select a mix:

1. Select the source of the Mix to be used for the test on the Mix Selection screen.
   - Select Use an Existing Mix and click the browse button to choose an existing Mix (*.mix) from your system.
   - Select Create a Mix to build a new Mix.
2. (Optional) You can set WebLOAD to monitor and collect statistics from the application server, database, Web server and stream technologies. Click Measurements Manager to configure the performance statistics to monitor and display in real-time during the test session.

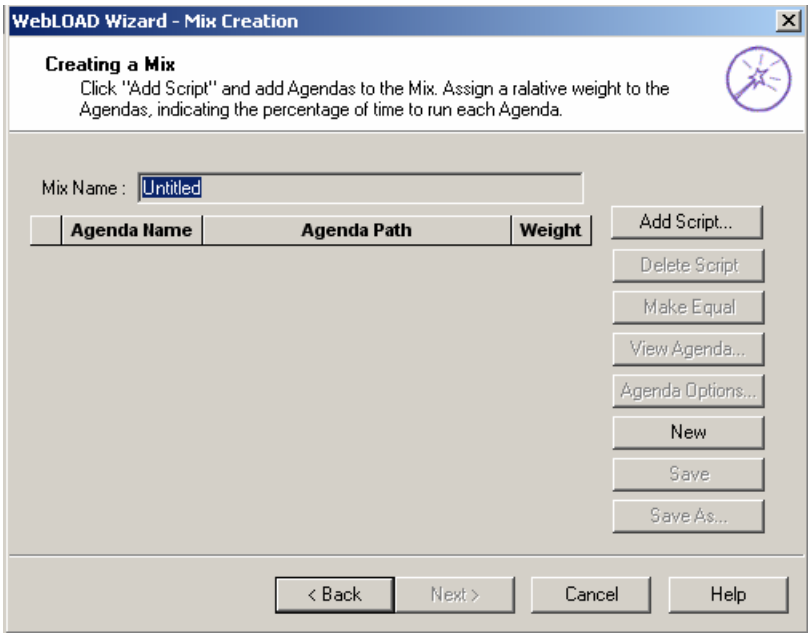
   The Performance Measurements Manager opens. For more information on configuring the statistics to monitor, see Performance Measurements Manager (on page 365).

3. Click Next.

   WebLOAD progresses to the Mix Definition dialog box. See Creating a Mix (on page 139).

Creating a Mix

A Mix is a set of Agendas, each performing different activity, to simulate groups of users performing different activities on the ABT at the same time. If you selected Use an Existing Mix on the Agenda / Mix Selection screen, the selected mix appears on the screen, enabling you to modify the Mix. You can also create a new Mix.



To create a new mix:

1. Click Add Script.

   The Open dialog box appears, allowing you to select the Agenda to add to the mix.

2. Select an Agenda from your file system and click Open.

   The Agenda is added to the mix.

3. (Optional) Select the Agenda and click Options to customize the settings (such as the browser to emulate, connection settings and playback sleep options) for only the current Agenda without affecting settings in other Agendas.

   The Agenda Options dialog box opens. For information on setting the Agenda Options, see Performance Measurements Manager (on page 365).

4. Repeat steps 1-3 to add additional Agendas to the mix.
5. In the Weight box adjacent to each Agenda, type or select the relative weight for the Agenda. The collective weight for all Agendas must be 100%. Click Make Equal to assign equal weight to all of the Agendas defined in the mix.
6. Click Save or Save As.
7. In the File name field, type a descriptive name for the Agenda and then click Save.

   The mix configuration is saved with the extension *.mix.
8. Click Next.

   WebLOAD displays the Host Selection dialog box. See Selecting Host Computers (on page 141).

Deleting an Agenda from a Mix

To delete an Agenda from a mix:

1. Select the Agenda you want to delete from the mix.
2. Click Delete Script.

Assigning Equal Weight to all Agendas in the Mix

To assign equal weight to all Agendas in the mix:

- Click Make Equal.

Creating a New Mix

To create a new mix:

- Click New.

  The screen is cleared.

Saving a Mix

After defining your mix you must save it as a *.mix file.

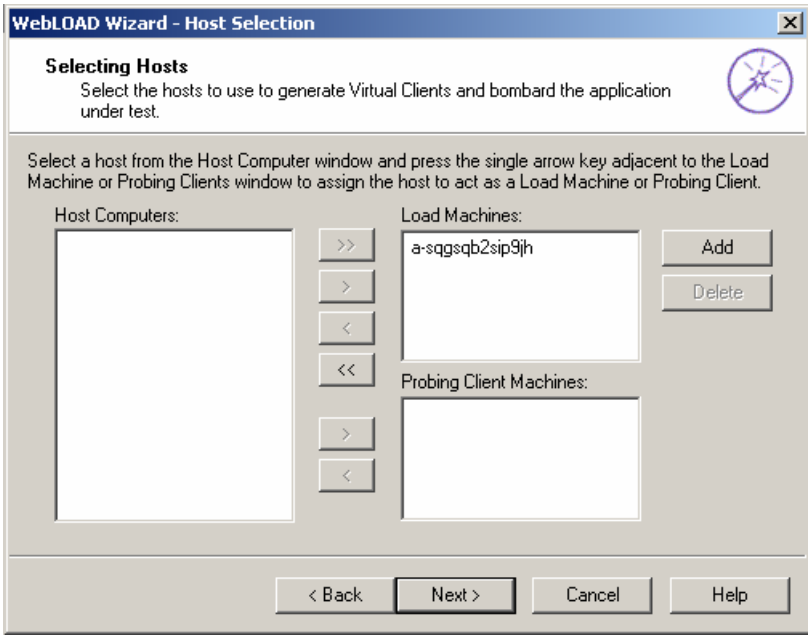
To save a mix:

1. Click Save or Save As.
2. In the File name field, type a descriptive name for the Agenda and then click Save.

   The mix configuration is saved with the extension .mix, and the wizard returns to the Mix Definition dialog box.

Selecting Host Computers

On the Host Selection dialog box you define the host computers (Load Machines and Probing Clients) participating in the test session.



Load Machines

By selecting a group of Load Machines, you define the pool of computers WebLOAD uses to generate the Virtual Clients. The number of Virtual Clients that can be generated on each host is dependent on the power of each host machine.

Probing Client Machines

Probing Clients are single instances of Virtual Clients running WebLOAD while the Load Machines bombard your Web application. WebLOAD reports exact measurements for Probing Clients.

This dialog box has three windows. The large window on the left, entitled Host Computers, displays all the host computers defined for testing. The two windows to the right contain the systems defined as Load Machines and Probing Client Machines. Use the arrow buttons between the windows to change the roles of the host computers.

Your local system is automatically defined as the default Load Machine.

Notes:

In Standalone Workstation mode only the local machine is available to act as a Load Machine

At least one system must be configured as a Load Machine to continue working with the wizard.

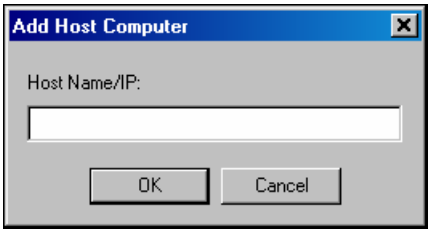
After configuring the hosts participating in the test session, the WebLOAD Wizard progresses to the Schedule dialog box. See Scheduling the Load Session (on page 144).

Adding Host Computers

To add a host computer:

1. In the Host Selection dialog box, click Add.

   The Add Host Computer dialog box displays.



2. Insert the host computer name or IP address in the Host Name / IP field.
3. Click OK.

   The host is added to the Host Computers window.

   For defining host computers as Load Machines and Probing Clients, see Designating a Host Computer as a Load Machine (on page 142), and Designating a Host Computer as a Probing Client Machine (on page 143).

Designating a Host Computer as a Load Machine

To designate a host computer as a Load Machine:

1. Select the host from the Host Computers window.
2. Press the single arrow key adjacent to the Load Machine window.

   The selected host moves to the Load Machines window.

3. To designate additional Load Machines repeat steps 1 and 2.

   After assigning all Load Machines and Probing Client Machines, click Next. The wizard progresses to the Schedule dialog box. See Scheduling the Load Session (on page 144).

Designating all Host Computers as Load Machines

To designate all host computers as Load Machines:

- Press the double arrow key adjacent to the Load Machines window.

  All of the host computers move to the Load Machines window.

Designating a Host Computer as a Probing Client Machine

Probing Clients act as single Virtual Clients to measure the performance of targeted activities and provide individual performance statistics of the ABT.

Note: Probing Client Machines cannot run Mix Agendas.

To designate a host computer as a Probing Client Machine:

1. Select the host from the Host Computers window.
2. Press the single arrow key adjacent to the Probing Client Machines window.

   The selected host moves to the Probing Client Machine window.
3. After assigning all Load Machines and Probing Client Machines, click Next.

   The wizard progresses to the Schedule screen. See Scheduling the Load Session (on page 144).

Replacing a Load Machine or Probing Client Machine

To replace a Load Machine or Probing Client Machine:

1. From the Host Selection dialog box, select the host computer in the Load Machines or Probing Client Machine window.
2. Click the left arrow button .

   The host computer is moved from the Load Machines or Probing Client Machine window to the Host Computers window.

3. From the Host Computers window, select a host computer.
4. Click the right arrow button .

   The host computer is moved from the Host Computers window to the Load Machines or Probing Client Machine window.

Deleting a Host Computer

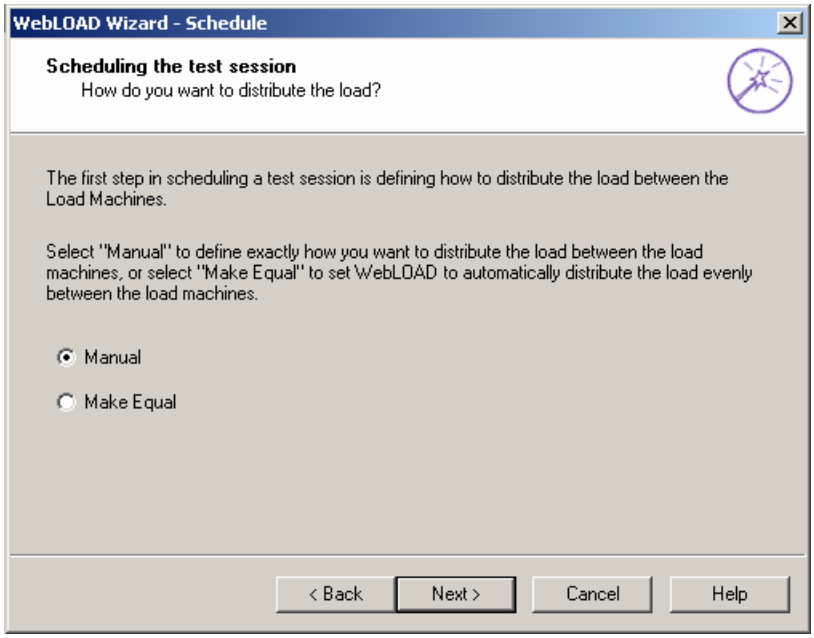
To delete a host computer:

1. Select the host name from the Host Computers list.
2. Click Delete.

   The host is deleted from the Host Computers list.

Scheduling the Load Session

When you schedule a test session, you define the load to be generated throughout the load session. Use the Schedule dialog box to plan your test session.

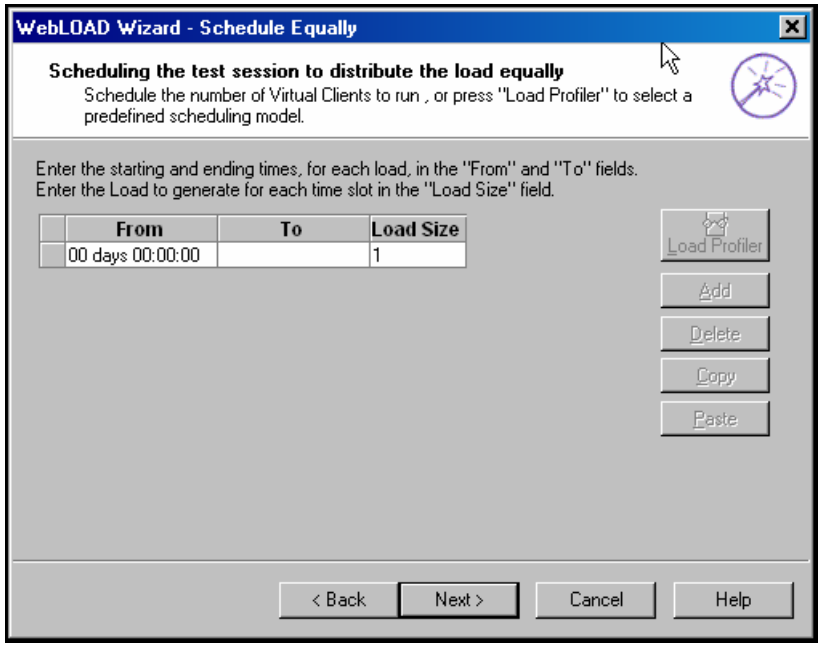


To schedule the load session:

1. Define how you want the to distribute the load between the Load Machines running the Agenda.
   - Select Manual to instruct WebLOAD how to distribute the load over the Load Machines throughout the test.
   - Select Make Equal to distribute the load evenly between the Load Machines.
2. Click Next.
   - If you selected Make Equal, the WebLOAD wizard progresses to the Schedule Equally screen. See Distributing the Load Equally (on page 145).
   - If you selected Manual, the Wizard progresses to the Schedule Manually screen. See Manually Scheduling the Load to Generate on Each Load Machine (on page 148).

Distributing the Load Equally

WebLOAD simplifies the process of scheduling the test session by enabling you to automatically distribute the load between the Load Machines, saving you the time of creating a load schedule for each Load Machine individually.



You can create a load schedule by:

- Manually defining time frames and the load to generate over each frame
- Using the Load Profiler
- A combination of both

Creating an Equal Load Schedule Manually

To create an equal load schedule manually:

1. Enter the number of days and the starting time to generate the load in the From field.

   The starting time is defined for each particular Agenda relative to the beginning of the test session.

   Note: Days are numbered 00 through to 99. Following them are hours:minutes:seconds. Seconds are numbered 1 through 59. To enter one minute, use the 1:00 syntax not 00:60.

2. Enter the number of days and the ending time to generate the load in the To field.

   The ending time is defined for each particular Agenda relative to the beginning of the test session.

   If the To field is left blank, WebLOAD continues generating the load until the From time in the next entry for this Agenda, or, if this is the last entry for this Agenda, until the test session is stopped manually.

3. Under Load Size enter the number of Virtual Clients to generate during the time frame defined in steps 1 and 2. The load is distributed equally among the Load Machines defined for the Agenda.

4. (Optional) Schedule additional Virtual Client loads for this Agenda:

   a. Click Add to add a blank line to the schedule grid.

   b. Repeat steps 1 through 2.

5. Click Next.

   WebLOAD displays the Finish dialog box.

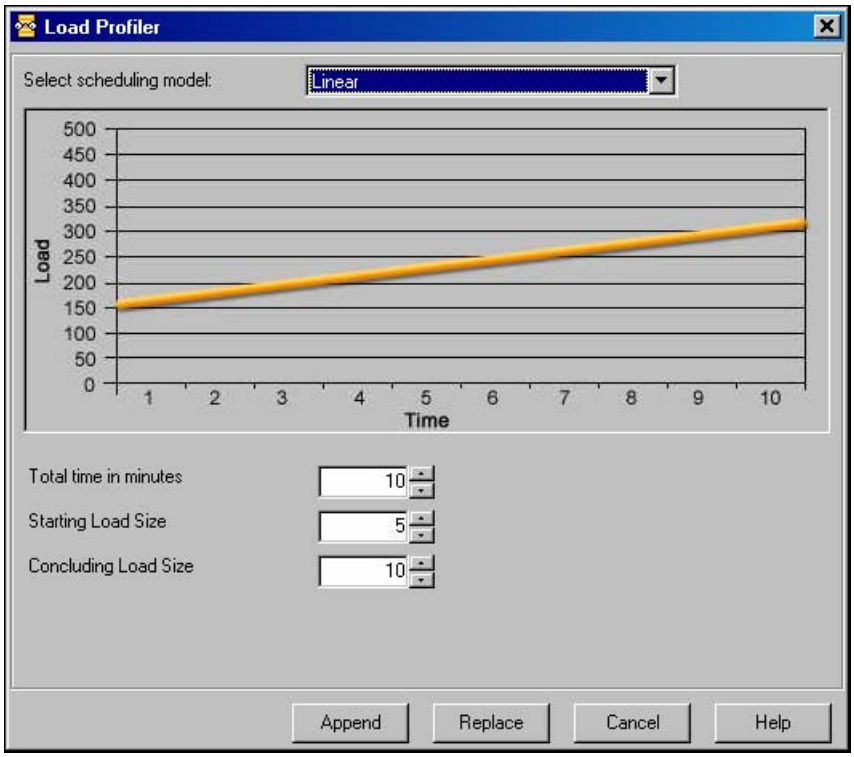
The WebLOAD wizard continues the configuration process with the Finish dialog box. For further instructions, see Completing the WebLOAD Wizard (on page 158).

Creating an Equal Load Schedule with the Load Profiler

To create an equal load schedule with the Load Profiler:

1. Select an entry in the schedule grid by clicking on it to enable the Load Profiler button for that entry.
2. Click on the Load Profiler button.

   The Load Profiler opens.



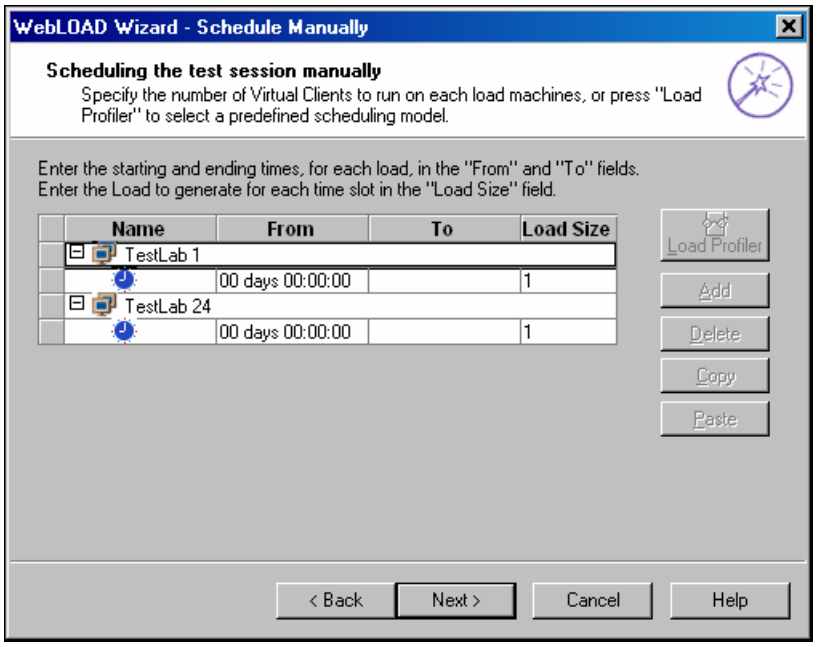
3. Select a schedule pattern from the Scheduling model field at the top of the dialog box. For an explanation of each model, see Scheduling Models (on page 150).
4. Fill in the fields at the bottom of the dialog box.
5. Click Append to include the load definition in the schedule,

   -Or-

   Click Replace to substitute the load definition for the one appearing in the Schedule dialog box.

Manually Scheduling the Load to Generate on Each Load Machine

You can manually schedule the load to be generated on each Load Machine participating in the test session, enabling you to distribute the load based on the strengths of the Load Machines running in the test session.



Scheduling the Session Manually

To schedule the session manually:

1. Expand the Load Machine tree using the + / - button adjacent to the Load Machine you want to schedule.
2. Enter the number of days and the starting time, to generate the load in the From field.

   The starting time is defined for each particular Agenda relative to the beginning of the test session.

   Note: Days are numbered 00 through to 99. Following them are hours:minutes:seconds. Seconds are numbered 1 through 59. To enter one minute, use the 1:00 syntax not 00:60.

3. Enter the number of days and the ending time, to generate the load in the To field.

   The ending time is defined for each particular Agenda relative to the beginning of the test session. If the To field is left blank, WebLOAD continues generating the load until the From

time in the next entry for this Load Machine, or if this is the last entry, until the test session is stopped manually.

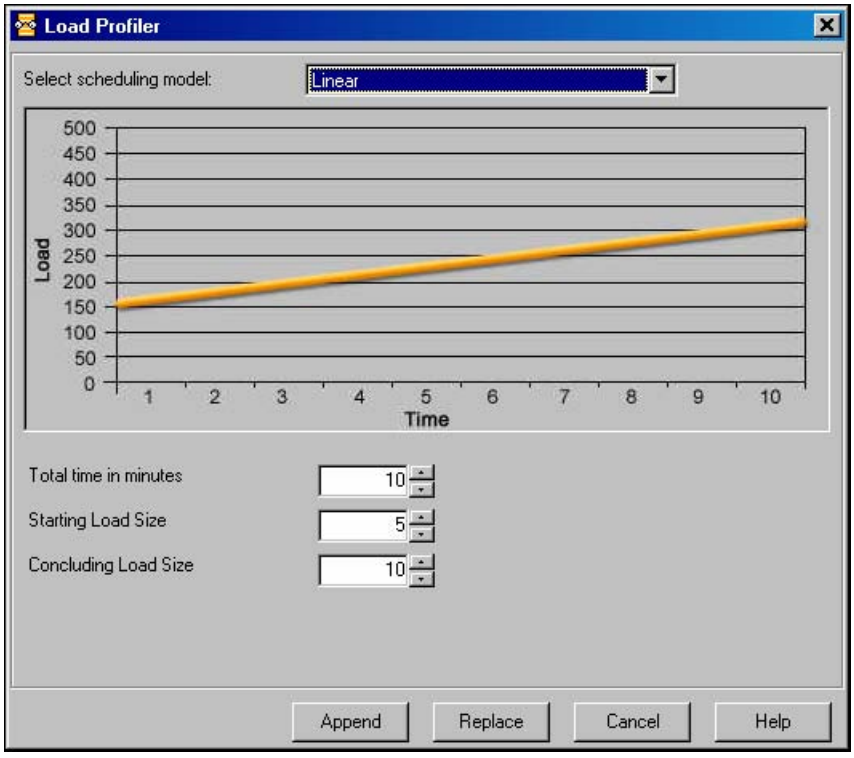
4. In the Load Size field enter the number of Virtual Clients to generate during the time frame defined in steps 2 and 3.
5. (Optional) Schedule additional Virtual Client loads for this Load Machine:
   a. Click Add to add a blank line to the schedule grid.
   b. Repeat steps 2 through 4.
6. Repeat steps 2 through 5 for each Load Machine or Probing Client Machine defined in the session tree.
7. Click Next.

   The WebLOAD wizard continues the configuration process with the Finish screen. For further instructions, go to Completing the WebLOAD Wizard (on page 158).

Scheduling a Session Manually with the Load Profiler

1. Click on the Load Profiler button.

   The Load Profiler opens.



2. Select a pattern from the Scheduling model field at the top of the dialog box.

   For an explanation of each model, see Scheduling Models (on page 150).

3. Fill in the fields at the bottom of the dialog box.
4. Click Append to include the load definition in the schedule,

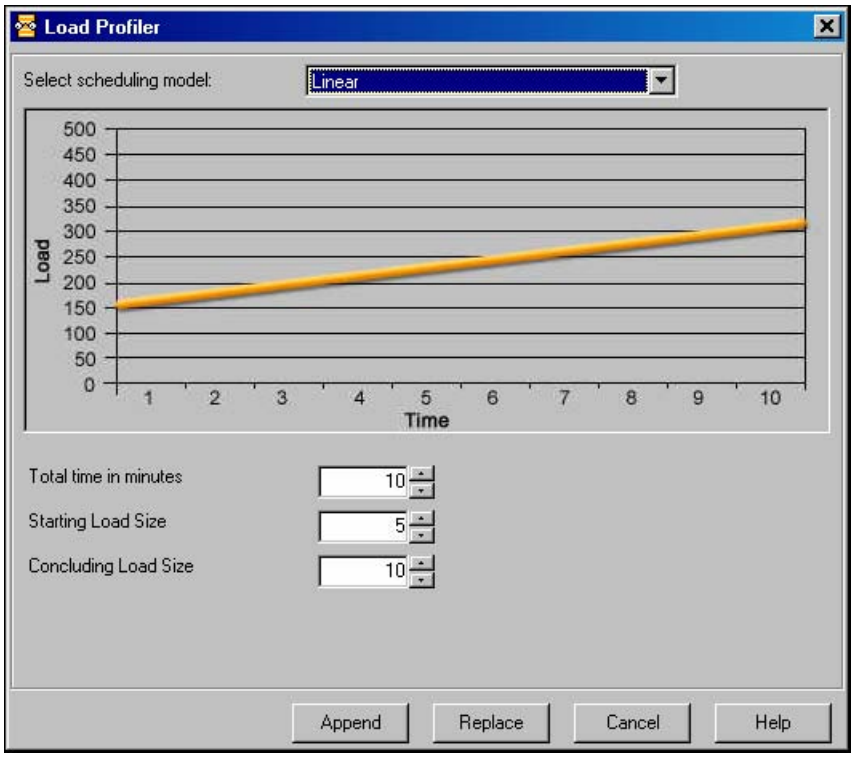
   -Or-

   Click Replace to substitute the load definition for the setting in the Schedule dialog box.

Scheduling Models

WebLOAD includes numerous scheduling models to provide a wide variety of options and aid you in quickly achieving a schedule that meets you testing goals.

Linear Scheduling Model



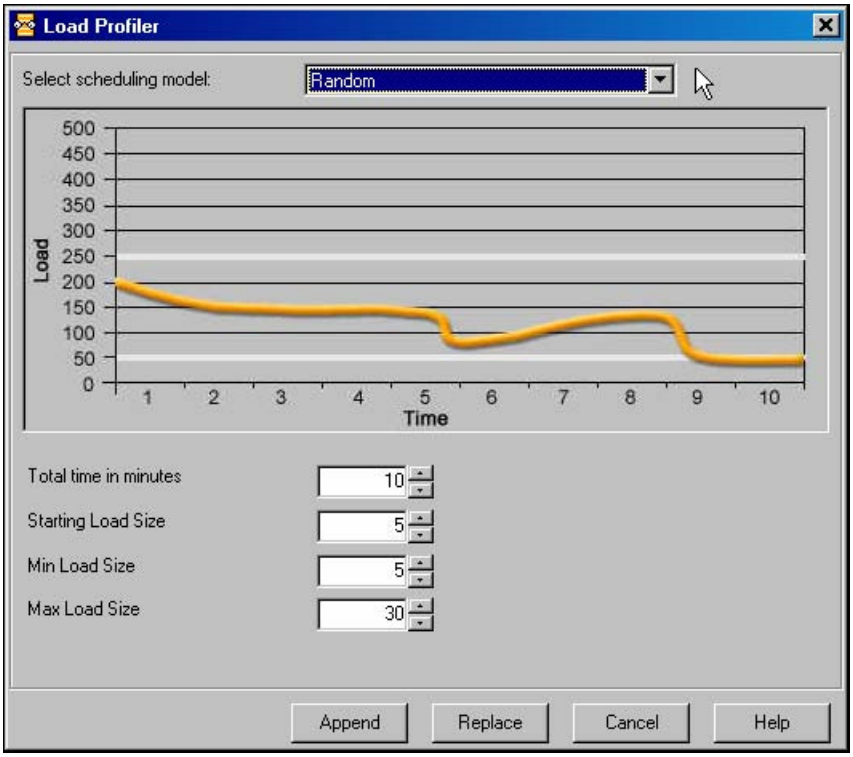
The linear model increases the load with linearly increasing values.

Using this model you need to define:

- Total time in minutes - the number of minutes to generate the load using this model.
- Starting Load Size - the load size to begin from.
- Concluding Load Size - the load size to end with.

For example, using this model the load will start at the value defined in the Starting Load Size field and increase the load (automatically calculating the intervals when the load should increase) over the time specified in the Total time in minutes field to the load size defined in the Concluding Load Size field.

Random



The random model scatters the load with no discernable pattern.

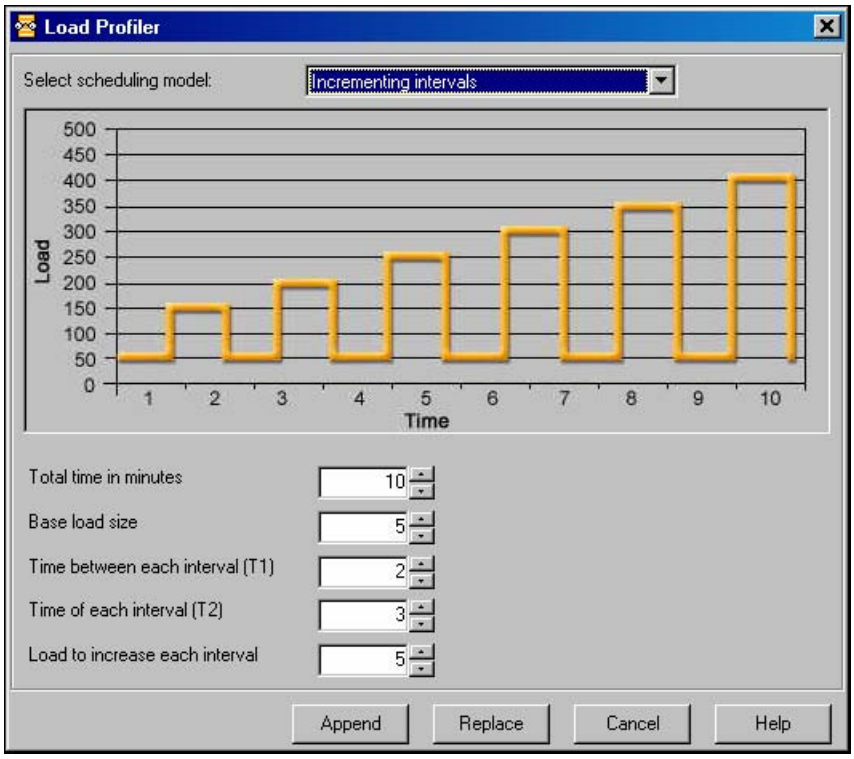
Using this model, you need to define:

- Total time in minutes - the number of minutes to generate the load using this model.
- Starting Load Size - the load size to begin from.
- Min. Load Size - the absolute minimum number of users to generate.
- Max. Load Size - the absolute maximum number of users to generate.

For example, using this model the load will start at the value defined in the Start Load Size field and randomly increase and decrease the load between the min and max load values with no

relationship or trend between each interval until the time defined in the Total time in minutes field has expired.

Incrementing Intervals



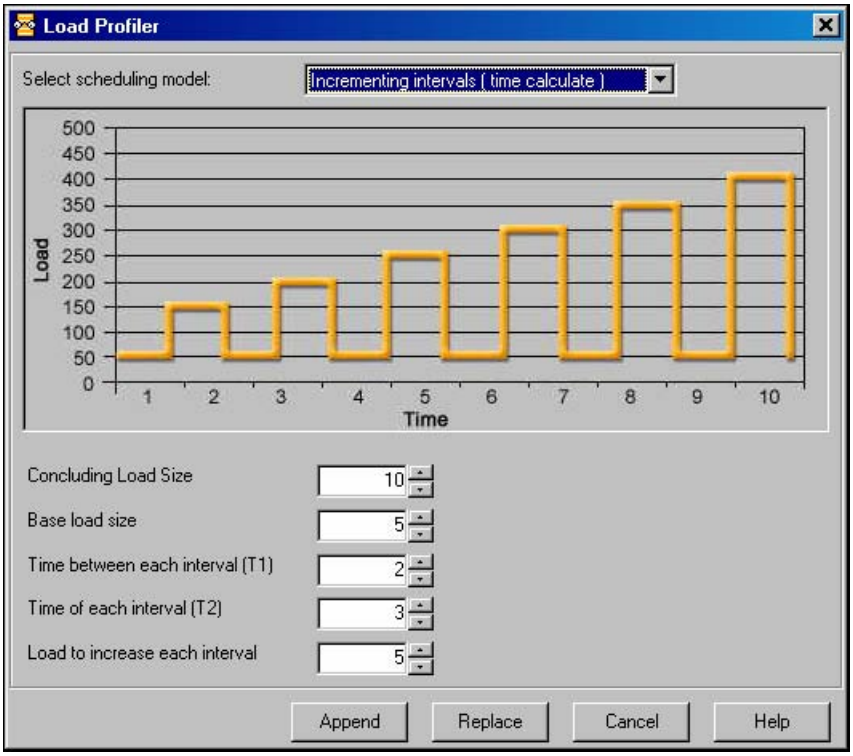
The incrementing intervals model generates an increasing load in intervals while returning to the base load between intervals.

Using this model you need to define:

- Total time in minutes - the number of minutes to generate the load using this model.
- Base Load Size - the load size to begin from, and to return to between each interval.
- Time Between Each Interval-T1 - the amount of time to wait before increasing the load for each interval.
- Time of Each Interval-T2 - the length of time for each interval.
- Load to Increase Each Interval - the number of users to add each interval (over the load generated in the last interval).

For example, using this model the load will start at the Base Load Size, increase for the time defined as T2, return to the base load and then increase again. Each subsequent increase will be the Load Size from the previous interval plus the increment.

Incrementing Intervals (time calculate)



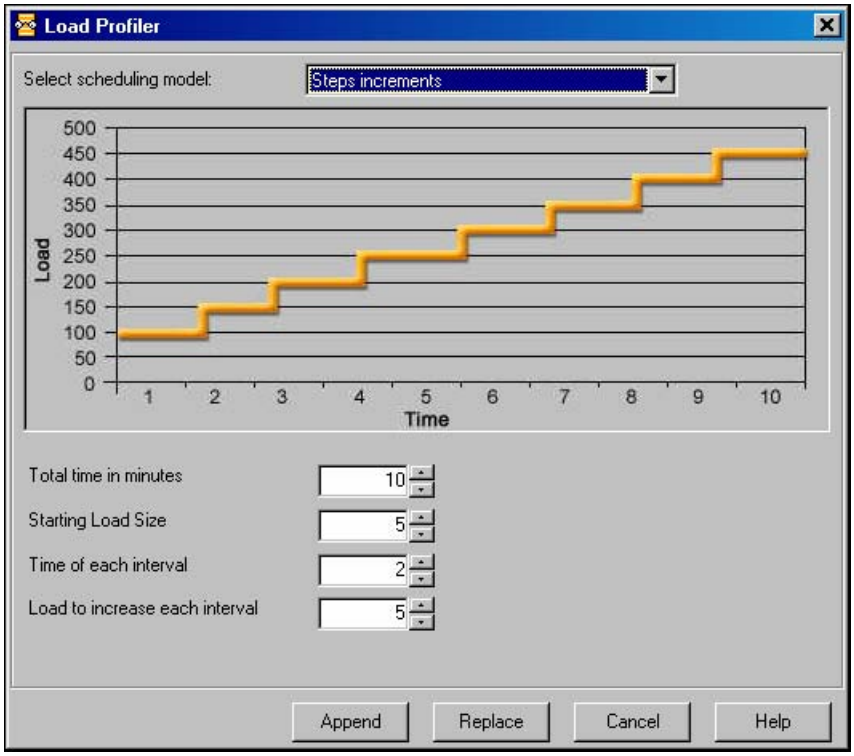
The incrementing intervals (time calculate) model generates an increasing load in intervals while returning to the base load between intervals. The schedule stops when the concluding load size is achieved.

Using this model you need to define:

- Concluding Load Size - the load size to end with.
- Base Load Size - the load size to begin from, and to return to between each interval.
- Time Between Each Interval-T1 - the amount of time to wait before increasing the load for each interval.
- Time of Each Interval-T2 - the length of time for each interval.
- Load to Increase Each Interval - the number of users to add each interval (over the load generated in the last interval).

For example, using this model the load will start at the Base Load Size, increase for the time defined as T2, return to the base load and then increase again. Each subsequent increase will be the Load Size from the previous interval plus the increment until the concluding load size is achieved. The time it takes to complete the task is calculated automatically by WebLOAD based on the parameters defined.

Step Increments



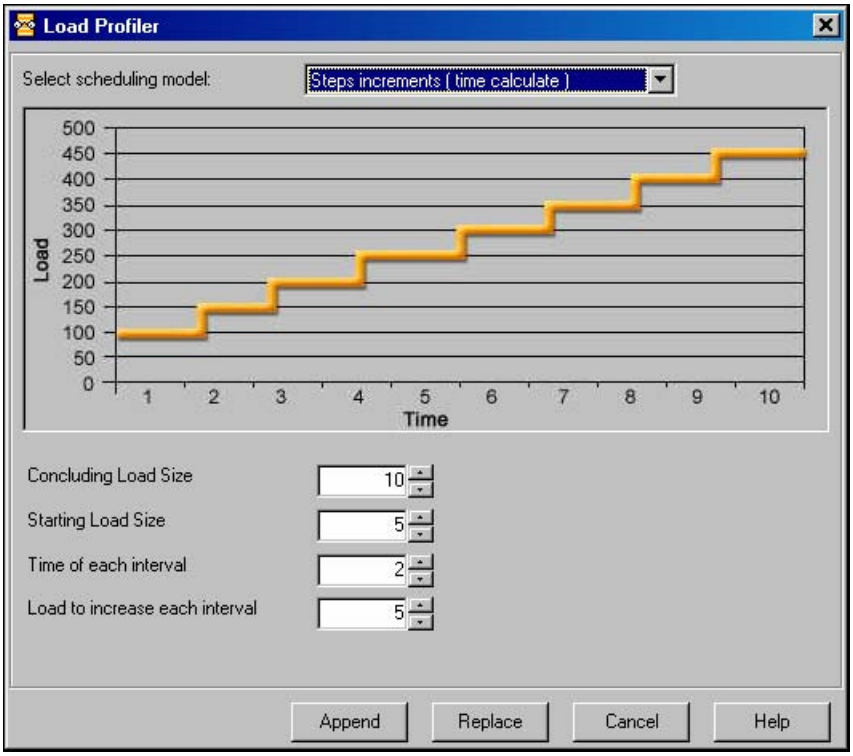
The step interval model generates an increasing load in intervals.

Using this model you need to define:

- Total time in minutes - the number of minutes to generate the load using this model.
- Starting Load Size - the load size to begin from.
- Time of each Interval - the length of time for each interval.
- Load to increase each interval - the number of Virtual Clients to increase each interval.

For example, using this model the load will start at the value defined in the Start Load Size field, and increase gradually the amount of user defined in the Load to increase each interval field.

Step Increments (time calculate)



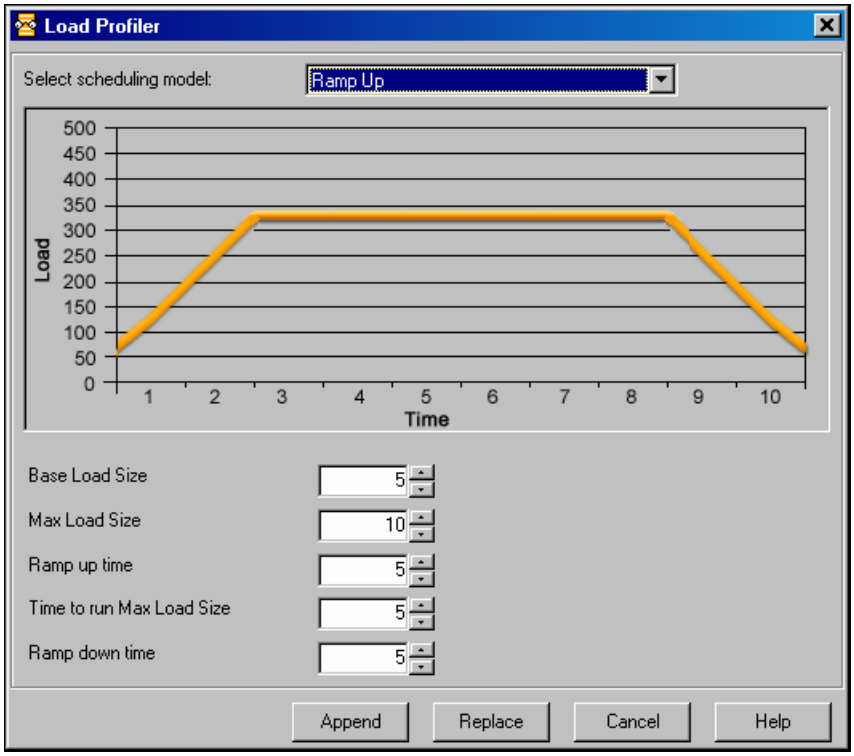
The step interval model generates an increasing load in intervals until a defined load size is reached. The time to complete the scheduled model is calculated automatically by WebLOAD based on the parameters defined.

Using this model you need to define:

- Concluding Load Size - the load size to end with.
- Starting Load Size - the load size to begin from.
- Time of each Interval - the length of time for each interval.
- Load to increase each interval - the number of Virtual Clients to increase each interval.

For example, using this model the load will start at the value defined in the Start Load Size field, and increase gradually the amount of user defined in the Load to increase each interval field until the concluding load size is achieved.

Ramp Up



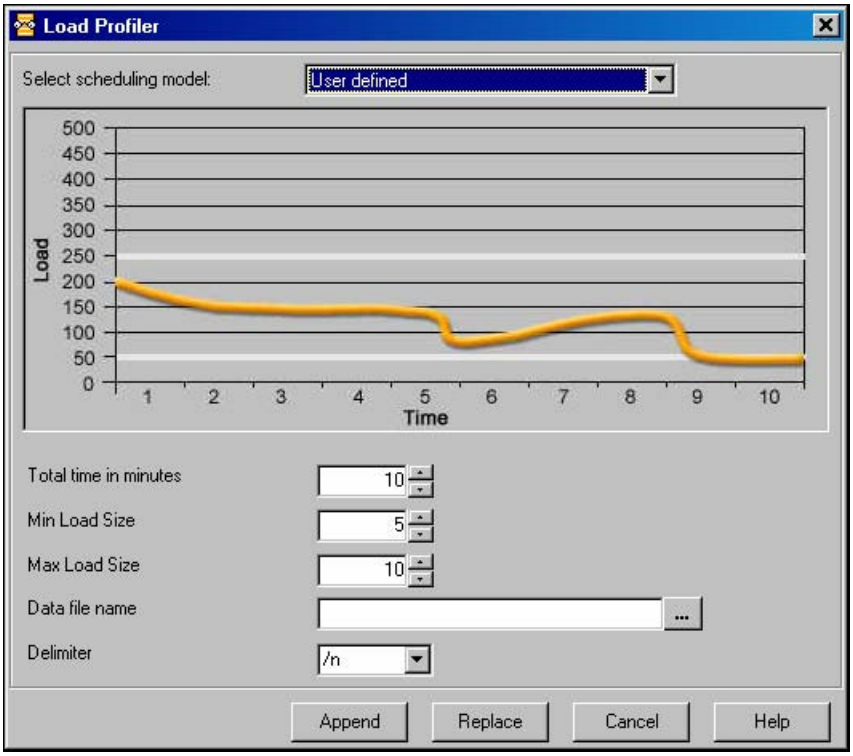
The ramp up model generates an increasing load over a set time frame to until reaching a defined maximum load size. The maximum load size is then run for a set period and then the load is reduced back to the base load again over a time frame defined by the user. The load increments are calculated automatically by WebLOAD based on the parameters defined.

Using this model you need to define:

- Base Load Size - the load size to begin with.
- Max Load Size - the maximum number of Virtual Clients to generate using this model.
- Ramp UpTime - the time interval over which to increase the load in order to reach the max load size.
- Time to Run Max Load Size - the time frame to run the load at the max load size.
- Ramp Down Time - the time interval over which to decrease the load in order to return to the base load size.

For example, using this model the load will start at the value defined in the Base Load Size field, and increase over the time frame defined in the Ramp Up Time field to reach the load size defined in the Max Load Size field. The test will then continue running with the max load size for the period of time defined in the Time to Run Max Load Size field after which, load will return to the base load size value over the time frame set in the Ramp Down Time field.

User Defined



The user-defined model generates the load based on values extracted from a data file. The time to complete the scheduled model is defined by the user as is the minimum and maximum load size.

Using this model you need to define:

- Total time in minutes - the number of minutes to generate the load using this model.
- Min Load Size - the absolute minimum load size to generate over the schedule. At no point in the schedule should the load size dip below this value.
- Max Load Size - the absolute maximum load size to generate over the schedule. At no point in the schedule should the load size raise above this value.
- Data file name - the name and location on the data file to read values from for generating the load.
- Delimiter - the delimiter used in the data file from which values are read to generate the load size.

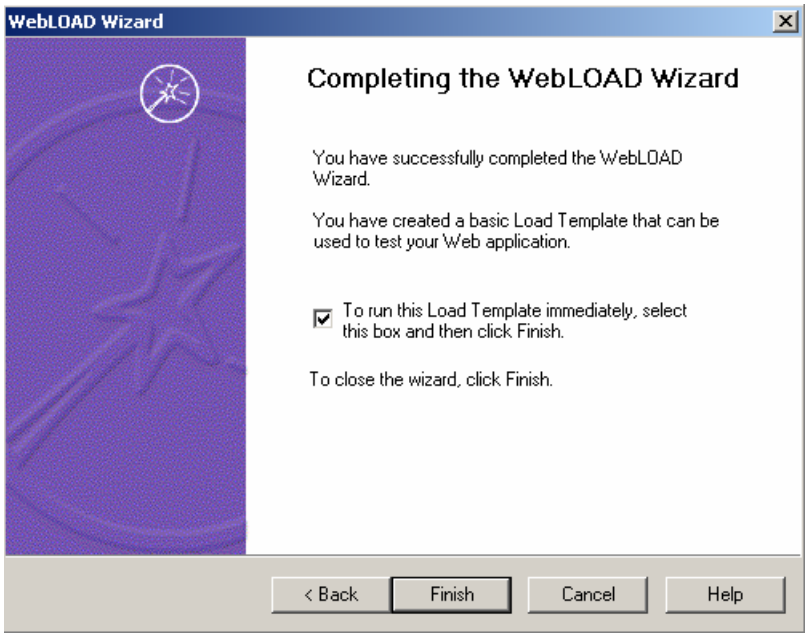
Using this model the TestView Scheduler will read values from the assigned data file and generate the load based on the values in the file. The load size will fluctuate based on the number of values in the file and the time frame for which the schedule is set to run.

Completing the WebLOAD Wizard

The product of your performance testing is the WebLOAD report. Here you can see how your Web application handles the load that you have tested. The WebLOAD wizard automatically generates a predefined report that measures:

- Hits per Second
- Round Time (the amount of time required to run one round of the Agenda)
- Load Size (the number of Virtual Clients being generated)
- Throughput (the number of bytes per second transmitted from the ABT)

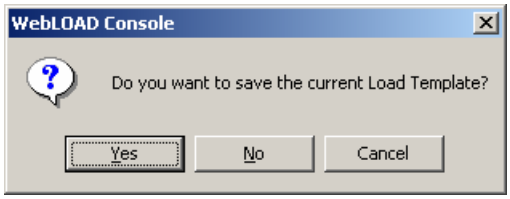
WebLOAD saves all the values for the complete set of measurements. You can decide after the test has run to display the results never before requested.



To complete the WebLOAD Wizard:

1. Select the check box to run the load template immediately.
2. Click Finish.

   A message box appears asking if you want to save the current load template.



3. Click Yes to save the current load template as a *.tpl file.

   Load templates contains the parameters that you defined for the test session, such as, Agendas to run, participating hosts, and the test plan. Using load templates saves you the time of reconfiguring the load session when repeating tests. For more information on load templates see Managing Load Templates (on page 125).

   The WebLOAD wizard closes and the load test begins.

After Completing the WebLOAD Wizard

After you have completed the WebLOAD wizard you can do the following:

- Edit the load template
- View the test results
- Modify the host selection
- Modify the load schedule

Editing the Load Template

After completing the WebLOAD wizard, a load template with one load machine is created. You can save this load template for future load sessions. The load template can then be edited through the Console menu and toolbar.

Viewing the WebLOAD Wizard Default Report

The WebLOAD wizard automatically generates a predefined report that measures:

- Hits per Second
- Round Time (the amount of time required to run one round of the Agenda)
- Load Size (the number of Virtual Clients being generated)
- Throughput (the number of bytes per second transmitted from the ABT)

You can configure your own report through the Console Reports menu. WebLOAD saves all of the values for the complete set of measurements. You can display the results never before requested. For information on viewing reports, see the material on running tests and analyzing the results.

Modifying the Host Selection

After closing the WebLOAD wizard you can edit the parameters in the Host Selection dialog box.

To modify the host selection:

1. Select Session Control | Modify Host Selection.

   The Host Selection dialog box opens.
2. Make the changes as needed.
3. Click OK.

   The host selection parameters are changed.

Note: If after closing the WebLOAD wizard you want to edit parameters in the Host Selection dialog box, select Session Control | Modify Host Selection.

Modifying the Load Schedule

After closing the WebLOAD wizard you can edit the parameters in the load schedule.

To modify the load schedule:

1. Select Console | Session Control | Modify Schedule.

   The Load Schedule screen appears.
2. Make the changes.
3. Click OK.

   The load schedule is changed.

C H A P T E R 1 2

Creating Load Templates with Cruise Control

WebLOAD offers the Cruise Control Wizard, a unique feature that enables goal-seeking performance testing.

In This Chapter

Using Cruise Control 161
How does Cruise Control Work? 162
The Cruise Control Wizard Workflow 163
Opening the Cruise Control Wizard 163
Selecting an Agenda or Mix 165
Running Single Agendas 166
Running a Mix of Agendas 167
Selecting Host Computers 172
Defining Performance Goals 175
Defining the Acceleration Rate 182
Defining WebLOAD's Action Once the Goals are Reached 183
Finishing the Cruise Control Wizard 184
Viewing the Cruise Control Default Report 186

Using Cruise Control

Cruise Control is a predictive feature within WebLOAD that identifies the performance threshold of a Web application. You define one or multiple performance parameters and Cruise Control automatically creates a schedule and increases the load until your Web application

performance falls below the user-defined profile. Cruise Control incorporates intelligent, distributed load generation capabilities for virtually unlimited load generation.

Using the Cruise Control wizard to create a load template, you define:

- The Agenda(s) to run.
- The load machines on which the load is generated.
- The performance goals to achieve.

You can also specify Agenda options including the type of browser to emulate, connection speed, and playback sleep time options.

Note: You cannot use the Cruise Control wizard if you have not yet recorded (or otherwise created) any Agendas. If you do not have any Agendas to work with, you must exit the wizard and create an Agenda using one of the authoring tools, such as WebLOAD IDE.

How does Cruise Control Work?

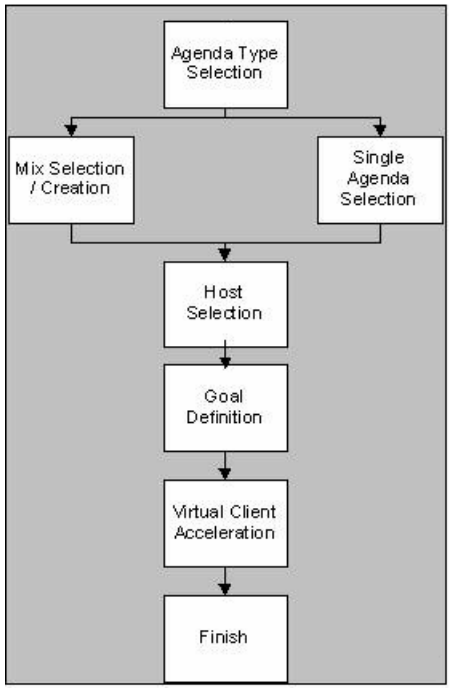
When operating WebLOAD manually, you specify the number of Virtual Clients that WebLOAD should simulate. The evaluation of the ABT's performance is based on this number. For example, you might configure WebLOAD to test the performance of your Web application when 200 Virtual Clients perform a certain function simultaneously. WebLOAD runs the test and may report that when 200 Virtual Clients are running, the application Response Time is “x”.

Cruise Control is different. Cruise Control enables you to specify the performance goals that you want to achieve, and WebLOAD generates a Cruise Control test that automatically schedules a test to achieve the specified goals. For example, if your performance goal is for your application server's response time not to exceed three seconds, Cruise Control devises the test for you. Cruise Control tells you how many users your application server is able to accommodate when your performance goals are met.

Cruise Control is a unique WebLOAD feature. It enables you to devise your test to reflect the real way that you want to see your application performance. Many times, you don't know the number of clients that will access your site, but you do know the quality of service you want to achieve. Rather than running multiple tests to measure the performance at different loads, you can run Cruise Control once, measuring the performance level you desire.

The Cruise Control Wizard Workflow

The following diagram illustrates the Cruise Control Wizard workflow.



Opening the Cruise Control Wizard

To open the Cruise Control Wizard:

1. Open the Cruise Control Wizard in one of the following ways:
   - Select Cruise Control Wizard from the Console shortcut dialog box at system startup,

     -Or-
   - Select File | Cruise Control Wizard from the Console menu,

     -Or-
   - Click the Cruise Control Wizard icon from the Console toolbar.

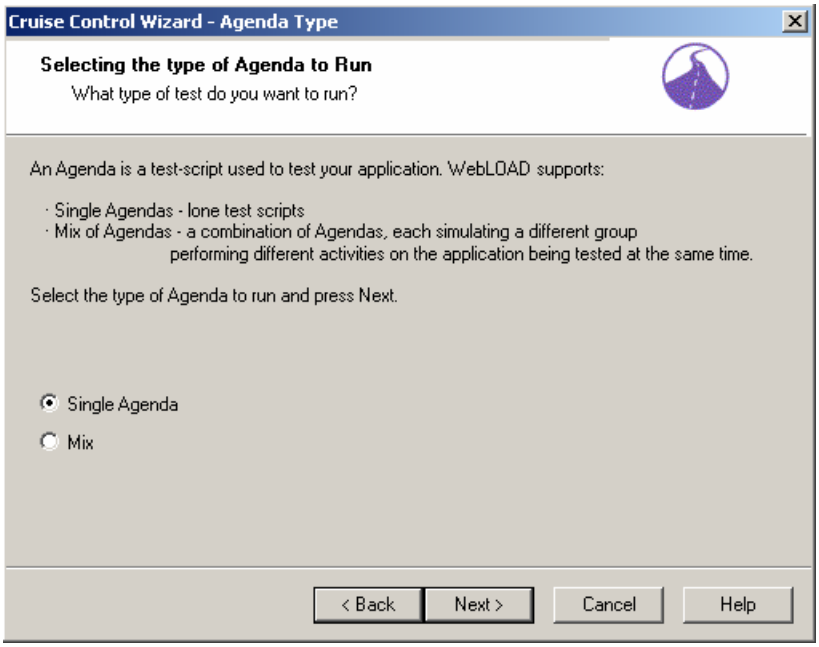
The Cruise Control Wizard Welcome dialog box opens.

2. Click Next.

   The Cruise Control Wizard progresses to the Agenda Type screen.

Selecting an Agenda or Mix

The next step in configuring a Cruise Control test is to define the type of Agenda you want to run during your test session. Agendas are test scripts (scenarios) that describe user activity, and are used to test your Web application.



WebLOAD supports two types of Agendas:

- Single Agendas, which are lone test-scripts.
- Mix of Agendas, which are a combination of Agendas used to simulate groups of users performing different activities on the application being tested (ABT) at the same time.

To run a single Agenda:

1. Select Single Agenda.
2. Click Next.

   WebLOAD progresses to the Agenda Selection screen. See Running Single Agendas (on page 166).

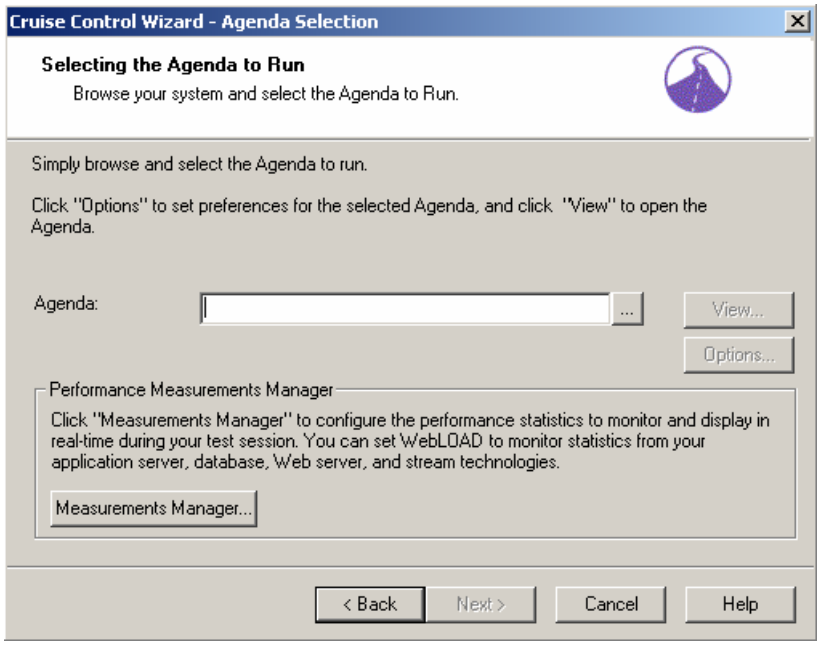
To run a Mix of Agendas:

1. Select Mix.
2. Click Next.

   WebLOAD progresses to the Mix Selection screen. See Running a Mix of Agendas (on page 167).

Running Single Agendas

You must have an Agenda to continue configuring a test with the Cruise Control wizard. If you do not have any Agendas to work with, you must exit the wizard and go into one of the authoring tools, for example WebLOAD IDE, and create an Agenda.



To run an agenda:

1. Select the source of the Agenda to be used for this test. Press the Browse button and choose an existing Agenda from your system.
2. (Optional) To open and view or edit the selected Agenda, press View.

   WebFT opens displaying the selected Agenda.
3. (Optional) To configure runtime options exclusive to this Agenda, press Options.

   The Agenda Options dialog box opens enabling you to define the runtime options for the selected Agenda. For more information on Agenda options, see Setting Agenda Options *(on page 221)*.

4. (Optional) Open the Performance Measurements Manager and configure WebLOAD to collect performance statistics from your application server, database, Web server and stream technologies while the test is running. Click Measurements Manager to configure the performance statistics to monitor and display in real-time during the test.

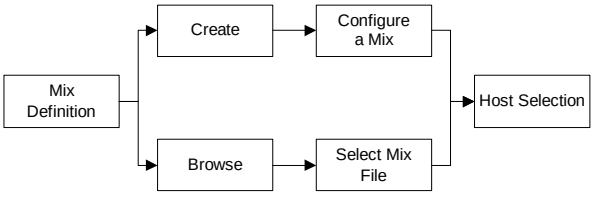
   The Performance Measurements Manager opens. For more information on configuring the statistics to monitor, see Performance Measurements Manager (on page 365).

5. Click Next.

   The Cruise Control wizard progresses to the Host Selection screen. See Selecting Host Computers (on page 172).

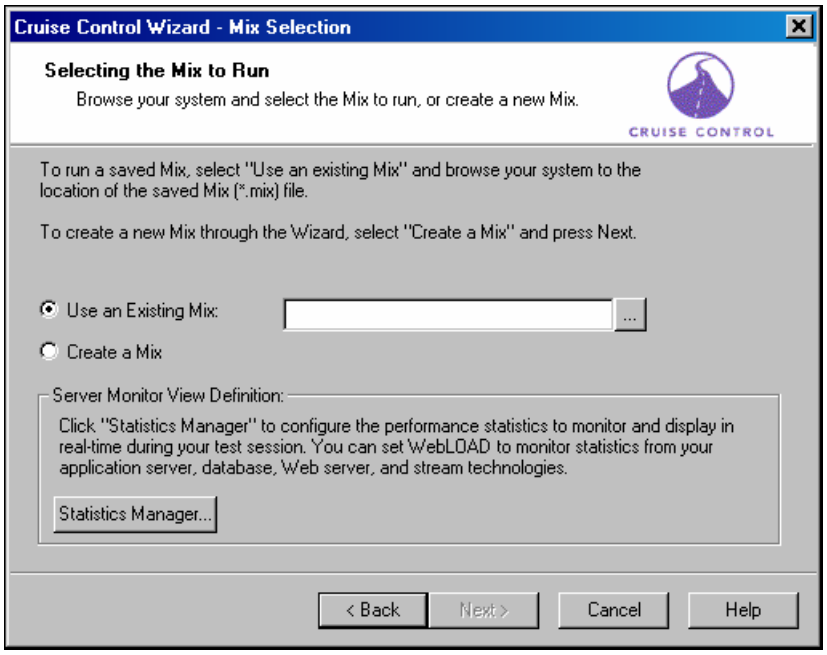
Running a Mix of Agendas

Using the Cruise Control wizard you can run your test using a saved mix or create a new mix of Agendas through the wizard. The following illustration shows the process for selecting a mix of Agendas to be used in a Cruise Control test.



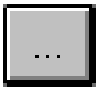
Using Mix you can prepare a combination of Agendas, each performing a specific activity. Then you can tune your testing by changing the combination of Agendas, and changing the weight of each Agenda in the Mix.

Combining Agendas through a Mix can simulate different groups of users performing different activities on the ABT at the same time. Using a Mix is another way you can create test scenarios emulating real-life behavior.



To run a mix of Agendas:

1. Select the source for the Agenda Mix to be used for this test from the Mix Selection dialog box. You can either:

   - Select Use an Existing Mix and click the browse button  to select a *.mix file from your system.
   - Select Create a Mix to configure a new mix.

2. (Optional) Open the Performance Measurements Manager and configure WebLOAD to collect performance statistics from your application server, database, Web server and stream technologies while the test is running. Click Measurements Manager to configure the performance statistics to monitor and display in real-time during the test.

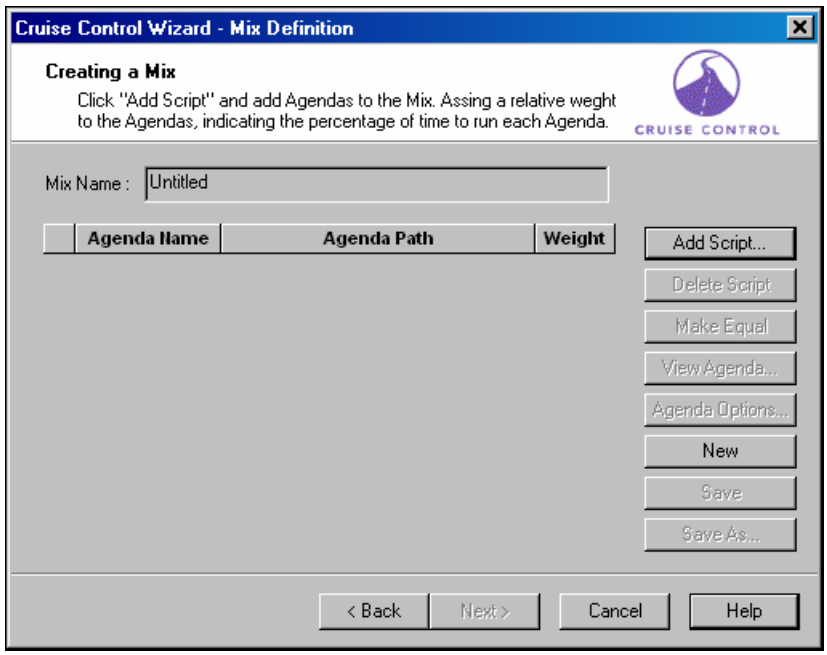
   The Performance Measurements Manager opens. For more information on configuring the statistics to monitor, see Performance Measurements Manager (on page 365).

3. Click Next.

   WebLOAD progresses to the Mix Definition dialog box. See Defining a Mix (on page 169).

Defining a Mix

A Mix is a set of Agendas, each performing different activity, to simulate groups of users performing different activities on the ABT at the same time. If you selected Use an Existing Mix on the Agenda / Mix Selection screen, the selected mix appears on the screen, enabling you to modify the Mix. You can also create a new Mix.



Creating a Mix

To create a mix:

1. Click Add Script.

   The Open dialog box appears, allowing you to select the Agenda to add to the mix.

2. Select an Agenda from your file system and click Open.

   The Agenda is added to the mix.

3. (Optional) Select the Agenda and click Options to customize the settings (such as the browser to emulate, connection settings and playback sleep options) for only the current Agenda without effecting settings in other Agendas.

   The Agenda Options dialog box opens. For information on setting the Agenda Options, see Setting Agenda Options (on page 221).

4. Repeat steps 1-2 to add additional Agendas to the mix.
5. In the Weight box, type or select the relative weight for the Agenda. Click Make Equal to assign equal weight to all of the Agendas in the mix.
6. Click Save or Save As.
7. In the File name field, type a descriptive name for the mix you have just defined and click Save.

   The mix configuration is saved with the extension *.mix.
8. Click Next.

   WebLOAD progresses to the Host Selection screen. See Selecting Host Computers (on page 172).

Adding an Agenda to a Mix

You can add Agendas to the mix to simulate different groups of users performing different activities on the ABT at the same time.

To add an Agenda to a mix:

1. Click Add Script.

   The Browse dialog box appears.
2. Select an Agenda from your file system.

   The Agenda is added to the mix.
3. In the Weight box, type or select the relative weight for the Agenda.

Deleting an Agenda from a Mix

To delete an Agenda from a mix:

1. Select the Agenda you want to delete from the mix.
2. Click Delete Script.

Assigning Equal Weight to all Agendas in the Mix

To assign equal weight to all Agendas in the mix:

- Click Make Equal.

Creating a New Mix

To create a new mix:

- Click New.

  All of the Agendas and values listed on the Mix Definition dialog box are deleted.

Saving a Mix

After defining your mix you must save it as a *.mix file.

To save a mix:

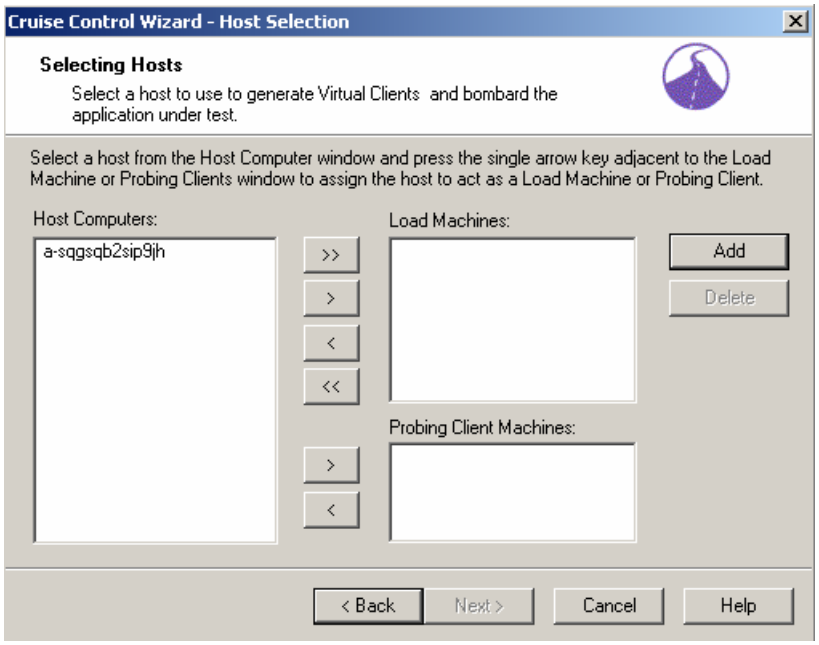
1. Click Save or Save As.
2. In the File name field in the Save As dialog box, type a descriptive name for the Agenda and then click Save.

   The mix configuration is saved with the extension .mix.
3. Click Next.

   WebLOAD displays the Host Selection dialog box. See Selecting Host Computers (on page 172).

Selecting Host Computers

After selecting the Agenda to use for the test, WebLOAD displays the Host Selection dialog box.



Hosts are the computer systems participating in the test session. In this dialog box you define the hosts to participate in the Cruise Control test. Hosts can act as Load Machines or Probing Client Machines.

Load Machines:

By selecting a list of Load Machines, you define a pool of Hosts that WebLOAD uses to generate a load until the Cruise Control goal is reached. To reach your defined goal, WebLOAD may need to generate a large number of users accessing the ABT. The number of users that can be generated on each host depends on the power of each host machine. Defining many machines as Load Machines allows for the simulation of more users throughout the Cruise Control test.

Probing Client Machines:

WebLOAD reports exact measurements for single Probing Clients. By defining Probing Client Machines on different hosts, you will receive exact performance measurements for that client, rather than averages calculated for users generated on Load Generators. By defining a list of Probing Client Machines, you can receive accurate measurements at different locations (depending on the location of the host). Therefore, it is recommended that you define at least one Probing Client Machine in a Cruise Control Session, see Defining Performance Goal (see "Defining Performance Goals" on page 175).

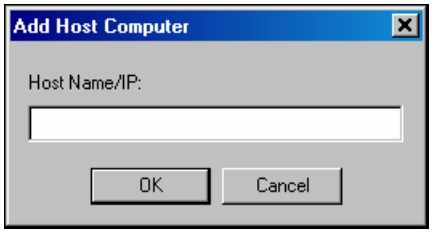
Note: In Standalone Workstation mode only your local system is available as the Load Machine.

Adding Host Computers to the Host Computers List

To add a host computer to the host computers list:

1. In the Host Selection dialog box, click Add.

   The Add Host Computer dialog box displays.



2. Insert the host computer name or IP address in the Host Name / IP field.
3. Click OK.

   The host is added to the Host Computers dialog box.

For defining Host Computers as Load Machines and Probing Clients, see Designating a Host Computer as a Load Machine (on page 173) and Designating a Host Computer as a Probing Client Machine (on page 174).

Designating a Host Computer as a Load Machine

To designate a host computer as a load machine:

1. Select a host from the Host Computer window.

   To add hosts not yet defined to the Host Computer window, see Adding Host Computers to the Host Computers List (on page 173).

2. Press the single arrow key adjacent to the Load Machine or Probing Clients window.

   The selected host moves to the Load Machine or Probing Clients window.

3. Repeat steps 1 and 2 to define all of the hosts participating in the test.
4. Click Next.
5. After configuring the hosts participating in the Cruise Control test, Cruise Control prompts you for the goals you want to achieve.

   The Goal Definition dialog box appears, see Defining Performance Goal (see "Defining Performance Goals" on page 175).

Note: If you define a Probing Client host, the Probing Client Agenda Selection dialog box displays.

Designating all Host Computers as Load Machines

To designate all Host Computers as load machines:

1. Press the double arrow key adjacent to the Load Machine window.

   All of the host computers move to the Load Machine window.

Note: At least one system must be configured as a Load Machine to continue working with the wizard.

Designating a Host Computer as a Probing Client Machine

To designate a host computer as a Probing Client Machine:

1. Select the host from the Host Computer window.
2. Press the single arrow key adjacent to the Probing Client Machines window.

   The selected host moves to the Probing Client Machine window.

Deleting a Host Computer

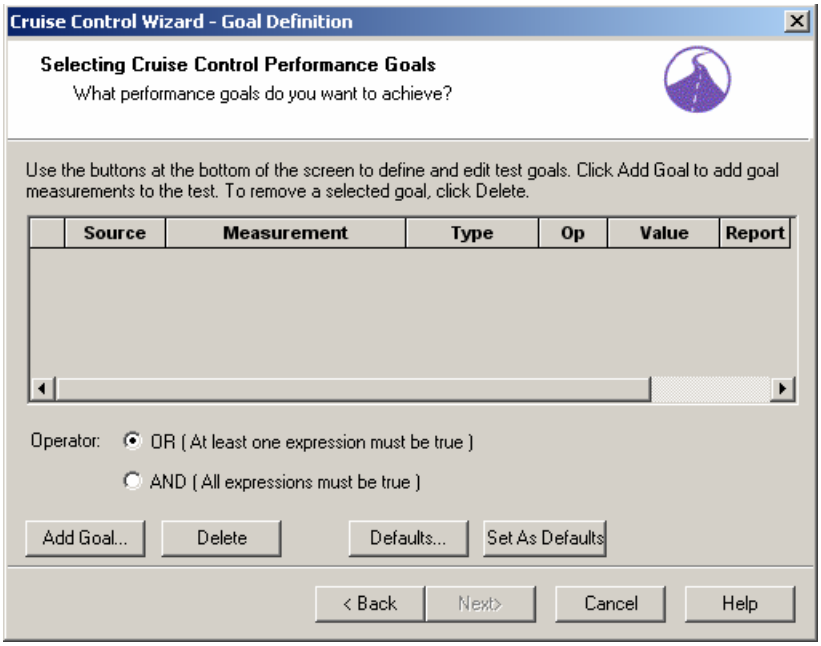
To delete a host computer:

1. Select the host name from the Host Computer list.
2. Click Delete.

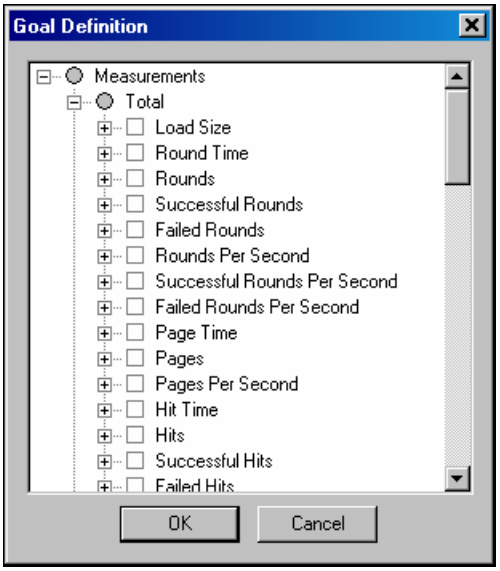
   The host is deleted from the Host Computer list.

Defining Performance Goals

After defining the host computers participating in your test, the Cruise Control Wizard prompts you for the performance goals you want to achieve in your test session.



About the Goal Definition Dialog Box



The Goal Definition dialog box enables you to define the performance goals for the Cruise Control test. The defined goals are displayed at the top of the dialog box where you can see at a glance all of the participants in the test, the measurements being tested and the goals to be achieved.

On the bottom of the dialog box define the (OR/AND) conditions of the goals where you specify whether at least one or all of the goals must be met. The buttons on the lower half of the dialog box are used for defining and editing these goals.

The Goal Table

The goal table in the center of the dialog box contains the following fields:

Field Name	Description
Source	The source you are tracking to determine when your goal is reached. The source can be either: ◆ A Probing Client Machine ◆ A Load Machine The source for Load Machines is "Total" because your results are the average of all Virtual Clients. The Source for Probing Client Machines is the name of the Probing Client.
Measurement	The measurement or activity that you are testing such as: ◆ Response Time ◆ Hits Per Second ◆ Round Time
Operator	The operator is the condition of the test. This may be either >= or <= depending on the measurement being tracked. Here you indicate whether the results should be greater than, or less than the desired value for the goal to be reached. For example, if the measurement being tracked were Response Time, you would choose the operator >= to achieve the goal when the time needed to receive a response from the server exceeds the limit for this goal.
Value	The limit that determines when this goal is achieved.
Report	When this field is checked, the default Integrated Report includes the measurement for which the performance goal is defined. The Report field is selected by default.

Operator

Below the Goal Table are the Operator radio buttons. When more than one goal is defined, the Operator enables you to establish the conditions under which the goal will be achieved.

- Select the OR radio button to require only one expression to be true for the goal to be achieved.
- Select the AND radio button to require all expressions to be true for the goal to be achieved.

Control Buttons

Below the Operator radio buttons, find the following control buttons:

Button	Description
Add Goal	Open the Add Goal dialog box to add a new measurement to the test goals.
Delete	Delete a test goal.
Defaults	Open the Default Goals dialog box. For more information on using Defaults, see Setting and Using Defaults (on page 180).
Set As Defaults	Set the current goals as default measurements and save them for future use.

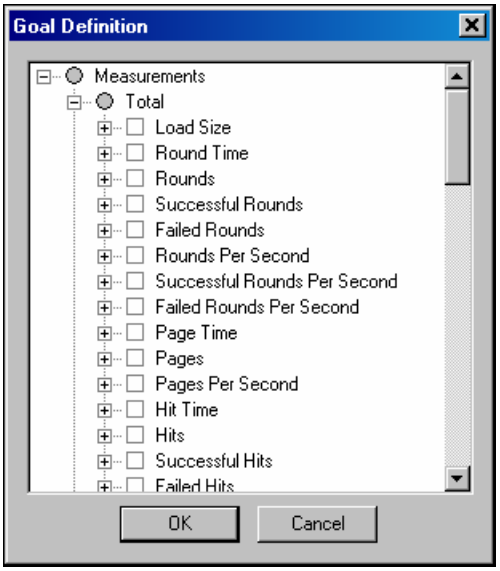
Adding Performance Goals

Goals are configured by defining boundary values for selected WebLOAD measurements. Goals define the performance level you expect from the ABT. WebLOAD increases the load and runs the test until the defined goals are reached.

To add performance goals:

1. Click the Add Goal button to add measurements on which to base your goals.

   The Goal Definition dialog box opens.



   All of the WebLOAD measurements are displayed in a tree structure.

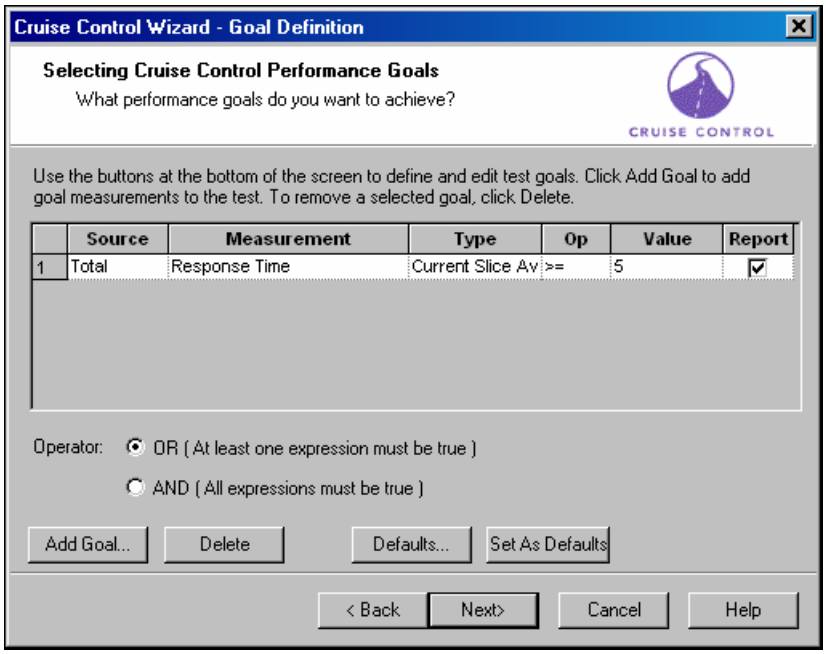
2. Select the checkbox adjacent to the measurements to include in the goal definition.

   For example to define a test goal for the response time not to exceed 5 seconds, select the checkbox adjacent to Response Time.

Note: A gray circle next to a measurement indicates that no default subcomponents are defined (the upper level component cannot be selected). To select subcomponents, click on the + to expand the tree.

3. Click OK.

   WebLOAD closes the list and opens a row for each of the selected measurements in the Goal Definition dialog box.



4. Click in the Op field and select an operator from the drop-down list.
5. Click in the Value field and type a boundary value for the goal.
6. Ensure that the Report field is checked to include the goal measurement in the Cruise Control default Integrated Report.
7. Click Next.

   WebLOAD displays the Acceleration dialog box. See Defining the Acceleration Rate (on page 182).

Configuring a Goal - An Example

Suppose your goal is for your Web application server Response Time not to exceed 5 seconds.

To configure the goal:

1. Click the Add Goal button.

   The Goal Definition window is displayed.

2. Select Response Time from the list of measurements and click OK.

   This measurement is displayed in the table on the Goal Definition dialog box.

3. Because the goal is for the Response Time not to exceed 5 seconds, select the operator >= and the Goal Value 5 by modifying the Op and Goal Value cells.

   When you run the test, WebLOAD increases the load on the ABT until the Response Time for each simulated user reaches or exceeds 5 seconds and then it stops the test.

Deleting a Goal

To delete a goal:

1. Select the goal to be deleted on the Goal Definition screen by clicking on the number to the left of the goal.
2. Click Delete.

   The goal is removed from the Goal Definition dialog box.

Setting and Using Defaults

The Cruise Control Wizard enables you to save goals that you have defined, and reuse them in subsequent tests. To do this, set your defined goals as the default, and then use the defaults in later tests.

Setting Defaults

The Set As Default button sets all of the Goals located on the Goal Definition dialog box as defaults. This operation overrides any previous default configurations.

To set defaults:

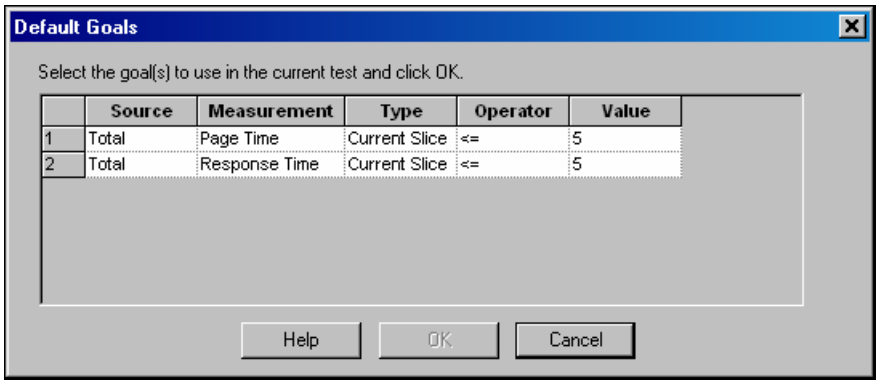
1. Configure the test goals that you want to save as the default.
2. Click Set As Defaults.

Using Defaults

To use defaults:

1. On the Goal Definition dialog box, click Default.

   The Goal Defaults dialog box appears.

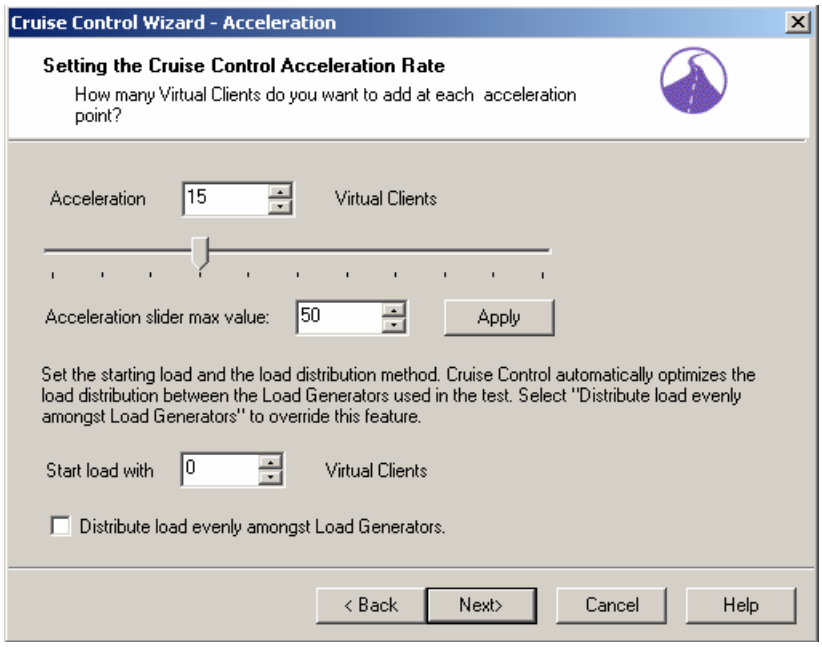


2. Select the goal to be included in the Cruise Control test session by selecting the number to the left of the goal (use the Ctrl or Shift keys to select more than one goal).
3. Click OK.

   The goal(s) are added to the test definition.

Defining the Acceleration Rate

Cruise Control increases the number of Virtual Clients accessing the application until the performance goals are met. The acceleration rate is the rate at which WebLOAD increases the number of Virtual Clients accessing the ABT at each acceleration step. Use a large number of clients for a fast acceleration to reach your goals quickly. Set WebLOAD to increment the number of Virtual Clients more slowly for fine-tuned results.



To define the acceleration rate:

1. Click on the slider and move up and down the scale to select an acceleration rate.

   We recommend:

   - Accelerating at a rate of 15 Virtual Clients to reach your goals quickly.
   - Accelerating at a rate of 5 Virtual Clients for fine tuned results.

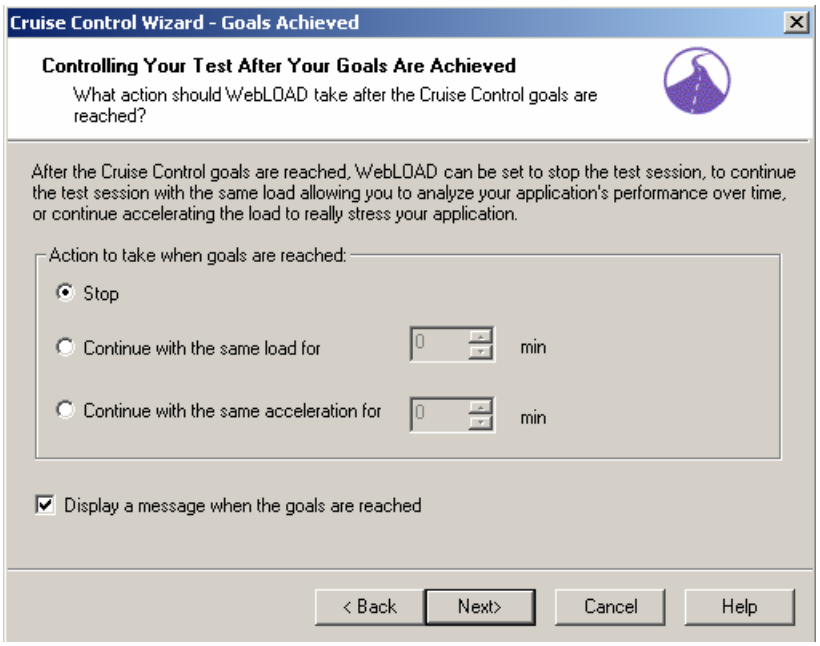
2. Type or select the initial load to begin the Cruise Control session in the Start load with Virtual Clients box.
3. Cruise Control automatically optimizes the load distribution of Virtual Clients among the Load Generators participating in the test. To override this feature and distribute the load evenly between the Load Generators select the Distribute load evenly amongst Load Generators checkbox.
4. Click Next.

   The wizard progresses to the Goals Achieved dialog box.

Note: The number of users that WebLOAD can generate on any given Load Machine in an efficient manner (so that the addition of users will not reduce the performance on that machine) is dependent on the power of the host machine. When WebLOAD estimates that it has reached the maximum output on a specific host, a log message is sent. WebLOAD continues to accelerate the load until the end of the Cruise Control test session.

Defining WebLOAD's Action Once the Goals are Reached

After configuring your Cruise Control test you must define WebLOAD's response once the test goals are reached.



Cruise Control can be set to stop, to continue testing under the same load, or to continue accelerating the number of Virtual Clients.

To set WebLOAD to stop:

1. Select Stop.
2. Click Next.

   The wizard progresses to the Finish dialog box.

To set WebLOAD to continue running with the same load:

1. Select Continue with the same load.
2. Type or select the amount of time (in minutes) WebLOAD should continue running with the same load after the test goals are reached.
3. Click Next.

   The wizard progresses to the Finish dialog box.

To set WebLOAD to continue accelerating the number of Virtual Clients:

1. Select Continue with the same acceleration.
2. Type or select the amount of time (in minutes) WebLOAD should continue accelerating the load after the test goals are reached.
3. Click Next.

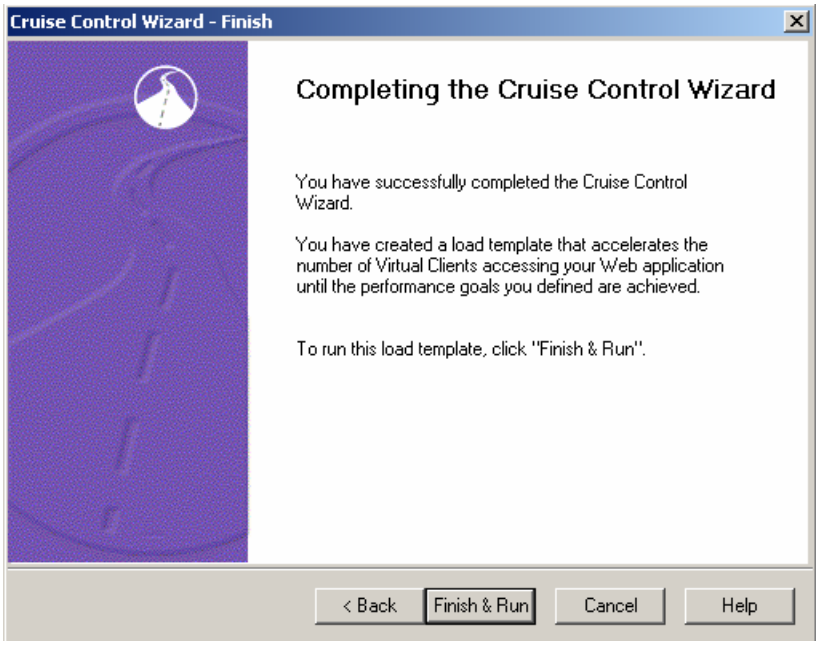
   The wizard progresses to the Finish dialog box.

Finishing the Cruise Control Wizard

The product of your testing is the WebLOAD report, where you can see how your Web application handles the load that you tested. The Cruise Control Wizard automatically generates a predefined report that measures:

- Load Size (the number of Virtual Clients being generated)
- The WebLOAD measurements for which goals are defined.

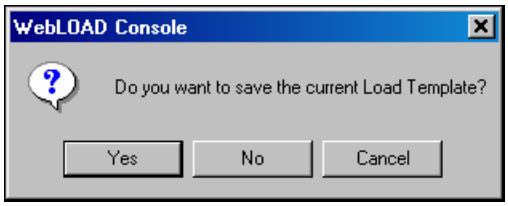
WebLOAD saves all of the values for the complete set of measurements. You can decide after the test has run to display the results never before requested.



To finish the Cruise Control Wizard:

1. From the Finish dialog box, click Finish & Run.

   A message box appears asking if you want to save the current load template.



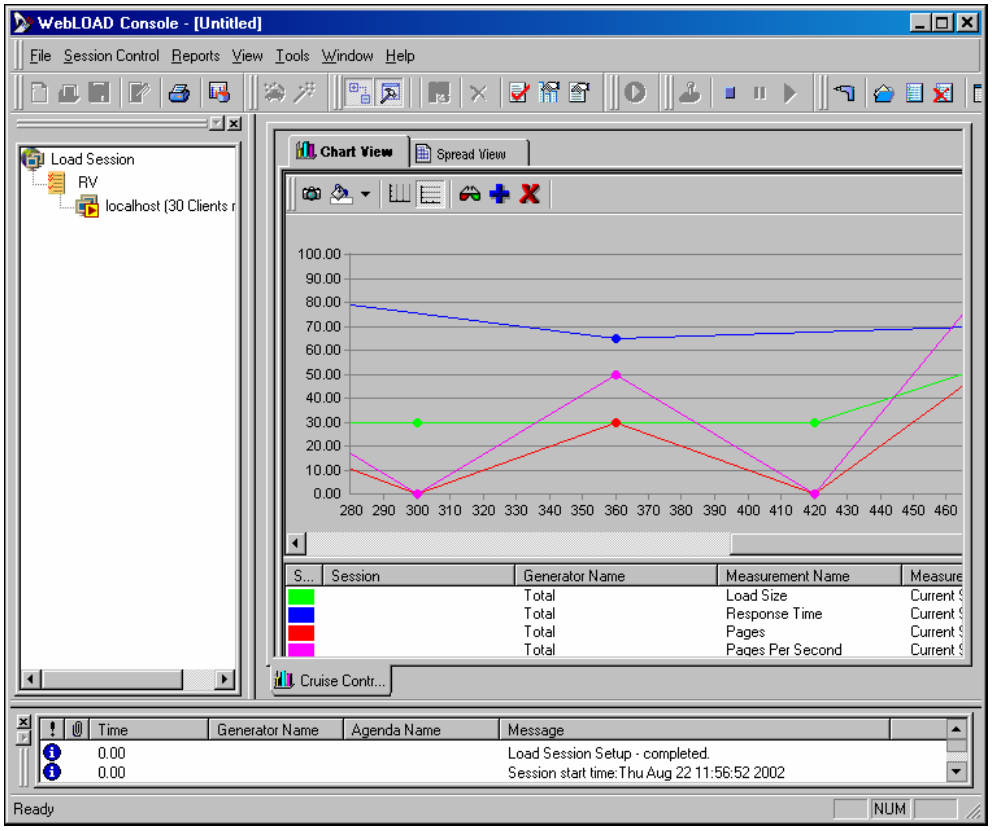
2. Click Yes to save the current load template as a *.tpl file.

   Load templates contain the parameters that you defined for the test, such as the participating hosts in a test and the test plan. Using load templates saves you the time of reconfiguring the load session when repeating tests.

   The Cruise Control wizard is finished and the test begins. WebLOAD displays the Cruise Control Default Report.

Viewing the Cruise Control Default Report

When running a Cruise Control test session, the WebLOAD Cruise Control Default Report opens automatically in Chart View.



Real-time results are displayed in graphs on the right side of the Console screen.

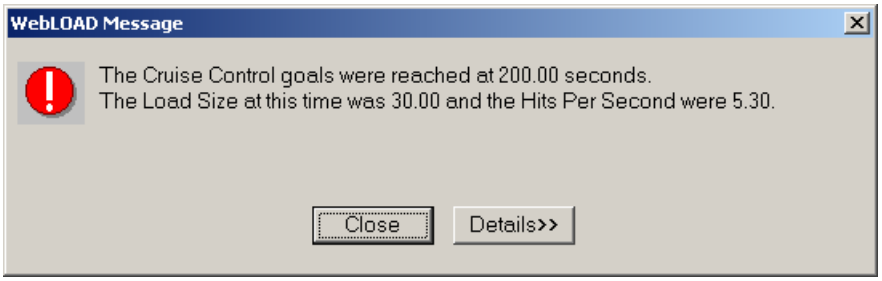
The following elements are displayed on the screen:

Elements	Description
Menu Bar	Contains the menus for configuring and controlling the Cruise Control test session.
Toolbar	Contains buttons for controlling the Cruise Control test session.
Gauges	Display the Load Size and Rounds Per Second during the Cruise Control test session.
Results Window	All of the reports opened during the Cruise Control test session will appear in this window. Use the tabs located at the top and bottom of the window to view different reports.

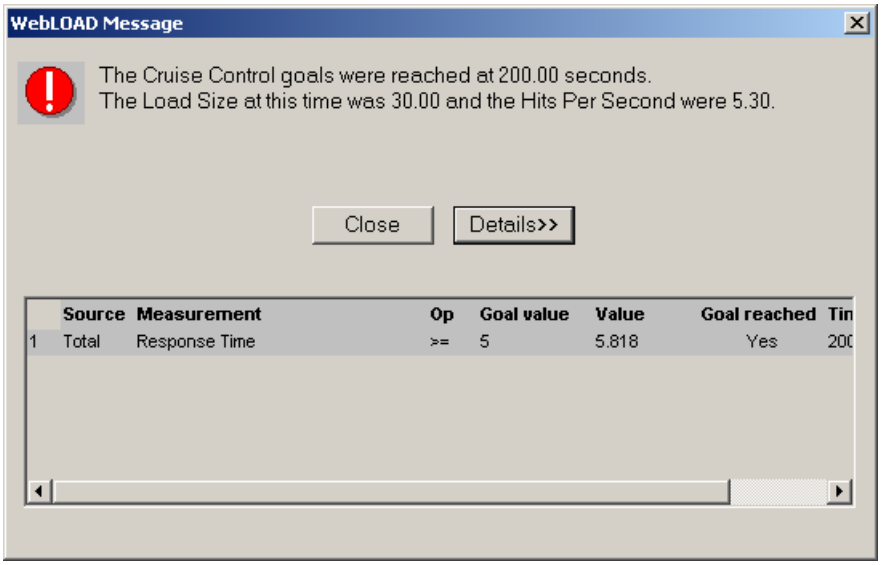
Elements	Description
Session Tree	Presents a graphical display of your Cruise Control test session.
Status Bar	Indicates the program status. When running with the Resource Manager the Status Bar displays the current mode, Standalone Workstation mode or Connected Workstation mode.

The Cruise Control Session continues running until the performance goals are achieved. When the performance goals are reached, WebLOAD displays a message box with the test results.

The following is an example of such a message box:



Click Details to examine the original goal settings and statistics for them. A complete report displayed as follows:



Note: You can view this goals message box while the test is running by right clicking on the load session icon in the session tree.

C H A P T E R 1 3

Configuring a Load Template Manually

This chapter describes the process of manually configuring a load template. Load templates are configured in the Console through a sequential series of dialog boxes. Through these dialog boxes, you can select and schedule the Agenda or Mix files to run during the test, define the number of Load Machines and Probing Clients to run each Agenda and the number of Virtual Clients to run.

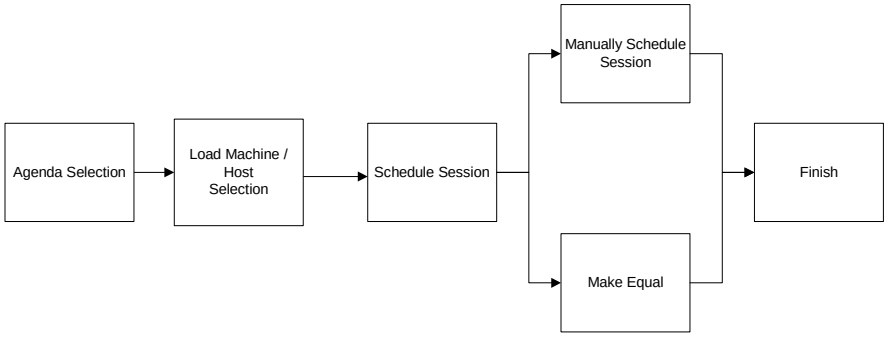
Note: You cannot create load templates if you have not yet recorded (or otherwise created) any Agendas. If you do not have any Agendas to work with, you must create an Agenda using one of the authoring tools, such as WebLOAD IDE.

In This Chapter

The Manual Test Configuration Workflow 190
Starting a New Template Manually 190
Selecting an Agenda 191
Selecting Hosts 193
Scheduling the Test Session 195
Completing Your Manual Load Template Configuration..... 198

The Manual Test Configuration Workflow

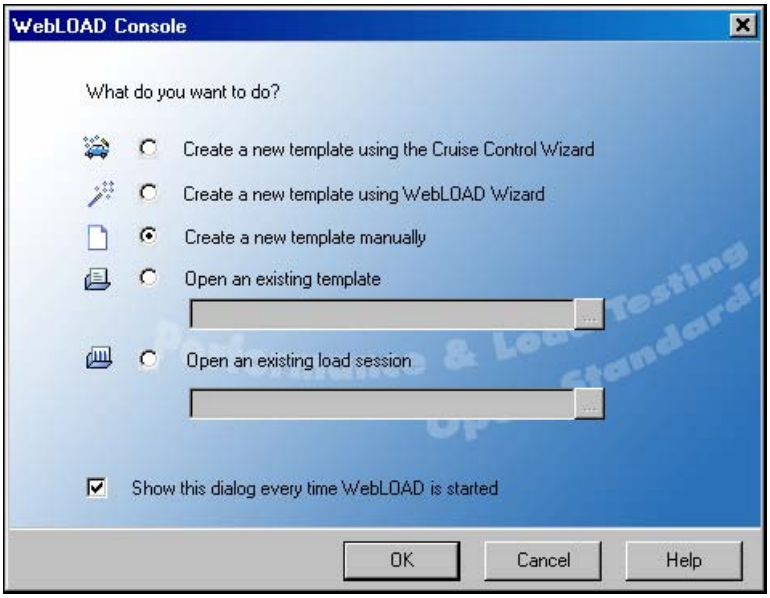
The following diagram illustrates the workflow for manually configuring a load test.



Starting a New Template Manually

To create a new load template manually:

1. Select the Create a new template manually option on the WebLOAD Console dialog box.



2. Click OK.

   The WebLOAD Console displays.

3. Select Session Control | Modify Agenda Selection,

   -Or-

   Right-click on the Load Session item in the Agenda tree and select Modify Agenda Selection.

   The Agenda/Mix Selection dialog box opens. See Selecting an Agenda (on page 191).

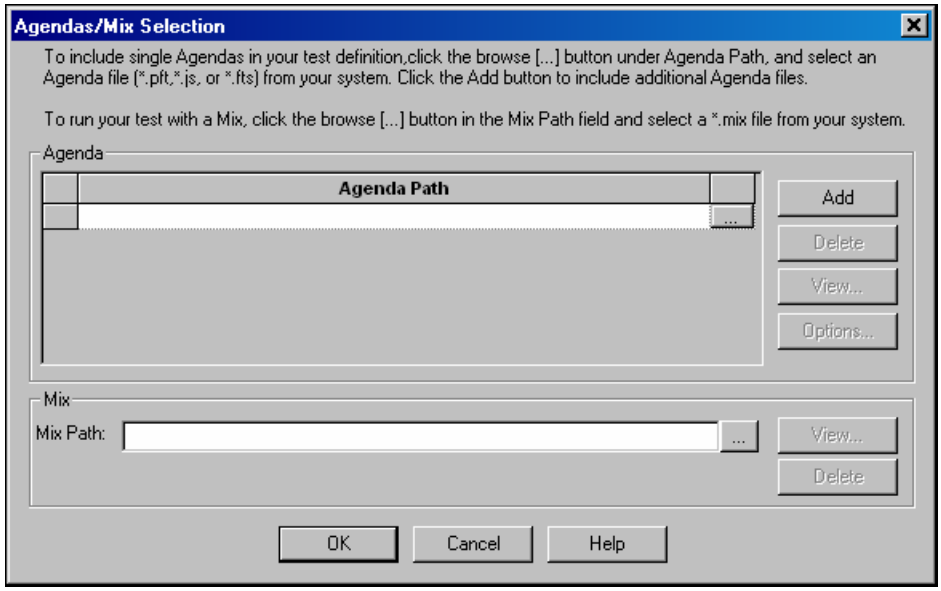
Selecting an Agenda

Agendas are test scripts that describe user activity and are run during WebLOAD test sessions.

WebLOAD supports two types of Agendas:

- Single Agendas, which are lone test-scripts.
- Mix of Agendas, which are a combination of existing single Agendas. Using a mix you can simulate different groups of users performing different activities on the application being tested (ABT) at the same time.

Select the Agendas to run through the Agenda / Mix Selection dialog box.



Creating Load Templates with One or More Single Agendas

To create load templates with one or more single Agendas:

1. Click in the Agenda Path field and enter the name and the full path for the Agenda.

   The path name can be entered manually, or through browsing the directory structure.

2. Click Add and repeat step 1 to include additional Agendas.
3. (Optional) To open and view or edit the selected Agenda, press View.

   WebLOAD IDE opens displaying the selected Agenda.
4. (Optional) To configure runtime options exclusive to this Agenda, press Options.

   The Agenda Options dialog box opens enabling you to define the runtime options for the selected Agenda. For more information on Agenda options, see Setting Agenda Options (on page 221).
5. Click Ok.

   The Agendas selected to run in the test are displayed in the Session Tree.

Creating Load Templates with an Agenda Mix

To create load templates with an agenda mix:

1. Click in the Mix Path field and enter the name and path of the Mix file (*.mix) to run.

   The path name can be entered manually, or through browsing the directory structure.
2. To add additional Agendas to be run with the Mix:

   a. Click in the Agenda Path field and enter the name and the full path for the Agenda.
   The path name can be entered manually, or through browsing the directory structure.

   b. Click Add and repeat step 2a to include additional Agendas.
3. (Optional) To open and view or edit the selected Agenda, press View.

   WebFT opens displaying the selected Agenda.
4. (Optional) To configure runtime options exclusive to this Agenda, press Options.

   The Agenda Options dialog box opens enabling you to define the runtime options for the selected Agenda. For more information on Agenda options, see Setting Agenda Options *(on page 221)*.
5. Click Ok.

   The Mix and Agendas selected to run in the test are displayed in the Session Tree.

Selecting Hosts

Hosts are the computer systems participating in the test session. Each host has an assigned task, acting as either a Load Machine or a Probing Client. Both Load Machines and Probing Clients are configured in the same Host Selection window. Hosts must be defined for each Agenda or Mix file running in the test session.

Opening the Host Selection Dialog Box

To open the host selection dialog box:

- Right click on the Agenda or Mix icon in the Session Tree and select Modify Load Machine Selection,

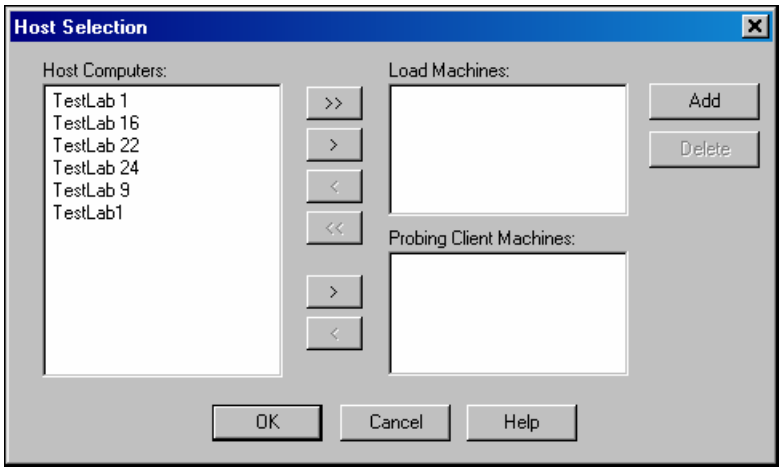
  -Or-

  Click on the Host Selection icon in the Console toolbar,

  -Or-

  Select Session Control | Modify Host Selection.

  The Host Selection dialog box opens.



In this dialog box, you define the host computers participating in the test. There are three windows in this dialog box. The large window on the left, entitled Host Computers, lists all of the host computers available for testing. The two windows to the right contain the systems defined as Load Machines and Probing Client Machines. Use the arrow buttons between the windows to change the roles of the host computers.

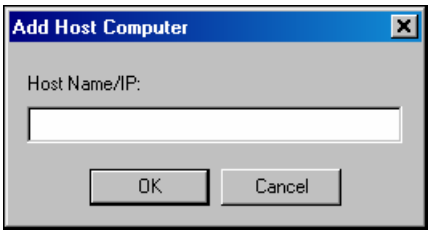
Note: In Standalone Workstation mode only your local machine is available to act as a Load Machine.

Defining Host Computers

To define a host computer:

1. Click Add.

   The Add Host Computer dialog box opens.



2. Enter the name or IP address of the host computer you want to add.
3. Click OK.

   WebLOAD adds the machine to the Host Computer list.

Defining Load Machines

Load Machines are the machines that generate the Virtual Clients and bombard the application being tested.

Designating a Host Computer as a Load Machine

To designate a host computer as a load machine:

1. Select the Host from the Host Computer window.
2. Press the single arrow key adjacent to the Load Machine window.

   The selected Host moves to the Load Machine window.
3. Click OK.

Designating all Host Computers as Load Machines

To designate all host computers as load machines:

1. Press the double arrow key adjacent to the Load Machine window.

   All of the Host Computers move to the Load Machine window.
2. Click OK.

Defining Probing Client Machines

Probing Clients act as single Virtual Clients to measure the performance of targeted activities and provide individual performance statistics of the ABT.

Designating a Host Computer as a Probing Client Machine

To designate a host computer as a Probing Client Machine:

1. Select the Host from the Host Computer window.
2. Press the single arrow key adjacent to the Probing Client Machines window.

   The selected Host moves to the Probing Client Machine window.
3. Click OK.

Deleting a Host Computer

To delete a host computer:

1. Select the host name from the Host Computer list.
2. Click Delete.

   The host is deleted from the Host Computer list.

Scheduling the Test Session

The Schedule dialog box is used to specify the test parameters. Here you specify the size of the load to be generated on each Load Machine throughout the test.

You can create a load schedule by:

- Manually defining time frames and the load to generate over each frame.
- Using the Load Profiler.
- A combination of both.

Creating a Load Schedule Manually

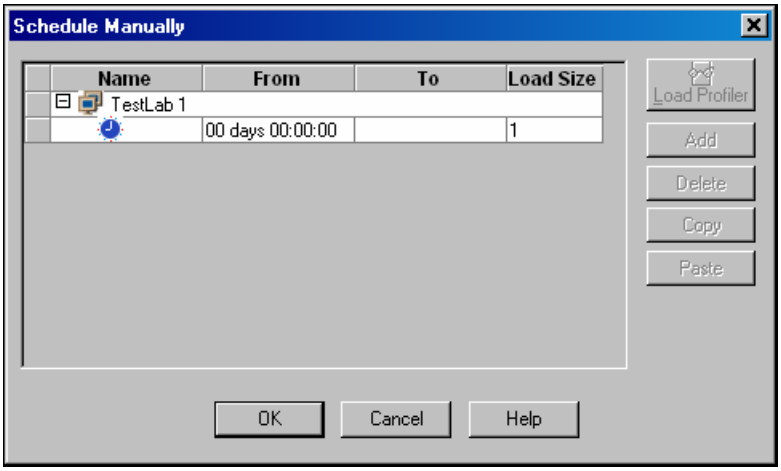
To create a load schedule manually:

1. Double click on a host in the Session Tree,

   -Or-

   Right click on the Agenda the Load Machines are running and select Modify Load Machine Schedule.

   The Schedule Manually dialog box opens.



2. Expand the Load Machine tree using the + / - button adjacent to the Load Machine you want to schedule.
3. Enter the number of days and the starting time to generate the load in the From field.

   The starting time is defined for each particular Agenda relative to the beginning of the test.

   Note: Days are numbered 00 through to 99. Seconds are numbered 1 through 59. To enter one minute, use the 1:00 syntax not 00:60.

4. Enter the number of days and the ending time to generate the load in the To field.

   The ending time is defined for each particular Agenda relative to the beginning of the test. If the To field is left blank, the WebLOAD continues generating the load until the From time in the next entry for this Load Machine, or if this is the last entry, until the test is stopped manually.

5. Enter the number of Virtual Clients to generate during the time frame defined in steps 3 and 4.
6. Schedule additional Virtual Client loads for this Load Machine:
   a. Click Add to add a blank line to the schedule grid.
   b. Repeat steps 3 through 5.
7. Repeat steps 3 through 6 for each Load Machine or Probing Client defined in the Session Tree.
8. Click OK.

Creating a Load Schedule with the Load Profiler

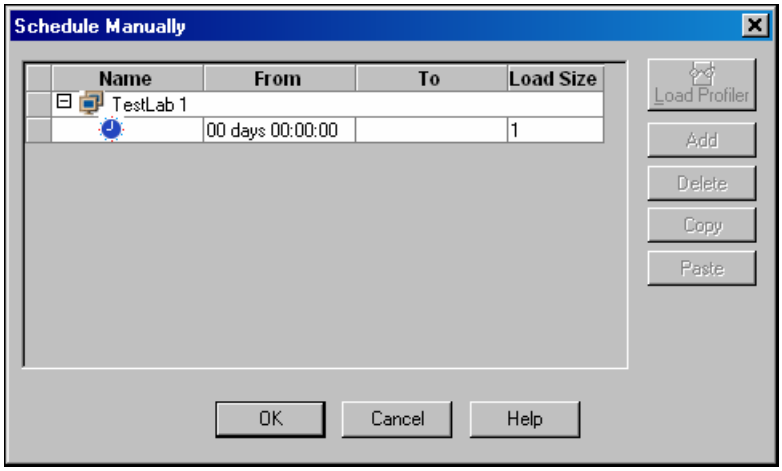
To create a load schedule with the load profile:

1. Double click on a host in the Session Tree,

   -Or-

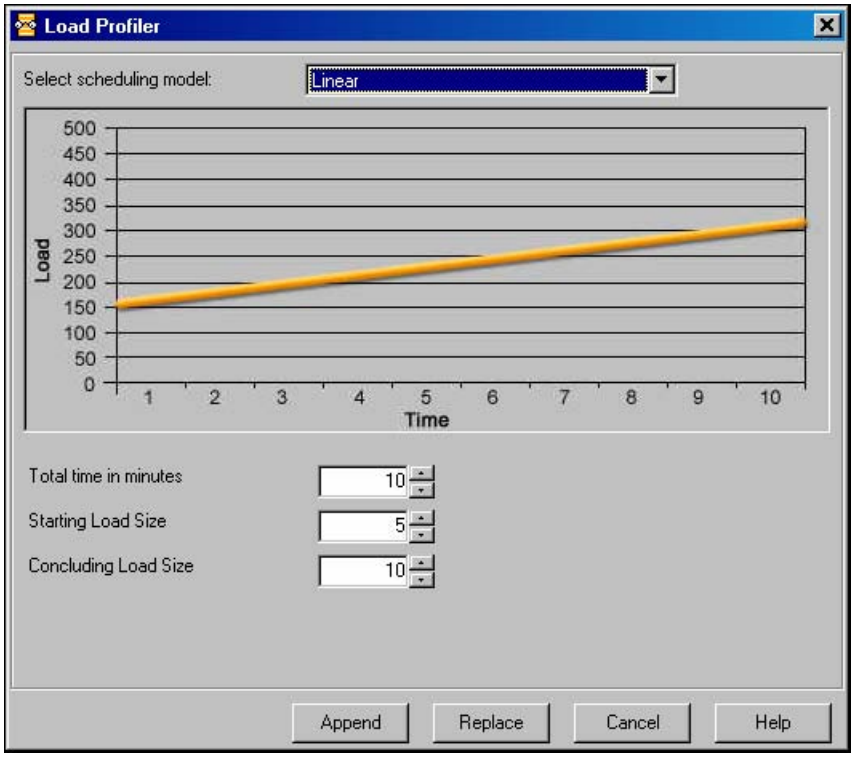
   Right click on the Agenda the hosts are running and select Modify Load Machine Schedule.

   The Schedule Manually dialog box opens.



2. Click on the Load Machine you want to schedule.
3. Click on the Load Profiler button.

   The Load Profiler opens.



4. Select a pattern from the scheduling model field at the top of the dialog box.
5. For an explanation of each model, see Scheduling Models (on page 150).
6. Fill in the fields at the bottom of the dialog box.
7. Click Append to include the load definition in the schedule,

   -Or-

   Click Replace to substitute the load definition for the one appearing in the Schedule Manually dialog box.

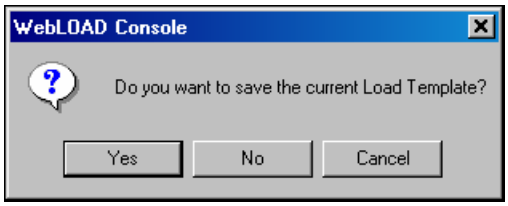
Completing Your Manual Load Template Configuration

After configuring the Agendas to run, the Load Machines to generate the load, and the test schedule, you are ready to run your load session.

To complete your manual load template configuration:

1. Click the start icon on the Console toolbar.

   A message box appears asking if you want to save the current load template.



2. Click Yes to save the current load template as a *.tpl file.

   Load templates contain the parameters that you defined for the test, such as the Agendas to run, participating hosts, and the test plan. Saving load templates saves you the time of reconfiguring the load session when repeating tests.

   For more information on running tests, see Running a Load Session (on page 275).

C H A P T E R 1 4

Defining Test Session Options

In This Chapter

Session Options..... 201

Session Options

The session option material shows you how define test session options and introduces you to some of WebLOAD's more advanced features. This material is divided into three chapters:

- Setting Global Options (on page 203) guides you through the process of setting WebLOAD global options.
- Setting Agenda Options (on page 221) guides you through the process of setting WebLOAD default Agenda options, individual Agenda options and session Agenda options.
- Functional Verification Testing (on page 259) details how to set WebLOAD to perform Functional Verification tests on your Web application.

C H A P T E R 1 5

Setting Global Options

Global Options enable you to modify WebLOAD defaults to your specifications. All WebLOAD tests use these defaults.

In This Chapter

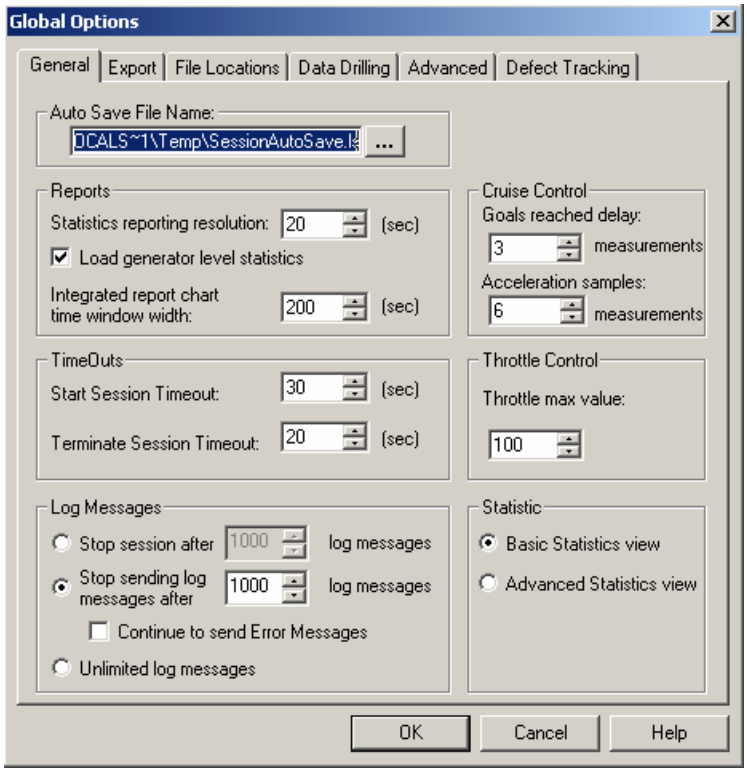
How to Set Global Options 203
The Notification Manager 215

How to Set Global Options

Global options can be modified at any time from the WebLOAD Console.

1. Select Tools | Global Options.

   The Global Options dialog box opens.



   The Global Options dialog box contains the following tabs:

   - General
   - Export
   - File Locations
   - Data Drilling
   - Advanced
   - Defect Tracking

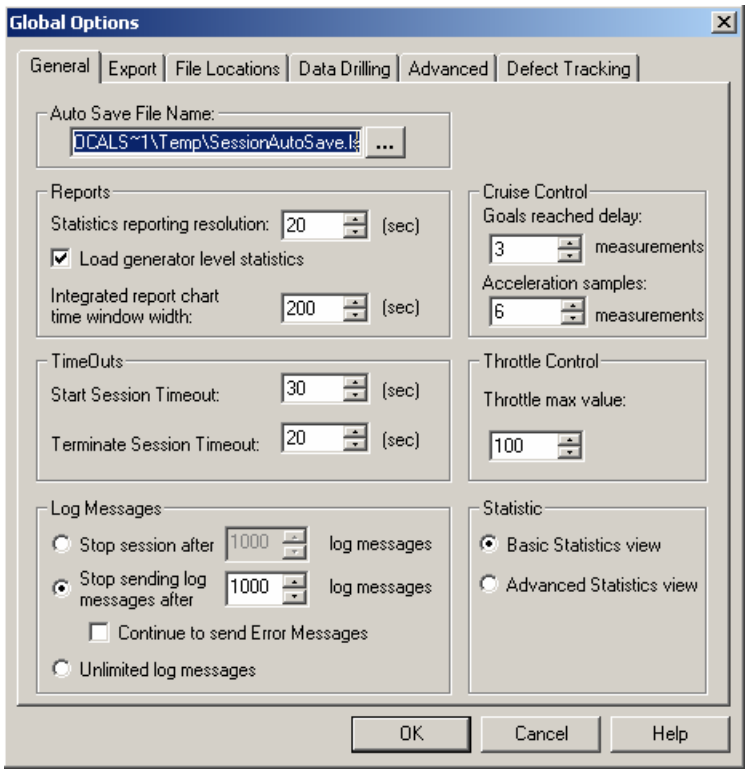
2. Select the desired tab.
3. Modify as many of the options you need.
4. Select another tab and modify its options.
5. Click OK to accept the modifications, or Cancel to reject them, and exit the Global Options dialog box.

The General Tab

The General tab lists a number of options that you can modify to facilitate your work in test sessions.

To access the General tab options:

1. Select Tools | Global Options.
2. Select the General (default) tab.



Setting the Autosave File Name Default Location

WebLOAD performs fast saves while running a test. This option enables you to specify the filename and location of the autosave (backup) file. It is recommended to save files to a central location where they are available to all users.

To set the Autosave file name default location:

1. Select Tools | Global Options.
2. Select the General (default) tab.
3. Type or select the desired file name and location in the Auto Save File Name field.
4. Click OK.

Setting the Reports Options

The Reports options enable you to define the:

- Interval between each reported sample (referred to as the Statistics reporting resolution).
- Statistics for each Load Generator to be collected separately for display in Integrated Reports.
- Default time range to be displayed as the x-axis of the Integrated Report chart (out of the total time span of the Load Session).

To set the Reports options:

1. Select Tools | Global Options.
2. Select the General (default) tab.
3. In the Statistics reporting resolution field, type or select (in increments of 10 seconds) the desired statistics reporting resolution.

   The default reporting interval is 20 seconds. When running long test sessions, it is a good idea to increase this interval.
4. Select the Load generator level statistics checkbox (default) to display the statistics for each Load Generator separately in Integrated Reports.
5. In the Integrated Report chart time window width field, type or select the desired time range of the chart window.

   The default time width is 200.

Setting Cruise Control Options

Cruise Control increases the load generation until your Web application performance falls below the profile you have defined.

The performance parameters that you can define are:

- Goals reached delay - the number of samples required to declare the Cruise Control goals as achieved. Cruise Control declares the defined goals reached only after receiving the desired goal results in several samples.
- Acceleration point measurements - the number of sample measurements to read before automatically accelerating the load if the Cruise Control goal has not been reached by then.

To set Cruise Control options:

1. Select Tools | Global Options.
2. Select the General (default) tab.
3. In the Goals reached delay field, type or select the number of samples to read before proclaiming goals reached.
4. In the Acceleration point measurements field, type or select the number of samples to read before accelerating the load.

Setting WebLOAD Timeout Values

To set WebLOAD Timeout values:

1. Select Tools | Global Options.
2. Select the General (default) tab.
3. In the Start Session Timeout field, type or select the amount of time the Console should attempt to contact a host at the start of a test session (default: 30 seconds).
4. In the Terminate Session Timeout field, type or select the amount of time the Console should attempt to contact a host at the close of a test session (default: 20 seconds).

Setting the Throttle Control Max Value

The throttle control utility enables you to dynamically modify the number of Virtual Clients participating in a test during runtime.

To set the Throttle Control max value:

1. Select Tools | Global Options.
2. Select the General (default) tab.
3. In the Throttle max value field, type or select the maximum number of Virtual Clients that can participate in the test.

Setting the Log Message Options

Log messages generated by the Console, the JavaScript compiler and any user messages programmed in the test script and detected by WebLOAD during runtime are displayed in the Log Window. For further information on Log Messages, see Viewing the Log Window (on page 287).

To set the log message options:

- In the Stop session after x log messages field, type or select the maximum number of log messages to read before stopping WebLOAD,

  -Or-

  In the Stop sending log messages after x log messages field, type or select the maximum number of log messages to record. WebLOAD will continue the Load Session but stop recording log messages. The maximum number of log messages is 32767,

  -Or-

  Set WebLOAD to send a unlimited number of log messages.

Setting the Statistic View Settings

To set the Statistic View settings:

1. Select Tools | Global Options.
2. Select the General (default) tab.
3. Select Basic Statistics View for access to only the Current Value for each Current Slice measurement in the Integrated Report,

   -Or-

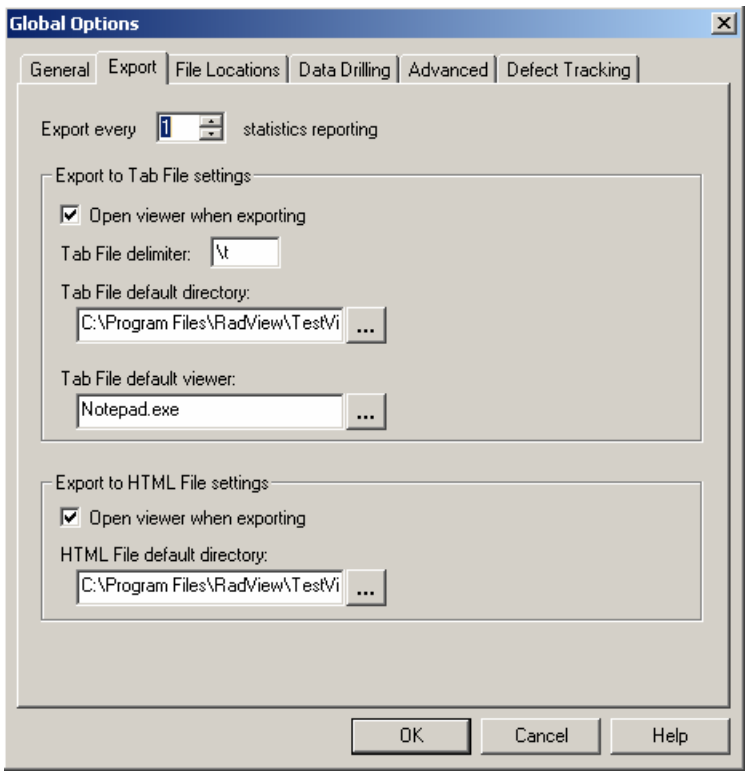
   Select Advanced Statistics View to display all values available for the current slice in the WebLOAD Integrated Reports.

The Export Tab

WebLOAD reports can be exported to other programs as Tab Files (tab-delimited text files). Export options, such as the default Tab file viewer, can be set through the Export dialog box.

To access the Export tab options:

1. Select Tools | Global Options.
2. Select the Export tab.



Setting the Statistics Resolution to Export

The Statistics exporting resolution is the frequency of the number of samples to export when exporting WebLOAD reports. For example, if "1" is entered for this value, WebLOAD exports every sample. If "2" is entered for this value, WebLOAD exports every second sample.

To set the Statistics resolution to export:

1. Select Tools | Global Options.
2. Select the Export tab.
3. Type or select the desired exporting resolution in the Export every statistics reporting field.

When deciding the value for this option you must consider the setting selected for the sampling. If, for example, the Statistics Reporting Resolution (set through Tools | Global Options | General) is set to 20 seconds and Export every statistics reporting is set to 2, the measurements exported

reflect a sampling every 40 seconds. They are the measurements gathered at the intervals 20, 60, 100, 140…

This option is useful in cases where a large amount of data results from a long test session.

Setting the Export to Tab File Settings

WebLOAD reports can be exported as Tab Files (tab-delimited text files).

Open Viewer When Exporting Tab Files

When the Open viewer when exporting option is enabled (default), WebLOAD launches the program which serves as the default viewer (e.g., Notepad, Excel) for an exported Tab file. The default viewer is defined in the Tab File default viewer field (see below).

Tab File Export Settings

These options define the Tab File that can be exported to other programs.

To define the Tab File for export:

1. Select Tools | Global Options.
2. Select the Export tab.
3. In the Tab File delimiter field, enter the delimiter to be used in the Tab files that contain the WebLOAD statistical data for export.

   Any string can be used as a delimiter.

   - To use a tab character as a delimiter enter \t.
   - To use a new line as a delimiter, enter \r or \n.
   - To use a space as a delimiter, enter \s.
4. In the Tab File default directory field, set the default directory where the Tab files should be saved.
5. In the Tab file default viewer field, specify which program should serve as the default viewer (e.g., Notepad, Excel) for an exported Tab file.

Export to HTML Settings

WebLOAD reports can be exported as HTML Files.

Open Viewer When Exporting HTML Files

When the Open viewer when exporting option is enabled (default), WebLOAD launches the program, which serves as the default viewer.

HTML Export Settings

These options define the HTML files that can be exported to other programs.

To define the HTML files for export:

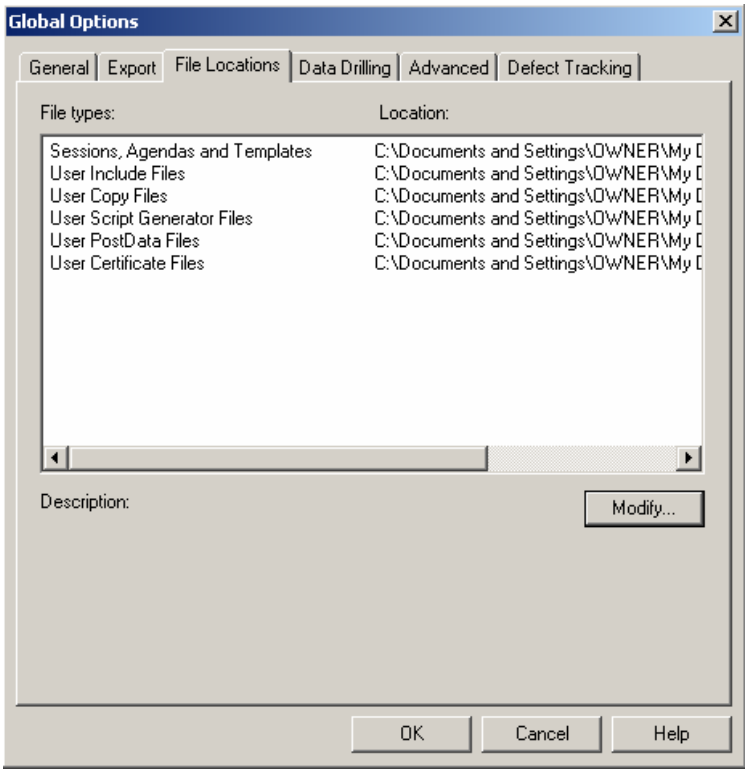
1. Select Tools | Global Options.
2. Select the Export tab.
3. In the HTML File default directory field, set the default directory where the HTML files should be saved.

The File Locations Tab

This dialog box lists the default locations of various WebLOAD files, such as Agenda and Table Compare files. You can modify these locations if you wish.

To access and configure the File Location tab options:

1. Select Tool | Global Options.
2. Select the File Locations tab.



3. Select the item whose default location you want to change, and click Modify.

   The Modify Location dialog box opens.

4. Select a new default location and click OK.

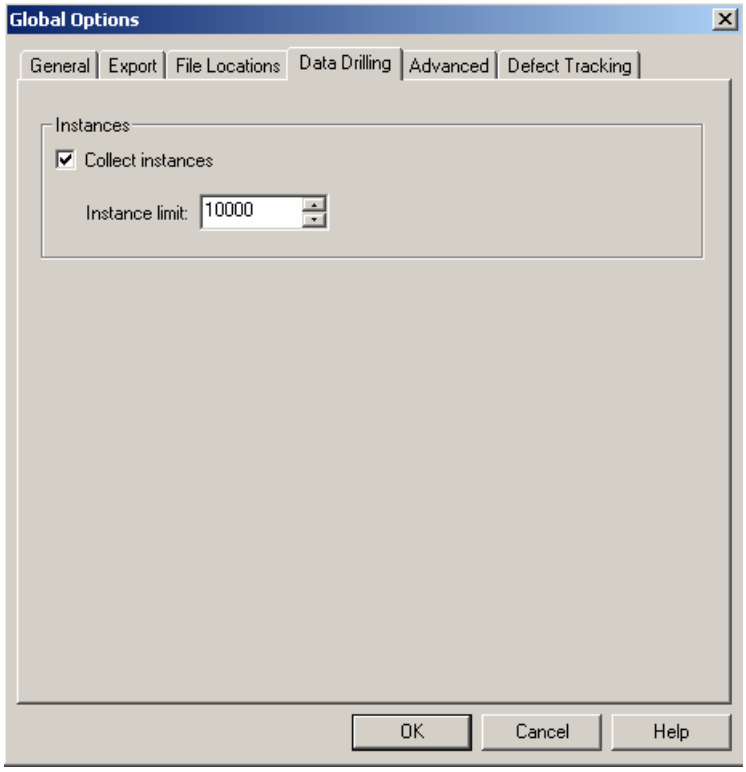
   The dialog box closes and the new location is listed.

The Data Drilling Tab

Data drilling enables you to display a detailed description of all user-defined and named transactions to the instance level. An instance is a single occurrence of a user-defined transaction in a given round. Data drilling is only enabled if the Collect Instances checkbox is selected.

To access and configure the Data Drilling tab options:

1. Select Tools | Global Options.
2. Select the Data Drilling tab.



3. Click the Collect Instances checkbox (default) to enable Data Drilling.
4. Specify the limit of collected instances in the Instance Limit field. The default limit is 1000.

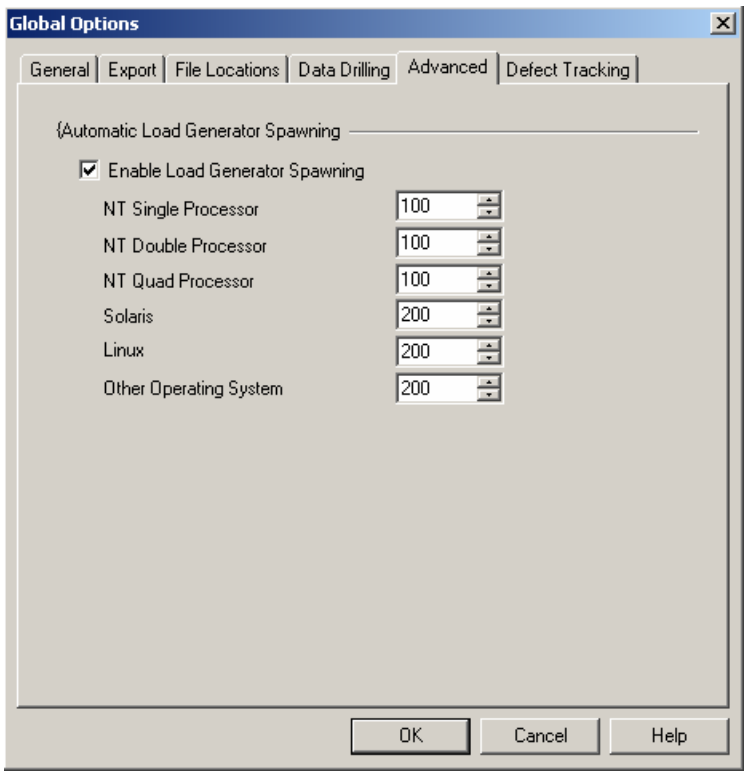
Note: If you disable the Collect Instances checkbox, the Data Drilling options of the Functional Testing tab are also disabled.

The Advanced Tab

Load Generator spawning sets the number of threads at which WebLOAD should automatically spawn Load Generators. Spawning Load Generators improves performance.

To access and configure the Advanced tab options:

1. Select Tools | Global Options.
2. Select the Advanced tab.



3. Under Automatic Load Generator Spawning, click Enable Load Generator Spawning (default).
4. Type or select the number of threads at which WebLOAD should automatically spawn Load Generators for the appropriate operating systems.
5. Click OK.

Notes:

Only client threads running within a single spawned process, on the same Load Generator, are able to share global variables. So if, for example, you have spawning set to 100 and you are running a total of 300 threads, realize that you are actually running three spawned processes on three separate Load Generators.

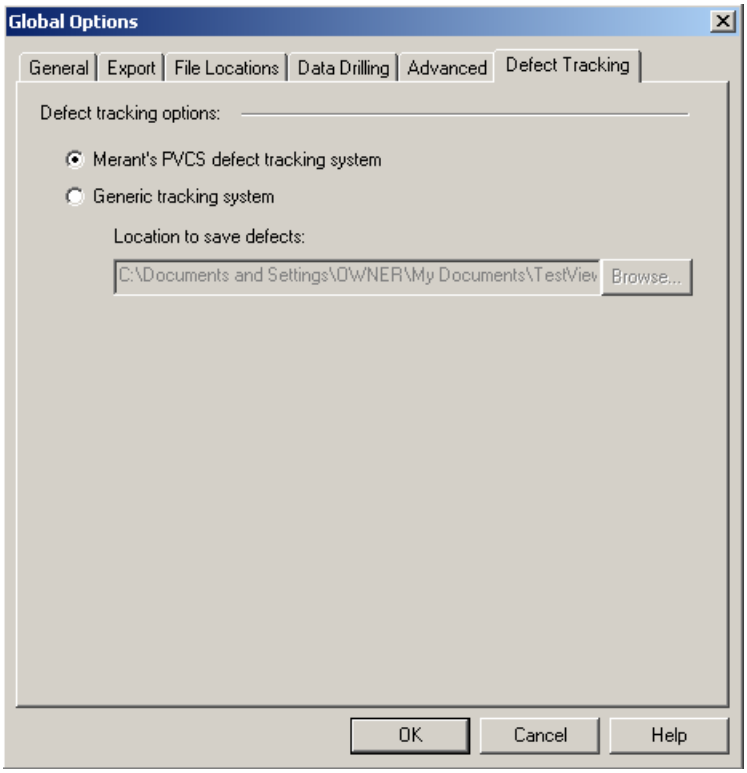
Only client threads running within a single spawned process, on the same Load Generator, can be synchronized.

The Defect Tracking Tab

The Defect Tracking options enable you to set the tracking system to use for tracking defaults.

To access and configure the Defect Tracking tab options:

1. Select Tools | Global Options.
2. Select the Defect Tracking tab.



3. Select the tracking system to use. If you select a generic tracking system enter the location to save the defects.

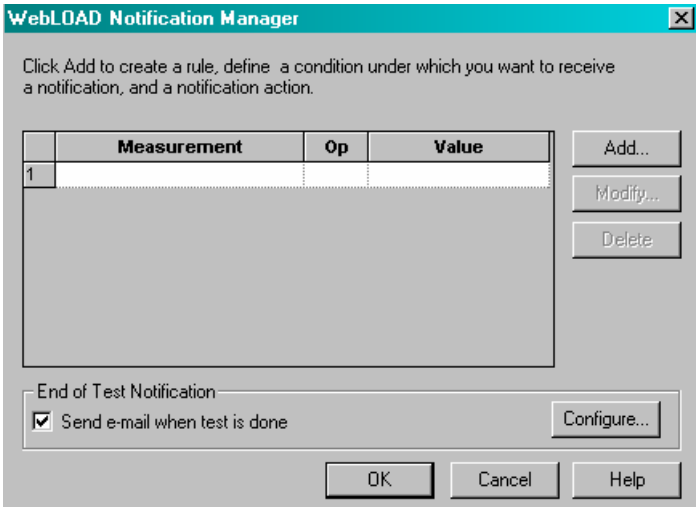
The Notification Manager

The Notification Manager allows you to define conditions under which someone should receive a notification, and optionally send an additional email action as well. Conditions include cases where a specified built-in Measurement value exceeds, equals, or drops down to less than a specified threshold value.

To access and configure the Notification Manager options:

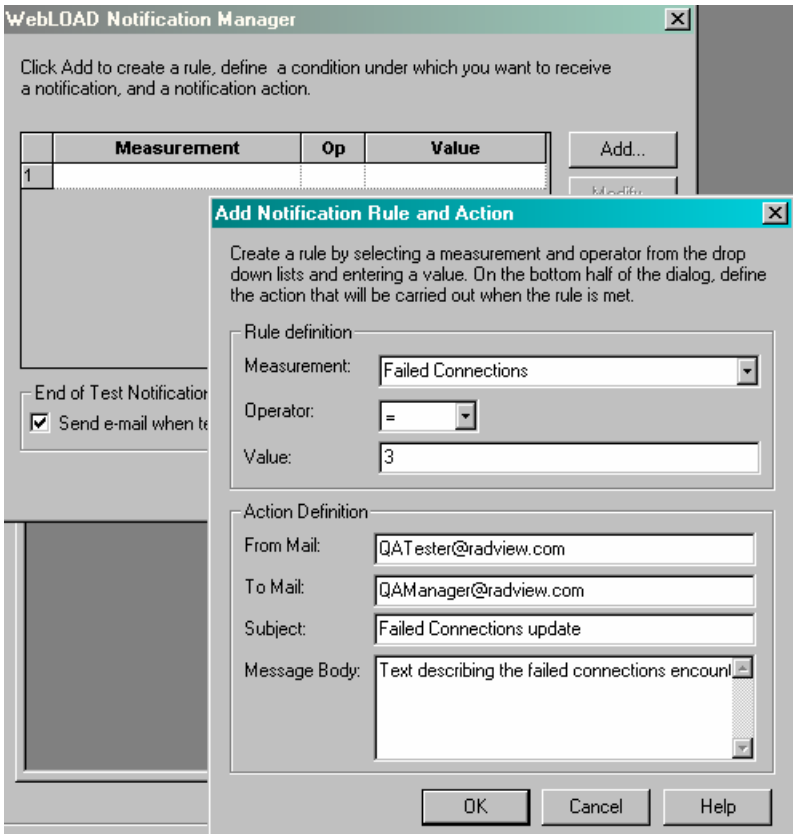
1. Select Session Control | Notification Manager from the main menu toolbar.

   The WebLOAD Notification Manager dialog box appears.



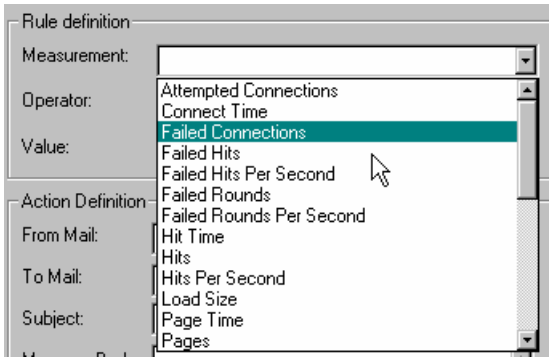
2. Click the Add button to define a new rule or condition under which someone should receive notification.

   The Add Notification Rule and Action dialog box appears.

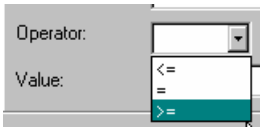


3. Create a new Mail Notification Rule by entering the following information into the Rule Definition input text boxes in the Add Notification Rule and Action dialog box.

   a. Select the measurement to be monitored from the list of built-in measurements that appears in the drop-down Measurement field, illustrated in the following figure:



b. Select the operator from the list of operator options that appears in the drop-down Operator field, illustrated in the following figure:



c. Specify the threshold value in the input text box of the Value field.

4. Define the parameters of the email notification to be sent out if the Rule conditions are met. Enter the email information in the Action Definition section of the Add Notification Rule and Action dialog box.

a. Define the official sender of the email in the input text box of the From Mail field.

b. Define the official recipient of the email in the input text box of the To Mail field.

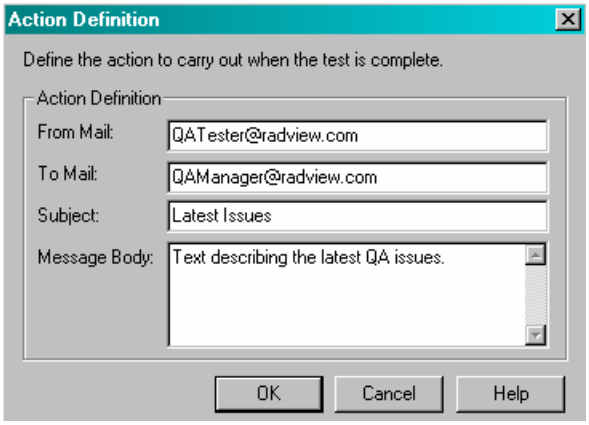
c. Define the email subject line in the input text box of the Subject field.

d. Type the message text body in the input text box of the Message Body field.

5. Click OK to return to the main WebLOAD Notification Manager dialog box.

6. To send an additional email notification when the test is completed, check the End of Test Notification checkbox in the main WebLOAD Notification Manager dialog box and click on the Configure button.

The Action Definition dialog box appears.



7. Define the parameters of the email notification to be sent out when the test is completed.

a. Define the official sender of the email in the input text box of the From Mail field.

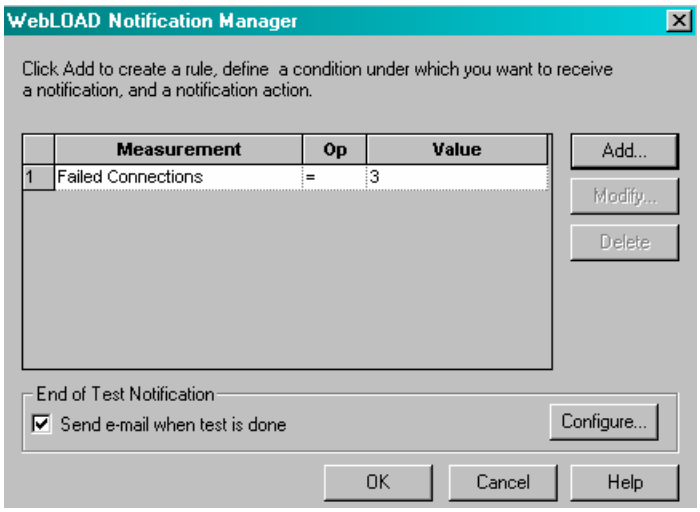
b. Define the official recipient of the email in the input text box of the To Mail field.

c. Define the email subject line in the input text box of the Subject field.

d. Type the message text body in the input text box of the Message Body field.

8. Click OK to return to the main WebLOAD Notification Manager dialog box.

The following figure illustrates a Notification Manager dialog box with one notification rule defined and a request for end-of-test notification.



C H A P T E R 1 6

Setting Agenda Options

Agenda Options enable you to modify Agendas to your specifications, and define the way WebLOAD behaves during runtime. Setting Agenda options allows you to emulate different types of user activity. For example, you can set WebLOAD to emulate users connecting to your application at different modem speeds.

Note: Agenda options, set through the Console, are not uploaded when editing the Agenda in WebFT. Debugger options are available (in WebFT) for debugging the Agenda in WebFT.

In This Chapter

About Agenda Options 222
Modifying Agenda Options 224
Setting the SSL Options 225
Client Type Settings 225
Setting Sleep Time Playback Settings 227
Setting the Browser Cache 229
Configuring Browser Parameter Settings 231
Setting the HTTP Parameters 237
Setting Authentication 240
Connection Speed Settings 242
Diagnostic Settings 243
Functional Testing Options 247
Pass / Fail Definition Settings 252
Configuring Reset Mechanism Settings 254
Configuring Emulation Settings 256
Web Services Settings 258

About Agenda Options

Agenda options allow you to configure WebLOAD behavior to emulate different types of user activity and ensure that the Virtual Clients running the test accurately emulate real users accessing your Web application.

There are three modes for modifying Agenda settings. Most of the Options tabs described in this section can be edited in any of these three modes. Option tabs that are only available through a specific mode are noted in the tab description.

Note: To modify Agenda options, you must be in Edit mode (select File | Edit Template, or click on the Edit toolbar button .

The three option modification modes are:

- Default Options- When building a new template or session, or editing an existing one, each new single Agenda or Agenda participating in a Mix is automatically assigned these defaults.

  To activate the Default Options from the WebLOAD Console:

  - Select Tools | Default Options,

    -Or-

    Right-click on the root node (Load Session node) and select Default Options.

- Current Session Options- It is possible to modify Agenda options for all Agendas participating in a specific load session or part of a specific load template without changing the default settings for all sessions.

  To activate this mode:

  - Select Tools | Current Session Options,

    -Or-

    Right-click the root node (Load Session node) and select Current Session Options.

  Modify the defaults as desired. All participating Agendas currently listed in the Session Tree are automatically assigned the modified default values.

- Agenda Options- It is possible to modify Agenda options for an individual Agenda within a new or existing load session or load template.

  To activate this mode:

  - Select Tools | Agenda Options,

    -Or-

    Right-click the desired Agenda in the Session Tree and select Agenda Options.

  The modifications you make here only affect that specific Agenda.

For Current Session Options and Agenda Options, WebLOAD automatically identifies the Client type (HTTP Client or Browser Client) which is associated with the Agenda file type and selects the right engine for execution.

File Type	Client / Engine Type	Client Type Options Behavior
.WLP	HTTP / Protocol	For Current Session Options and Agenda Options, the Client Type dialog box is disabled for .WLP Agendas.
.WLS	HTTP / Protocol	For Current Session Options only, the Client Type dialog box is disabled for .WLS Agendas.
.PFT	Browser / Emulation	For Current Session Options and Agenda Options, the Client Type dialog box is disabled for .PFT Agendas.
.FTS	Browser / Emulation	For Current Session Options only, the Client Type dialog box is disabled for .FTS Agendas.
.JS	May match both types so you can define the client type (HTTP by default)	For Current Session Options only, the Client Type dialog box is disabled for .JS Agendas.

Modifying Agenda Options

The Options dialog boxes are identical for all three modes, although they are activated differently, as described in About Agenda Options (on page 222).

The Options dialog boxes include the following tabs:

- SSL tab (see "Setting the SSL Options" on page 225)
- Client Type tab (see "Client Type Settings" on page 225) (only edited through the Default Options menu)
- Sleep Time Control tab (see "Setting Sleep Time Playback Settings" on page 227)
- Browser Cache tab (see "Setting the Browser Cache" on page 229)
- Browser Parameters tab (see "Configuring Browser Parameter Settings" on page 231)
- HTTP Parameters tab (see "Setting the HTTP Parameters" on page 237)
- Authentication tab (see "Setting Authentication" on page 240)
- Connection tab (see "Connection Speed Settings" on page 242)
- Diagnostic tab (see "Diagnostic Settings" on page 243)
- Functional Testing tab (see "Functional Testing Options" on page 247)
- Pass / Fail Definition tab (see "Pass / Fail Definition Settings" on page 252) (only edited through the Default and Current Session Options menus)
- Reset Mechanism tab (see "Configuring Reset Mechanism Settings" on page 254) (only edited through the Default and Current Session Options menus)
- Emulation Settings tab (see "Configuring Emulation Settings" on page 256)
- Web Services tab (see "Web Services Settings" on page 258)

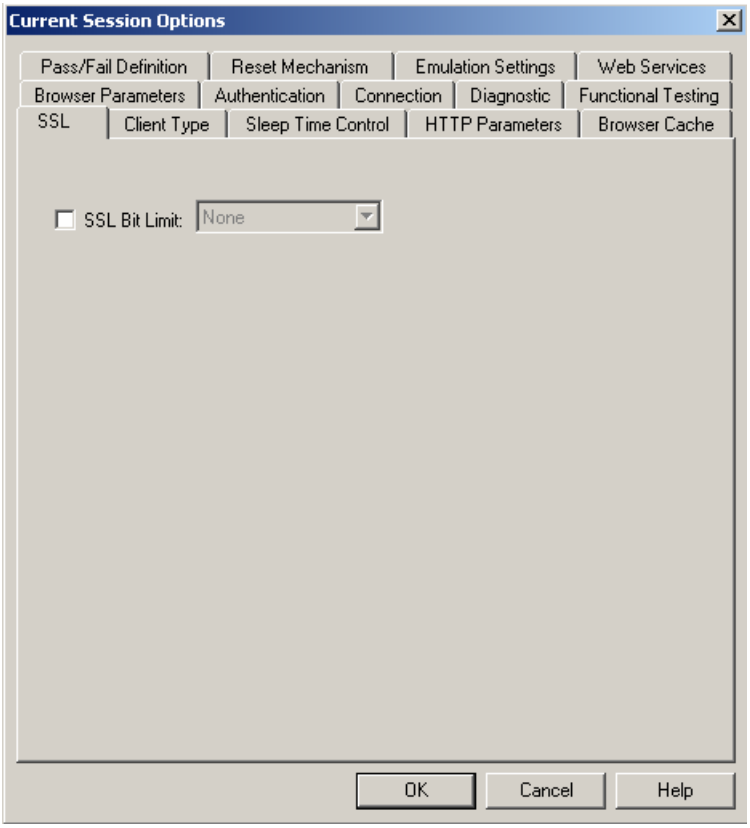
To modify Agenda or Session configuration options:

1. Select the desired tab.
2. Modify as many of the options you need.
3. Select another tab and modify its options.
4. Click OK to accept the modifications, or Cancel to reject them, and exit the Default / Current Session Options dialog box.

Setting the SSL Options

To access and configure the SSL tab options:

1. Select Tools | Default/Current Session Options.
2. Select the SSL tab.



3. Check SSL Bit Limit and select the highest SSL Bit Limit available between the Virtual Clients and the server.
4. Click OK to accept the modifications, or Cancel to reject them, and exit the Default / Current Session Options dialog box.

Client Type Settings

Test sessions are not all the same. Sometimes you want to test and scrutinize every single client activity, and verify every single transaction, no matter how trivial it seems or how frequently it occurs. Other times you want to focus on a specific category of problematic activities and can't waste time looking at anything else. You may even decide to generate a huge number of client

transactions in as short a time as possible to see how your application copes when pushed to the limit and beyond. And sometimes, you don't want to take any extra time or resources to measure anything specific, you just want to see how your Web application behaves for a typical user under standard operating conditions.

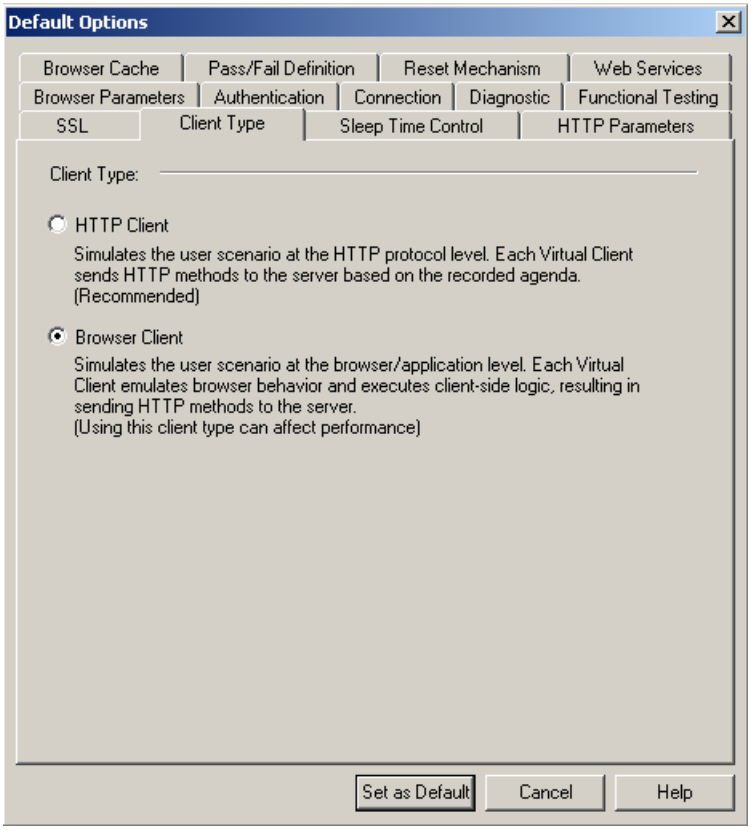
Multiple Client Type modes provide the ability to test your application using greater or lesser load to ensure optimal WebLOAD performance by defining the behavior of clients according to your specific requirements. The advantages of the different Client Type modes are explained in Getting Started (on page 21).

Notes:
1) This option is not available for individual Agendas. It can only be set for the entire session though the Default Options dialog boxes.

2) The Client Type dialog box is disabled for Current Session Options.

To configure the client type options:

1. Select Tools | Default/Current Session Options.
2. Select the Client Type tab.



3. Choose the Client Type you prefer.

   Each Client Type option is described in this section.

4. Click Set as Default to accept the modifications, or Cancel to reject them, and exit the Default Options dialog box.

Selecting HTTP Client Type Mode

During test sessions that run under HTTP Client Type mode, Agendas that concentrate on HTTP transactions and the responses they trigger are run through an HTTP protocol engine. This Client Type can be used for additional protocol-level testing, using other IP protocols in combination with HTTP. Use this Client Type for the most powerful low-level testing scenarios, for the highest load stress testing, and when working with Agendas not created in WebFT.

To select HTTP Client Type mode:

1. Select Tools | Default Options.
2. Select the Client Type tab.
3. Select HTTP Client.

Selecting Browser Client Type Mode

During test sessions that run under Browser Client Type mode, Agendas that recreate user browser activities are run through an emulation engine that simulates actual browser activities. Browser Client Type mode provides complete browser emulation and parsing of the DOM. Use this Client Type to achieve accurate emulation of real-life browsers combined with complex functional verifications.

To select Browser Client Type mode:

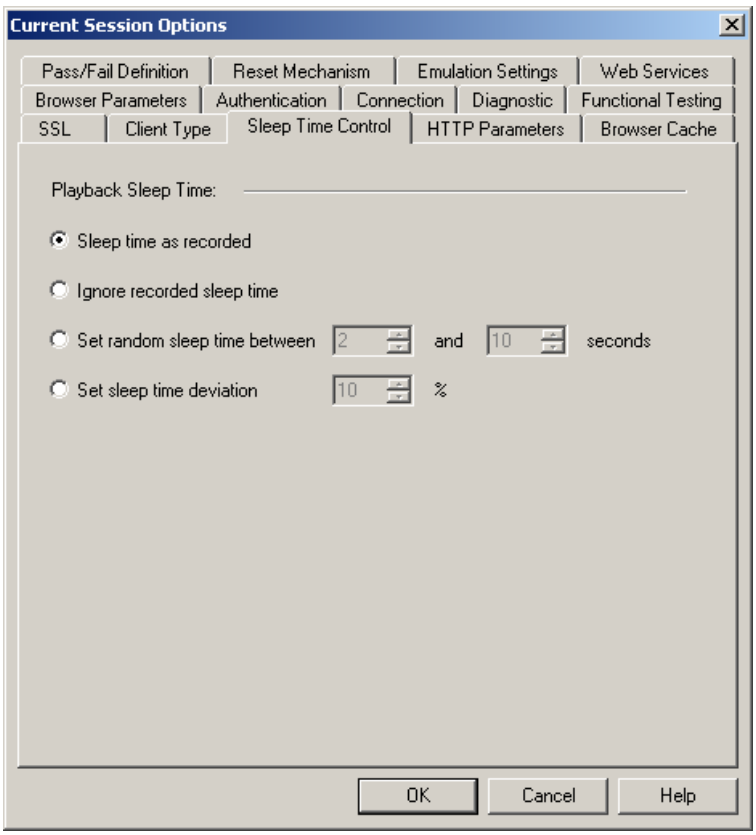
1. Select Tools | Default Options.
2. Select the Client Type tab.
3. Select Browser Client.

Setting Sleep Time Playback Settings

Sleep time is a pause to simulate the intermittent activity of real users. WebLOAD enables you to set an Agenda to execute with the sleep times recorded in the Agenda, random sleep times or no sleep times at all.

To set sleep time playback settings:

1. Select Tools | Default/Current Session Options.
2. Select the Sleep Time Control tab.



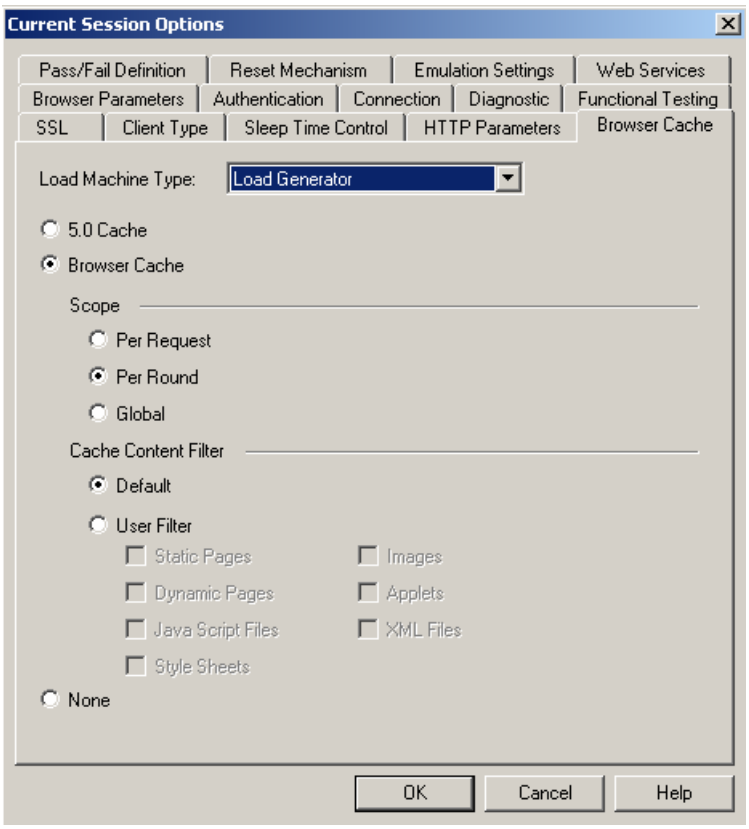
3. Specify the type of sleep to use when playing the Agenda.
4. There are four options:
   - Select Sleep time as recorded (default) to run the Agenda with the delays corresponding to the natural pauses that occurred when recording the Agenda.
   - Select Ignore recorded sleep time to eliminate any pauses when running the Agenda and run a worst-case stress test.
   - Select Set random sleep time and set the range (min and max) of delays to represent a range of users.
   - Select Set sleep time deviation and set a random percentage of the recorded sleep time to use when running the Agenda.
5. Click OK to accept the modifications, or Cancel to reject them, and exit the Default / Current Session Options dialog box.

Setting the Browser Cache

The Browser Cache option enables you to define the type and behavior of the cache that WebLOAD uses in order to simulate the behavior of a browser's cache.

To set the Browser Cache options:

1. Select Tools | Default/Current Project Options.
2. Select the Browser Cache tab.



3. From the Load Machine Type drop-down list, select Load Generator or Probing Client.
4. Select the type of cache by clicking one of the following: 5.0 Cache, Browser Cache, or None.
5. If you selected Browser Cache, do the following:
   a. Select a scope by clicking one of the following: Per Request, Per Round, or Global.
   b. Select a filter by clicking one of the following: Default or User Filter.
      If you selected User Filter, the filter checkboxes are enabled.

c. If you selected User Filter, select one or more User Filter checkboxes.

6. Click OK.

The following table describes the fields and buttons in the Browser Cache dialog box.

Field	Description
5.0 Cache	All Virtual Clients simulate the browser cache the same way that WebLOAD 5.0 used to: 1. Only images were cached. 2. The content in the cache was not stored, just an indication that the image was read. 3. Whenever the engine had a request for a cached image, it simply sent nothing to the server. 4. If the same image appeared more than one time in the same page, it would always be cached only once, even if the cache is turned off. Of course, it would be cached in further pages if the cache was turned on.
Browser Cache	Stores the most recent downloaded document in a virtual browser cache. This speeds up document accesses, just as real browser cache speeds up access time. The browser cache is automatically cleared at the end of each round. WebLOAD behaves as follows: 1. WebLOAD can cache anything. 2. WebLOAD stores actual HTML pages except for other resources, which is only an indication. 3. Whenever the engine has a request for a cached resource, it will send the request with an "if-modified-since" header. If the server responds with a 200 status, the engine will refresh the cache. 4. If the same resource appears more than one time in the same page, it will be requested with an "if-modified-since" header.
Scope	Defines when the cache will be cleared.
Per Request	Defines that cache be cleared after each request. Note: Cache will be cleared after each request on the Agenda level. It will not be cleared after internal requests.
Per Round	Defines that cache be cleared after each round.

Field	Description
Global	Defines that the cache will never be cleared. Each client maintains its own cache by the way and this is true for the old cache as well.
Cache Content Filter	The type of content to filter.
Default	Select to use the default, which is that WebLOAD will cache only images and static pages.
User Filter	You can specify which content types to filter. Note: If you selected User Filter and do not select any of the content types, then WebLOAD will cache all content types.
None	All Virtual Clients simulate a browser with no available cache.

Configuring Browser Parameter Settings

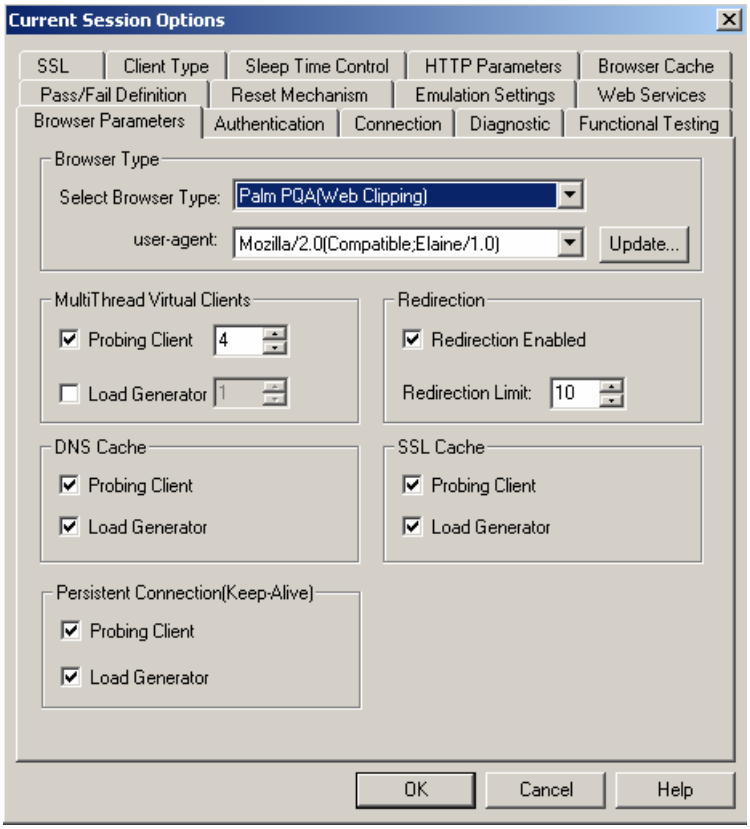
The Browser Parameters option enables you to define Virtual Client behavior such as:

- Browser type
- SSL encryption strength
- HTTP version
- Keep-Alive status

Each of these configuration options is described in this section.

To configure the Browser Parameters options:

1. Select Tools | Default/Current Session Options.
2. Select the Browser Parameters (default) tab.



3. Check the browser features you wish to use.

   Each option is described in this section.

4. Click OK to accept the modifications, or Cancel to reject them, and exit the Default / Current Session Options dialog box.

Note: The script in the Agenda (coded to account for a particular option) always overrides the Browser Parameters options. WebLOAD recommends always setting these options through the Console user interface.

Setting the Browser Type and User-Agent

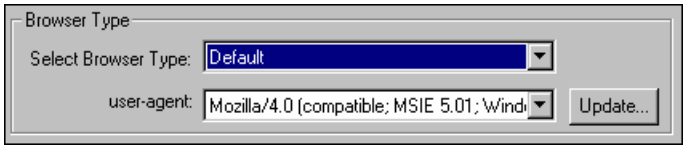
The browser type and user-agent settings specify the type of browser the WebLOAD Virtual Clients should emulate. You can set WebLOAD to emulate any of the standard browsers, or you

can specify any specific application by supplying a custom user-agent that is included in all HTTP headers. By default, all Virtual Clients use the WebLOAD Default browser agent.

To set the browser type and user-agent:

1. Select Tools | Default/Current Session Options.
2. Select the Browser Parameters (default) tab.
3. Select a browser type from the drop-down list.

   An appropriate user-agent automatically appears in the user-agent field.



4. You can select an alternative user-agent from the drop-down list, or click Update to add additional user-agents. See Adding a User-Agent (on page 233).

   If you select WAP as the browser type, the user-agent field changes to WAP-emulation. When WAP-emulation is selected, WebLOAD automatically sends all necessary headers, including the screen size.

Note: For the WebLOAD 7.x protocol engine, the default User Agent is Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.0). For the WebLOAD 5.x protocol engine, the default HTTP version was Mozilla/4.0 (compatible; MSIE 5.01; Windows NT; DigExt).

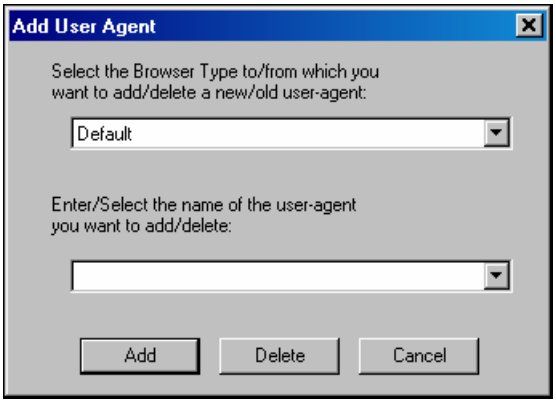
Adding a User-Agent

The available user-agent list is appropriate for the browser type you select. You can add to or delete from the user-agent list.

To add a user-agent:

1. Click the Update button beside the user-agent field on the Browser Parameters tab.

   The Add User Agent dialog box opens.



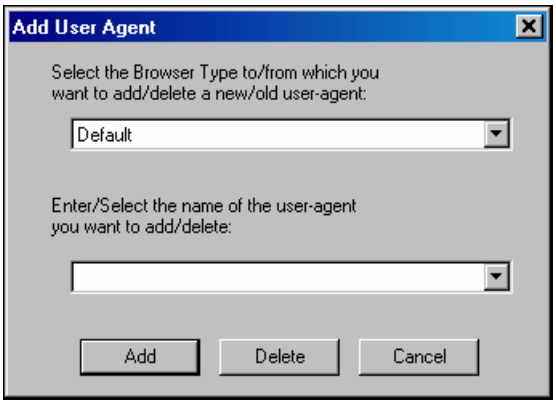
2. Type or select the browser type to which you want to add a user-agent.
3. Type or select the new user-agent.
4. Click Add.

Deleting a User-Agent

To delete a user-agent:

1. Click the Update button beside the user-agent field on the Browser Parameters tab.

   The Add User Agent dialog box opens.

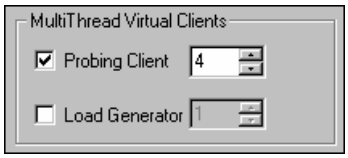


2. Type or select the browser type from which you want to delete a user-agent.
3. Type or select the user-agent to be deleted.
4. Click Delete.

Setting Load Generator and Probing Client Thread Properties

To set the Load Generator and Probing Client thread properties:

1. Select Tools | Default/Current Session Options.
2. Select the Browser Parameters (default) tab.
3. Set the Virtual Client properties by enabling / disabling multithreading for Load Generators or Virtual Clients.

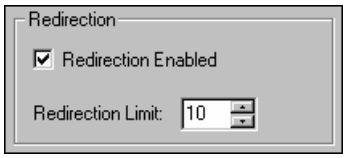


By default, Probing Client multithreading is enabled and is set to 1.

Setting Redirection Limits

To set a redirection limit:

1. Select Tools | Default/Current Session Options.
2. Select the Browser Parameters (default) tab.
3. Select Redirection Enabled.



4. In the Redirection Limit box, type or select the desired redirection limit.

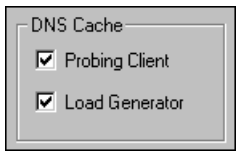
   The default limit is 10.

Simulating DNS Cache

When DNS Caching is enabled (default), WebLOAD caches the IP addresses received from the domain name server, reducing the domain name resolution time. Disable DNS caching if you want to include the time for domain name resolution in the WebLOAD performance statistics.

To simulate DNS cache:

1. Select Tools | Default/Current Session Options.
2. Select the Browser Parameters (default) tab.
3. Under DNS Cache, select Probing Client and Load Generator.

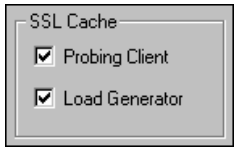


Simulating SSL Cache

When SSL Caching is enabled (default), WebLOAD caches the SSL decoding keys received from an SSL server, and WebLOAD only receives the key on the first SSL connection in each round. In subsequent connections, WebLOAD retrieves the key from cache. Enabling SSL Cache also reduces transmission time during SSL communication. Disable SSL caching if you want to measure the transmission time of the decoding key in the WebLOAD performance statistics for each SSL connection.

To simulate SSL cache:

1. Select Tools | Default/Current Session Options.
2. Select the Browser Parameters (default) tab.
3. Under SSL Cache, select Probing Client and Load Generator.



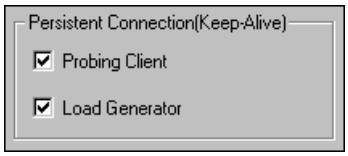
Enabling Persistent Connection (Keep-Alive)

When enabled, WebLOAD keeps an HTTP connection alive between successive accesses in the same round of the main script. Keeping a connection alive saves time between accesses. WebLOAD attempts to keep the connection alive unless you switch to a different server.

However, some HTTP servers may refuse to keep a connection alive. You should not keep a connection alive if establishing the connection is part of the performance test.

To enable persistent connection:

1. Select Tools | Default/Current Session Options.
2. Select the Browser Parameters (default) tab.
3. Under Persistent Connection, select Probing Client and Load Generator.

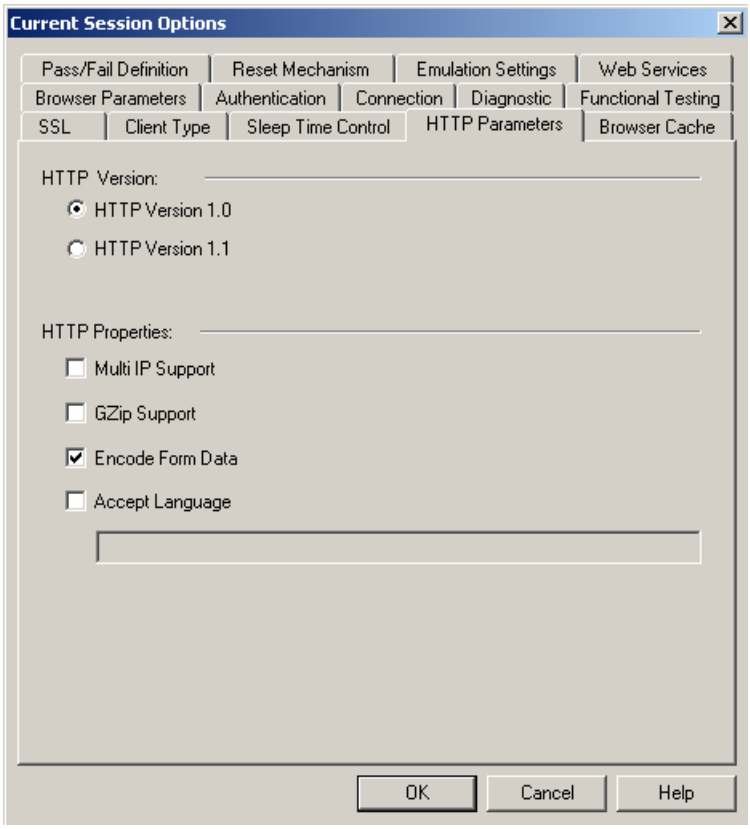


Setting the HTTP Parameters

The HTTP Parameters option enables you to define HTTP client behavior on the HTTP protocol level.

To set the HTTP Parameters options:

1. Select Tools | Default/Current Sessions Options.
2. Select the HTTP Parameters tab.



3. Set the HTTP version by clicking HTTP Version 1.0 or HTTP Version 1.1.

Note: For the WebLOAD 7.x protocol engine, the default HTTP version is 1.1. For the WebLOAD 5.x protocol engine, the default HTTP version is 1.0.

4. Select one or more HTTP Properties checkboxes.
5. Click OK.

The following table describes the fields and buttons in the HTTP Parameters dialog box.

Field	Description
HTTP Version	The appropriate HTTP protocol version (for example "HTTP/1.1").
HTTP Version 1.0	Sets the HTTP protocol version to 1.0.
HTTP Version 1.1	Sets the HTTP protocol version to 1.1.

Field	Description
HTTP Properties	
Multi IP Support	Sets the wlGlobals.MultiIPSupport flag.
GZip Support	Sets the wlGlobals.AcceptEncodingGzip flag. WebLOAD will behave as follows: ▪ For each request, sends the header "Accept-Encoding: gzip, deflate". This tells the server that the client can accept zipped content. ▪ When this header is turned on, the server MAY send a response with the header "content-encoding: gzip" or "content-encoding: deflate". If either of these headers is sent, it means that the response is zipped/deflated and WebLOAD will unzip/inflate the content. Notes: ◆ Most servers will work correctly even if the client does not send the "Accept-Encoding: gzip, deflate" header. Therefore, unless needed, it is recommended not to set the wlGlobals.AcceptEncodingGzip flag because it is performance heavy. However, some servers will fail if it is not sent. Microsoft Internet Explorer/Mozilla sends it by default. ◆ You can manually code the agenda to send the "Accept-Encoding: gzip, deflate" header even if the wlGlobals.AcceptEncodingGzip flag is not set. In this case, if the server returns zipped content, WebLOAD will not unzip it, so the content will not be available for WebLOAD to function properly.
Encode Form data	Sets the wlGlobals.EncodeFormdata flag. In general, when an HTTP client (Microsoft Internet Explorer/Firefox or WebLOAD) sends a post request to the server, the data must be HTTP encoded. Special characters like blanks, ">" signs, and so on, are replaced by "%xx". For example, space is encoded as "%20". This encoding can be performance heavy for large data, so WebLOAD allows you to turn it off. This should ONLY be done if you are sure that the data does not contain special characters. If it does, and the customer wants to improve performance via this flag, then you should replace the special characters within the script or use wlHttp.EncodeFormdata to set the flag for specific requests.

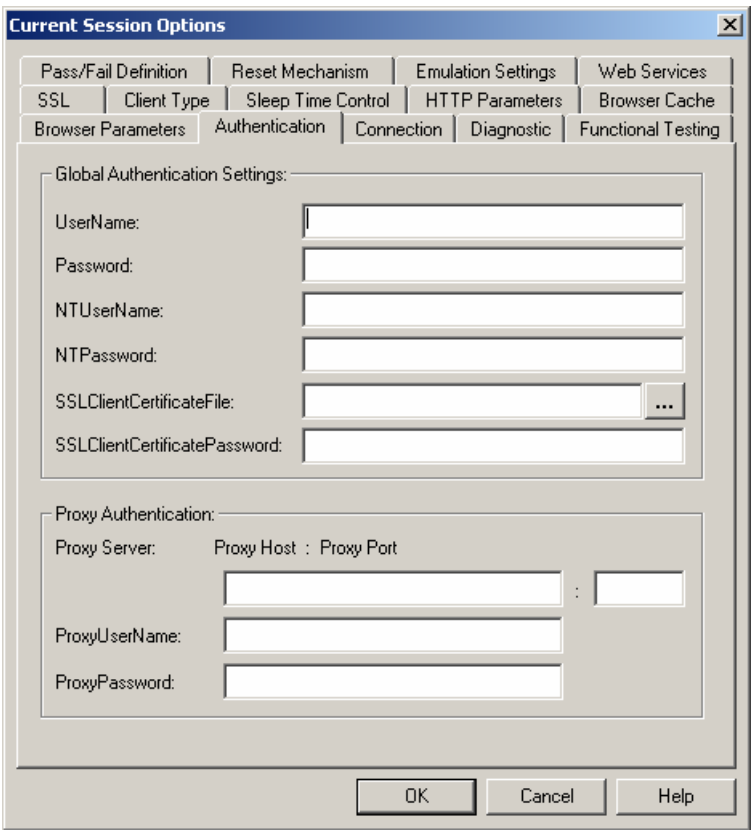
Field	Description
Accept Language	Sets the wlGlobals. AcceptLanguage flag. This flag defines a global value for the "Accept-Language" header which will be sent with each request. Some applications/servers will behave differently depending on the setting. It is a simple string and WebLOAD does not enforce any checks on the values. It is similar to the wlGlobals.UserAgent property in the sense that it is a wlGlobals/wlHttp setting that affects the value of request headers.

Setting Authentication

The Authentication option enables you to define the Global and Proxy authentication settings.

To set authentication:

1. Select Tools | Default/Current Session Options.
2. Select the Authentication tab.



3. Type the appropriate user names and passwords into the corresponding Global Authentication Settings input text boxes:

Field	Description
UserName and PassWord	The user name and password the Agenda should use to log onto restricted HTTP sites.
NTUserName and NTPassWord	The user name the Agenda should use for Windows NT Challenge response authentication.
ProxyUserName and ProxyPassWord	The user name and password the Agenda should use for proxy servers that require user authorization.

Field	Description
SSLClientCertificateFile and SSLClientCertificatePassWord	The filename (optionally including a directory path) and password of the certificate WebLOAD makes available to SSL servers.

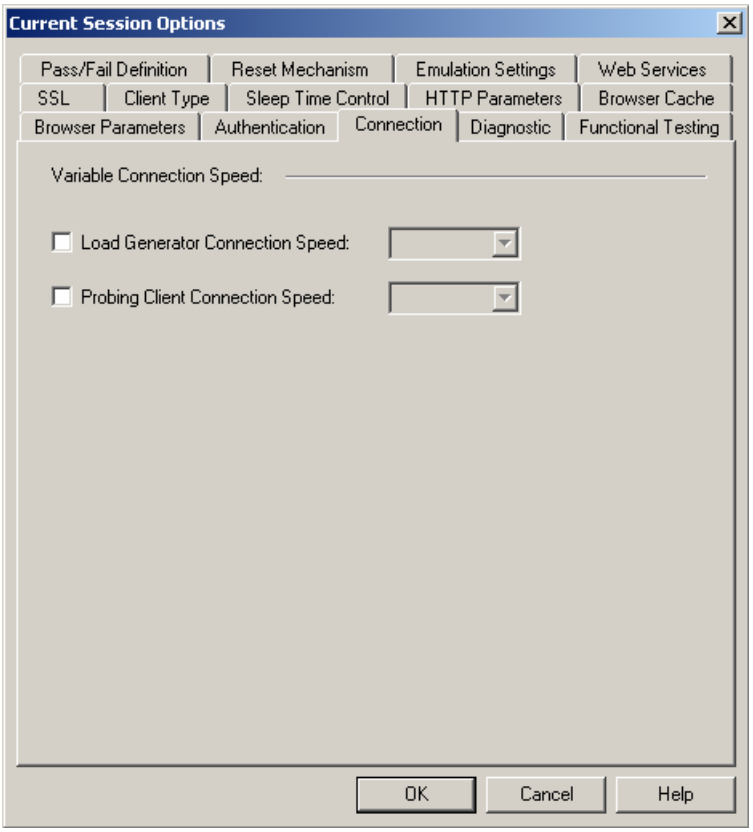
4. Type the appropriate proxy host name, port number, user name, and password into the corresponding Proxy Authentication input text boxes when working with proxy authentication.
5. Click OK to accept the modifications, or Cancel to reject them, and exit the Default / Current Session Options dialog box.

Connection Speed Settings

WebLOAD enables you to simulate various systems and connection configurations, including a 'virtual limit' on the connection speed available during a test. Setting a limit on connection speed allows testers to test applications for users that may be limited in their workplace connections speeds.

To configure the connection speed:

1. Select Tools | Default/Current Session Options.
2. Select the Connection tab.



The Variable Connection Speed feature enables you to set the connection speed for the load and emulate users connecting through different modem connections (14400, 28800…).

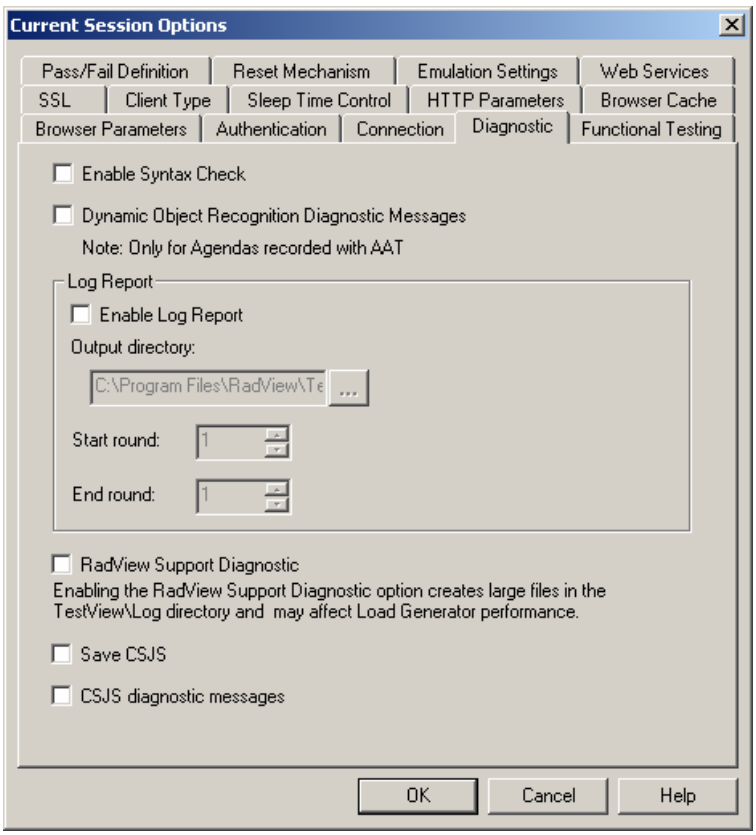
3. Select the Load Generator Connection Speed checkbox and type or select the appropriate speed from the list.
4. Select the Probing Client Connection Speed checkbox and type or select the appropriate speed from the list.
5. Click OK to accept the modifications, or Cancel to reject them, and exit the Default / Current Session Options dialog box.

Diagnostic Settings

Diagnostic options can be enabled while developing an Agenda or for tracking problems in existing Agendas.

To configure the diagnostic options:

1. Select Tools | Default/Current Session Options.
2. Select the Diagnostic tab.



Note: Running a load session with the diagnostic options enabled may affect system performance. Therefore, we recommend that these options remain disabled while running a load session with multiple threads.

3. Check the diagnostic features you wish to use.

   Each diagnostic option is described in this section.

4. Click OK to accept the modifications, or Cancel to reject them, and exit the Default / Current Session Options dialog box.

Enabling Syntax Checking

Enable Syntax Checking to perform the following tests on an Agenda while it is running:

To enable syntax checking:

1. Select Tools | Default/Current Session Options.
2. Select the Diagnostic tab.
3. Check Enable Syntax Check.

The following diagnostics are performed:

- Type inspections - WebLOAD checks that each property receives the correct type.

  For example, `wlLocals.ParseForms = 14` prompts the following log message:

  ```
  "Wrong type for the property ParseForms. The correct type is
  Boolean. Legal values are: "Yes"/"No" or true/false."
  ```

- Value inspections - WebLOAD checks to ensure that each property is assigned a legal value.

  For example, `wlHttp.Version = "2.1"` prompts the following log message:

  ```
  "2.1 is an illegal value for the property Version. Legal
  values are: 1.0, 1.1."
  ```

- Scope inspections - WebLOAD checks that each property is assigned a permitted scope.

  For example, `wlLocals.ConnectionSpeed = 28800` prompts the following log message:

  ```
  "The property ConnectionSpeed is not valid for the object
  wlLocals."
  ```

- Case inspections - WebLOAD objects and properties are case sensitive. When Syntax Check is enabled, WebLOAD checks to ensure that all objects and properties are written correctly.

  For example, `wlLocals.parse = "No"` prompts the following log message:

  ```
  "The property parse should be written as Parse."
  ```

It is recommended that a syntax check be run at least once while developing your Agenda.

Enabling Dynamic Object Recognition Diagnostic Messages

Enable this option to set WebLOAD to send InfoMessages on the status of Dynamic Object Recognition (DOR) IdentifyObject to the Log Window. This option is only available for Agendas created for WebLOAD 4.x or WebLOAD 5.0.

Note: This option is for backward compatibility purposes only, to continue support of scripts that were developed using the AAT.

To enable Dynamic Object Recognition diagnostic messages:

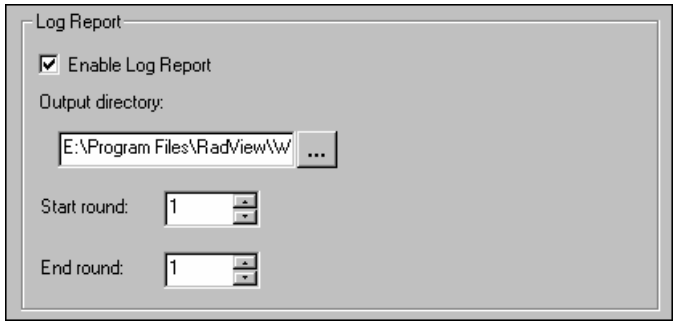
1. Select Tools | Default/Current Session Options.
2. Select the Diagnostic tab.
3. Check Dynamic Object Recognition Diagnostic Messages.

Log Report Diagnostics

Log reports are used to see the Agenda flow, and the response headers received while an Agenda is running. When log report is enabled, an outfile is automatically created for each get and post transaction and for each frame received.

To enable log report diagnostics:

1. Select Tools | Default/Current Session Options.
2. Select the Diagnostic tab.
3. Check Enable Log Report to configure WebLOAD to create a Log Report.



4. Click in the Output File Directory field and select the directory for saving Log Report files.

   The default directory is: `C:\Program Files\RadView\WebLOAD\User\Log`.

5. Select which rounds should generate the Log Report in the Start round and End round fields.

The log report displays the time a request was made, the round number, the request number, the URL received and the response headers. If `GetContents = "Yes"` the same information is displayed for each frame.

A message is sent to the Log Window for each request. The message appears in the following format:

```
### Thread:000, Round Num:001, Request Num:001 ### URL
```

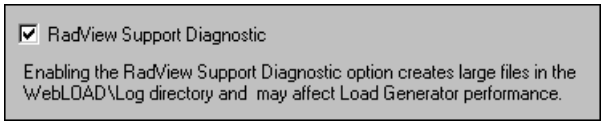
Each request is numbered sequentially for each thread and round number.

Enabling RadView Support Diagnostic

Enabling the RadView Support Diagnostic option creates large files in the WebLOAD\User\Log directory and may affect Load Generator performance.

To enable RadView support diagnostic:

1. Select Tools | Default/Current Session Options.
2. Select the Diagnostic tab.
3. Check RadView Support Diagnostic.



Enabling Save CSJS and CSJS Diagnostic Messages

Save CSJS and CSJS diagnostic messages can be used for generating troubleshooting information for RadView support. This option should only be set to gather additional data which may be requested by RadView support.

To enable Save CSJS and CSJS diagnostic messages:

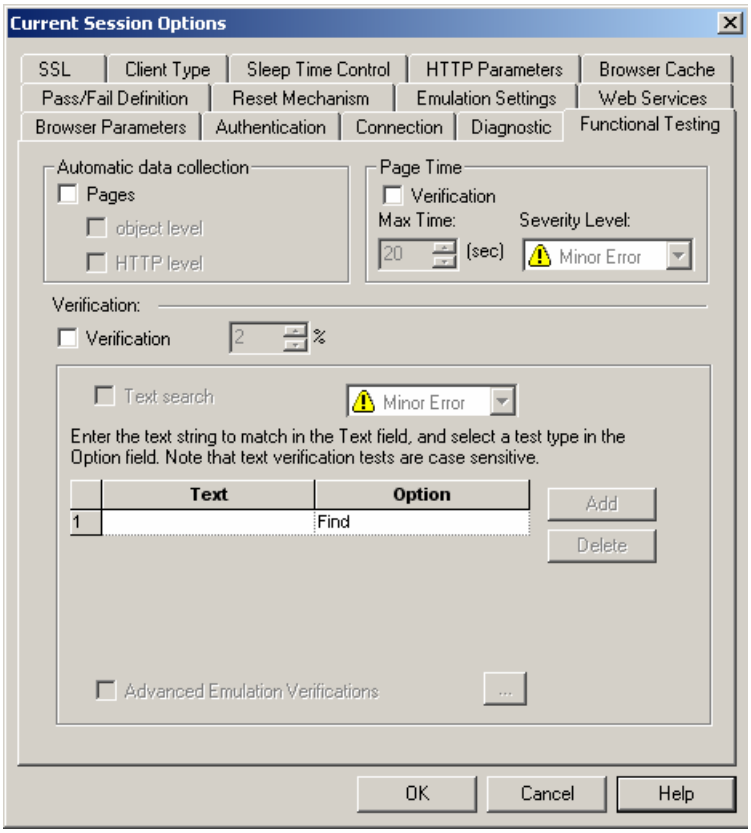
1. Select Tools | Default/Current Session Options.
2. Select the Diagnostic tab.
3. Select or clear the Save CSJS checkbox.
4. Select or clear the CSJS diagnostic messages checkbox.

Functional Testing Options

Functional testing enables you to insert verification tests into the Agenda to verify that the links, images, forms, tables and frames in your Web application continue to act as expected.

To access and configure the functional testing options:

1. Select Tools | Default/Current Session Options.
2. Select the Functional Testing tab.



3. Check the functional testing tools you wish to use.

   Each testing option is described in this section.

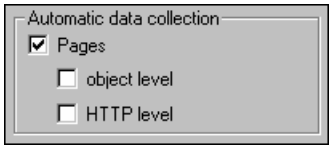
4. Click OK to accept the modifications, or Cancel to reject them, and exit the Default / Current Session Options dialog box.

Setting Automatic Data Collection

Setting automatic data collection instructs WebLOAD to handle every page, object and HTTP object in the Agenda as a transaction. The name of the transaction is the name of the page or object and data is gathered for each transaction.

To set automatic data collection:

1. Select Tools | Default/Current Session Options.
2. Select the Functional Testing tab.
3. Check the Pages checkbox to turn all page transactions in the Agenda into Named Transactions. The page name is the URL being accessed without the query string. Statistics are gathered for all page transactions.



4. Check Object level and/or HTTP level to specify whether to collect data at object or HTTP level or both.

Note: For the WebLOAD 7.x protocol engine, the default automatic data collection setting is Pages Off. For the WebLOAD 5.x protocol engine, the default automatic data collection setting is Pages On.

Setting Page Time Verifications

Setting page time verifications enables you to define a limit to the maximum amount of time a page should take to load. If the maximum time is exceeded, a message or error entry appears in the log window, depending on the severity level set for page time verifications.

To set page time verification:

1. Select Tools | Default/Current Session Options.
2. Select the Functional Testing tab.
3. Check the Verification checkbox.

4. Specify the maximum amount of time that a page should take to load in the Max Time field.
5. Specify the severity level if the page exceeds the specified time in the Severity Level field.

Note: It is important to specify a meaningful time period. For example, if you specify 3 seconds for a page which is likely to take longer, numerous essentially meaningless failures will result. Conversely, if you specify 10 seconds for a page which should take less than 1 second, the test becomes meaningless.

Inserting a Text Verification Test

Use Text verifications to verify the absence or presence of selected text expressions within your Web page.

To insert a text verification test into your Agenda:

1. Select Tools | Default/Current Session Options.
2. Select the Functional Testing tab.
3. Under Verification, select the Verification checkbox.
4. Select the percentage of time the tests should be performed. This is the percentage of Virtual Clients that will perform the verification tests.

   Note: You may select the Verification checkbox without selecting the Text Search checkbox and the Advanced Emulation Verifications checkbox in the frame beneath it. It is for backward compatibility for scripts that were recorded with the AAT and support table comparisons.

5. Select the checkbox adjacent to the Text Search test.

   Note: Selecting a lower level checkbox selects and highlights the upper level checkbox.

6. Select the severity level from the drop-down list next to the test. For further information, see Verification Function Return Values (on page 266).
7. Enter a literal string with the text for which you are searching in the Text field.

   Note: The text string is case sensitive.

8. Choose the type of test to run from the drop down list in the Option field:
   - Text to Find - the verification test will pass if the selected text is found in the current Web page.
   - Text not to Find - the verification test will fail if the selected text is found in the current Web page.

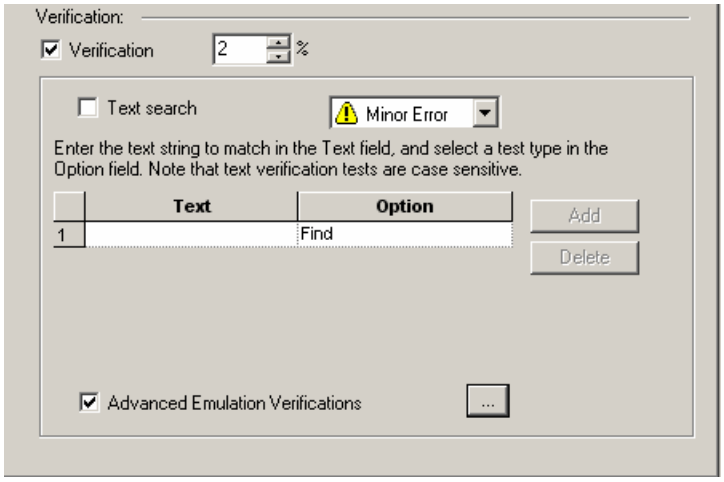
   For example, to ensure that the word "error" does not appear on the page during runtime, enter the word "error" in the Text field, and select Not to Find. If the text error is found on the page, WebFT will report the error.
9. Click Add to add another string to the list of Text Search strings.
10. Click Delete to delete a selected string from the list of Text Search strings.

Inserting Test Selections

To select a test:

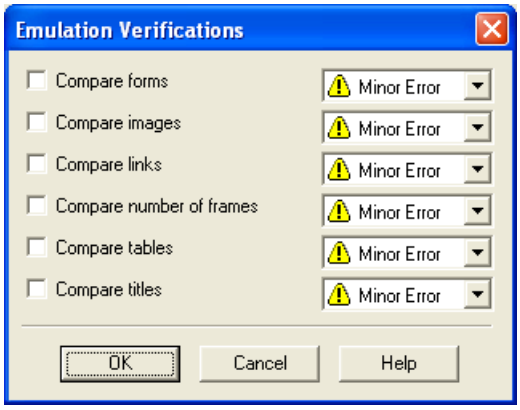
1. Select Tools | Default/Current Session Options.
2. Select the Functional Testing tab.
3. Under Verification, select the Verification checkbox.

   The Advanced Emulation Verifications checkbox is enabled.



4. Select the Advanced Emulation Verifications checkbox, and click the [...] button.

   The Emulation Verifications dialog box appears.



5. Select one or more individual tests to perform by marking the checkbox adjacent to the test (for information on each test, see Functional Verification Test List (on page 264)).
6. Select the severity level for each selected test (for information on severity levels, see Verification Function Return Values (on page 266)).
7. Click OK.

For a full description of the tests in Functional Testing and their results, see Functional Verification Testing (on page 259).

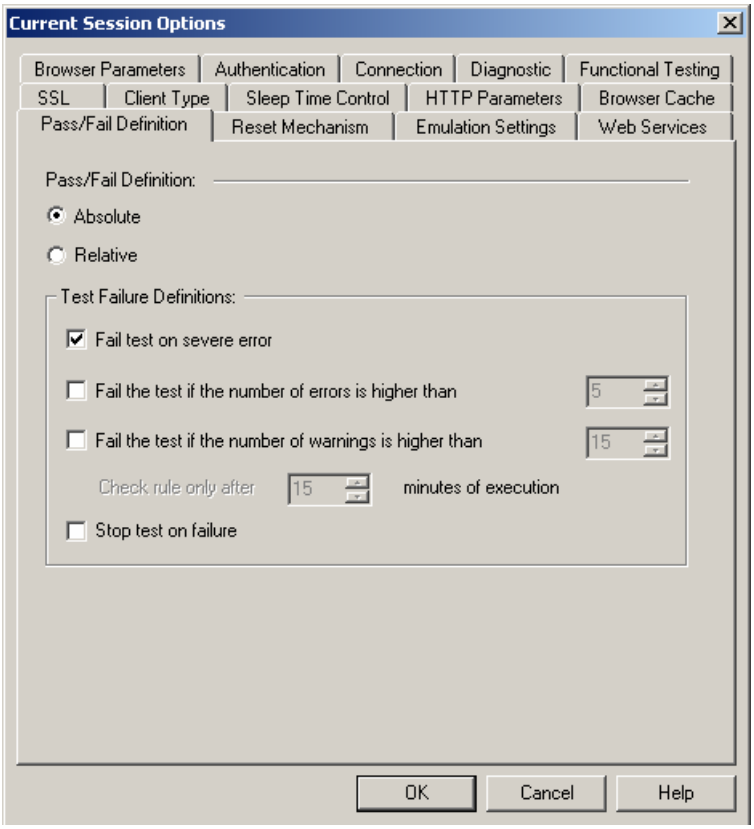
Pass / Fail Definition Settings

The Pass / Fail Definition options enable you to set test failure conditions.

Note: This option is not available for individual Agendas. It can only be set for the entire session though the Default Options or the Current Session Options dialog boxes.

To configure pass/fail definition options:

1. Select Tools | Default/Current Session Options.
2. Select the Pass / Fail Definition tab.



You can set WebLOAD to declare a test as failed based on a number of criteria. First, you must decide if you want the test to be professed a failure after an absolute number of failures or a relative number of failures. Then you set the conditions upon which a test is declared a failure.

3. Select Absolute (default) to set the total number of failures required for a test to be declared failed,

   -Or-

   Select Relative to set the percent of failed transactions need for a test to be declared failed.

4. Set the conditions upon which to declare a test failed and the action to take if a test fails.

   - Fail test on severe error - Sends a failure message and declares the test failed in the event of a severe error.

- Fail the test if the number / percent of errors is higher than - Declares the test failed if the number of errors received is higher than the number stated. If the pass / fail definition is set to relative then the percentage of errors is considered rather than the absolute value.
- Fail the test if the number / percent of warnings is higher than - Declares the test failed if the number of warnings received is higher than the number stated. If the pass / fail definition is set to relative then the percentage of warnings is considered rather than the absolute value.
- Check rule only after X minutes of execution - This option is only enabled when the pass / fail definition is set to relative. Use this option to set a stabilizing period for the test when it starts. Errors or warning received during this time frame will not fail the test, and the rules set for failing the test will only be checked after this timeframe has expired.
- Stop test on failure - Stops the test if the test is declared failed.

5. Click OK to accept the modifications, or Cancel to reject them, and exit the Default / Current Session Options dialog box.

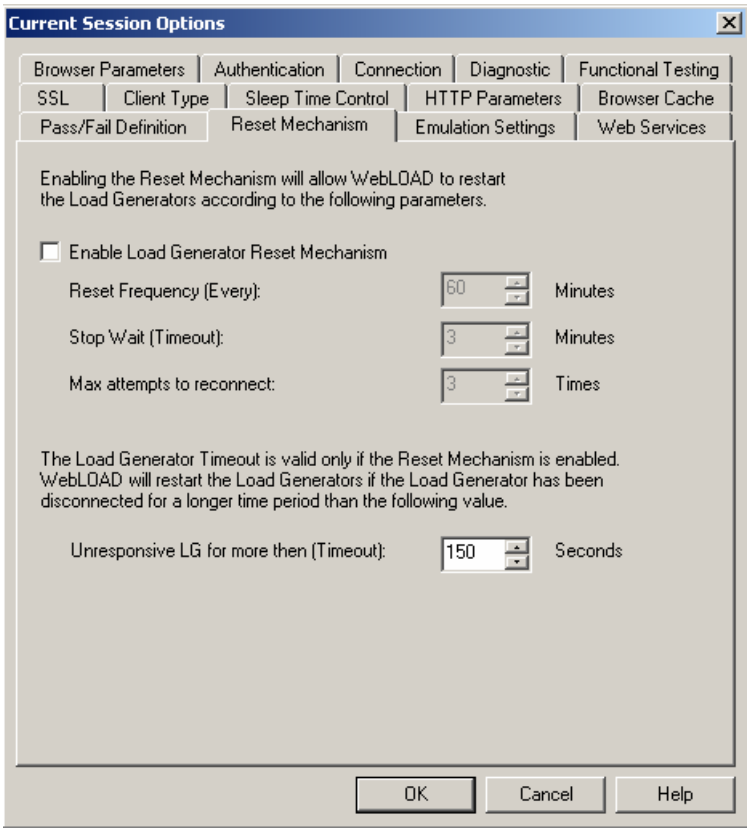
Configuring Reset Mechanism Settings

The Reset Mechanism options enable you to set the reset values.

Note: This option is not available for individual Agendas. It can only be set for the entire session though the Default Options or the Current Session Options dialog boxes.

To configure reset mechanism options:

1. Select Tools | Default/Current Session Options.
2. Select the Reset Mechanism tab.



3. Select the Enable Load Generator Reset Mechanism checkbox to allow WebLOAD to restart the Load Generators.
4. Set the number of minutes for Reset Frequency (Every) and Stop Wait (Timeout).
5. Set the number of times for Max attempts to reconnect.
6. Set the number of seconds for Unresponsive LG for more then (Timeout).

Note: The Load Generator Timeout is valid only if the Reset Mechanism is enabled.

7. Click OK to accept the modifications, or Cancel to reject them, and exit the Default / Current Session Options dialog box.

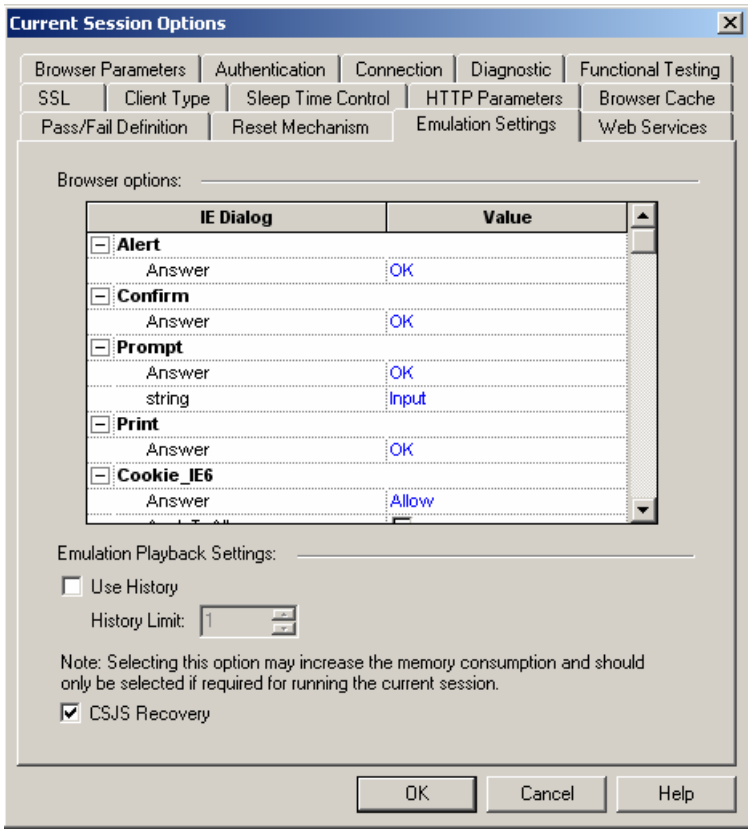
Configuring Emulation Settings

At times, Web applications may display pop-up dialog boxes unexpectedly. Examples of such dialog boxes include prompts, security warnings, JavaScript alerts, and confirm dialog boxes. Sometimes, the test cannot continue until you close the dialog box, for example, by clicking OK to close the dialog box. WebLOAD functional testing session Agendas can be set to automatically handle unexpected dialog boxes appropriately during the course of a test session by dismissing the unexpected dialog boxes.

Using the Browser Dialog options you instruct WebLOAD to detect dialog boxes and define the way WebLOAD should react to dialog boxes that may or may not appear during runtime by defining default values. For example, you can instruct WebLOAD to detect the security alert "`You are about to leave a secure Internet connection. It will be possible for others to view the information you send. Do you want to continue?`" message. You can instruct WebLOAD to answer Yes to close the dialog box.

To configure the emulation options:

1. Select Tools | Default/Current Session Options.
2. Select the Emulation Settings tab.



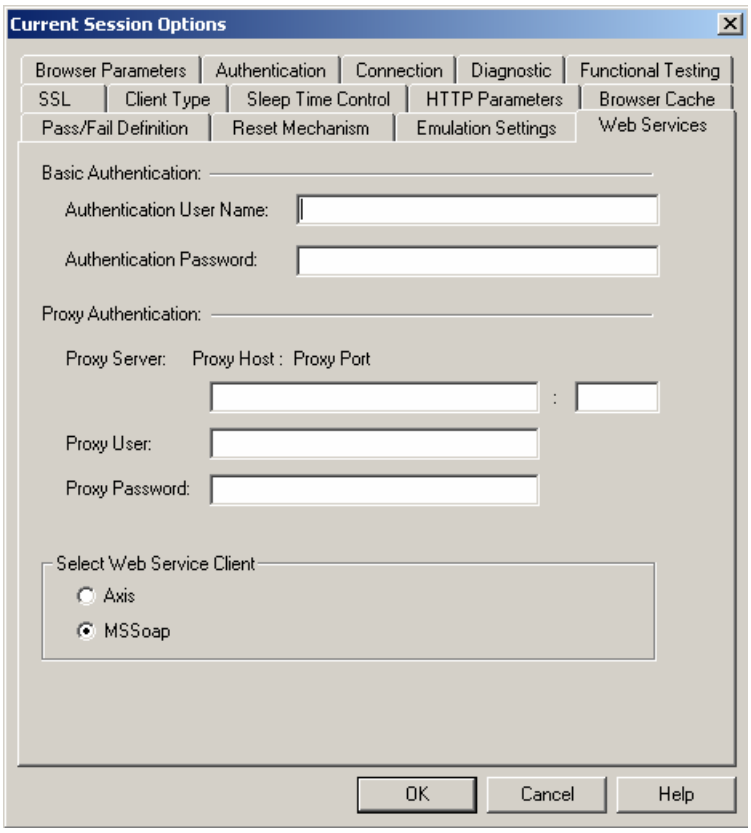
3. For each type of supported dialog select the action to take if the dialog should appear unexpectedly during runtime. Define the answer by clicking on the value next to the dialog box name and selecting a value from the drop down list, clicking or clearing a checkbox or entering the value in the field box provided.
4. Select Use History to instruct WebLOAD to store previous commands. Set a history limit to define an upper limit to the number of commands to be saved in a history file.
5. CSJS Recovery is selected by default.
6. Click OK to accept the modifications, or Cancel to reject them, and exit the Default / Current Session Options dialog box.

Web Services Settings

The Web Services tab enables you to set authentication values for working with Web Services in your tests and the Web Service client.

To configure the Web Services options:

1. Select Tools | Default/Current Session Options.
2. Select Web Services tab.



3. Enter the appropriate values in the basic and proxy authentication fields.
4. Select a Web Service client.
5. Click OK to accept the modifications, or Cancel to reject them, and exit the Default / Current Session Options dialog box.

C H A P T E R 1 7

Functional Verification Testing

WebLOAD combines unified load, performance, and functional testing into a single process. allowing you to verify the scalability and integrity of Web applications at the per-client, per-transaction and per-instance level under user-defined load conditions. By creating and analyzing the Document Object Model (DOM) for every Virtual Client during a test session, WebLOAD is able to verify each success and failure and present detailed information to you about each transaction.

The WebLOAD Functional Verification Test Tool is an easy-to-use tool which automates the process of inserting verification tests into the Agenda and verifies that the links, images, forms, tables and frames in your Web application continue to act as expected.

WebLOAD also enables you to insert application level transactions into the Agenda and define verification criteria, which compare expected outcomes against actual ones. WebLOAD tracks the success and failure rates of the transactions for each Virtual Client and integrates the data into the WebLOAD Statistics Report. You can find additional information about functional testing in the *TestView Programmer's Guide*.

In This Chapter

The Functional Testing Tab 260
User-Defined Transactions and Verification 264
Verification Function Return Values 266
Reporting Events 267
Tracking Reasons for Transaction Failure 267
Viewing Verification Test Results 269

The Functional Testing Tab

The verification tests you specify within WebLOAD are performed on all Web pages in the Agenda. To configure verification tests on all pages, set the global verification options through the Tools | Default / Current Session Options | Functional Testing dialog box. This section describes the options available through the Functional Testing tab.

Opening the Functional Testing Tab

To open the Functional Testing tab:

- Select Tools | Default Options... to set the verification tests to run for all Agendas,

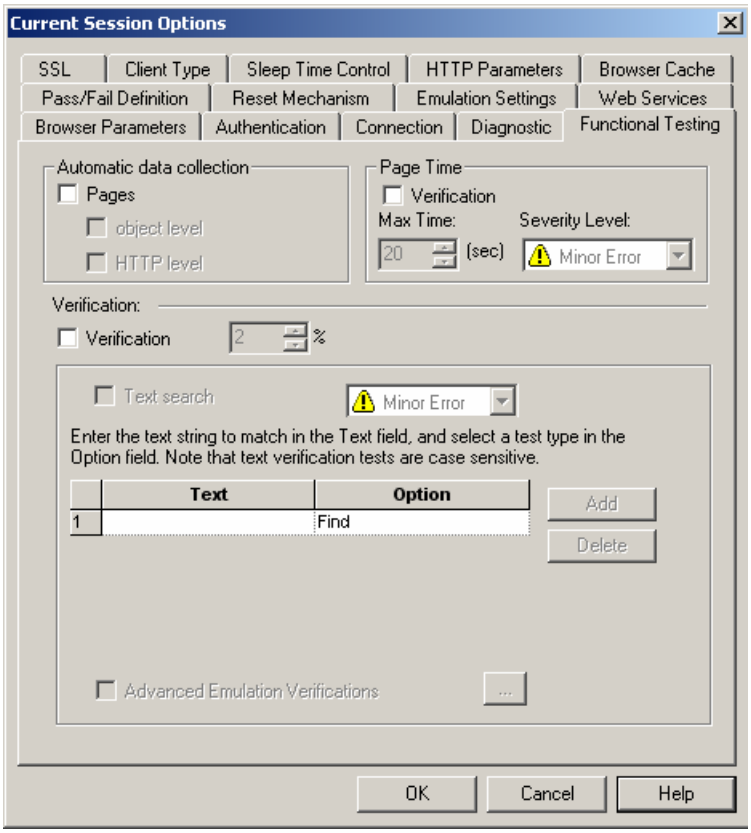
  -Or-

  Select Tools | Current Session Options to set the verification tests to run for the current session,

  -Or-

  Click the Functional Verification icon on the Console toolbar to set the verification tests to run for the current session.

The Functional Testing tab opens.



No global verification tests are selected by default.

The Functional Testing tab enables you to insert verification tests into the Agenda that verify that the links, images, forms, tables, titles and frames on your Web application continue to act as expected. You can also set tests to verify selected text and to report the default values presented on each form.

Note: Tests set through the Current Sessions Options override the Default Options settings.

You can choose to compare any or all of the following:

- Each Web page title.
- For each collection of forms, frames, images, and links, the URL and the number of each object within each collection, as well as the number of elements found on each form.
- The number of tables and/or structure.

You can also:

- Search for a specific user-defined string throughout the Agenda.
- Report the default values for each element in all forms found in the Agenda.
- Select the tests you want to run and specify the severity level of the test results, which determines how the Agenda will continue to execute.

For information on severity levels, see Verification Function Return Values (on page 266).

Selecting the Tests to Perform

Once the Functional Testing tab is open, you must decide which tests you want to perform. The tests are performed on all pages in the Agenda.

WebLOAD enables you to select specific tests to run.

In each case you can specify the severity level in case of test failure, which determines whether the Agenda will continue to execute or not.

For full details, see Verification Function Return Values (on page 266).

Verifying Text

Use Text verifications to verify the absence or presence of selected text expressions within your Web page.

To insert a text verification test into your Agenda:

1. Open the Functional Testing tab, as described in Opening the Functional Testing Tab (on page 260).
2. Select the Verification checkbox.
3. Select the percentage of time the tests should be performed. This is the percentage of Virtual Clients that will perform the verification tests.
4. Mark the checkbox adjacent to the Text Search test.

   Note: Marking a lower level checkbox selects and highlights the upper level checkbox.

5. Select the severity level from the drop-down list next to the test. For further information, see Verification Function Return Values (on page 266).
6. Enter a literal string with the text for which you are searching in the Text field.

   Note: The text string is case sensitive.

7. Choose the type of test to run from the drop down list in the Option field:
   - Text to Find - the verification test will pass if the selected text is found in the current Web page.

- Text not to Find - the verification test will fail if the selected text is found in the current Web page.

For example, to ensure that the word "error" does not appear on the page during runtime, enter the word "error" in the Text field, and select Not to Find. If the text error is found on the page, WebFT will report the error.

8. Click Add to add another string to the list of Text Search strings.
9. Click Delete to delete a selected string from the list of Text Search strings.
10. Click OK.

Selecting Specific Tests to Run

To select specific tests to run:

1. Open the Functional Testing tab, as described in Opening the Functional Testing Tab (on page 260).
2. Select the Verification checkbox.

Note: Marking a lower level checkbox selects and highlights the upper level checkbox.

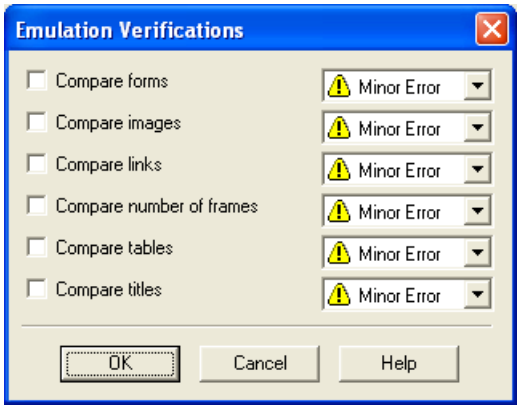
3. Select the percentage of time the tests should be performed.

   This is the percentage of Virtual Clients that will perform the verification tests.

   The extra fields used when defining text verification tests are described in Verifying Text (on page 262).

4. Select the Advanced Emulation Verifications checkbox, and click the [...] button.

   The Emulation Verifications dialog box appears.



5. Select the one or more checkboxes (for information on each test, see Functional Verification Test List (on page 264)).
6. Select the severity level for each selected test (for information on severity levels, see Verification Function Return Values (on page 266)).
7. Select individual tests to perform by marking the checkbox adjacent to the test.

   For an explanation of each of the tests available, see Functional Verification Test List (on page 264).
8. Select the severity level from the drop-down list next to the test. For further information, see Verification Function Return Values (on page 266).
9. Click OK.

Functional Verification Test List

The WebLOAD Functional Testing tab enables you to configure the following tests:

Test	Description
Compare Forms	Verifies the number of forms on each page, the URL of each form, and the number of elements per form.
Compare Images	Validates the images found on each page by image count or by URL.
Compare Links	Checks the number of links on each page and verifies the link URL.
Compare Number of Frames	Verifies that the number of frames received by the Virtual Clients on each page while the test is running is the same as the number of frames found on each page when the Agenda was recorded.
Compare Tables	Verifies the number of tables on each page, as well as the structure of each table.
Compare Titles	Verifies that the page titles received by the Virtual Clients while the test is running are the same as the “expected” page titles which were recorded for the same transaction.

User-Defined Transactions and Verification

Using WebLOAD, you can define application-level transaction and verification criteria to determine whether the transaction succeeded or failed. When a user-defined transaction executes, WebLOAD initiates a timer and a counter. The timer measures the time required to execute the transaction. The counter tracks the success and the failure rate of the transactions, based on user-defined criteria. For each transaction, the counter automatically increments either the “successful transaction count” or the “failed transaction count.” WebLOAD also lets you define the behavior of the program depending on the transaction results.

WebLOAD tracks all of the results from user-defined transactions and integrates the data into the WebLOAD Statistics Report, including the timer and counter statistics as well as statistics on specific failures. The results are then automatically displayed with the other WebLOAD statistical information.

Including a User-Defined Transaction in your Agenda

Application-level transactions and verifications are defined through the WebFT Agenda editing functions. See the *WebFT User's Guide* for more information.

User-Defined Transactions and Verification Syntax

```
BeginTransaction(“<Transaction-Name>”)
     <any valid JavaScript statements>
EndTransaction(“<Transaction-Name>”,
               <Verification-Expression>,<Save-Flag>)
```

Where:

- `Transaction-Name` is a user-supplied string.
- `Verification-Function` (optional) is the name of the function to be called that evaluates the transaction severity level (MinorError, Error, SevereError or Custom Function). If no such expression is present, the default value is Success.
- `save-flag` is an optional Boolean flag specifying whether WebLOAD should save only the results of problematic transaction instance that triggered an error, (default) or save all transaction instances.

User-Defined Transaction Example

The following example illustrates implementing a user-defined transaction:

```
BeginTransaction("UpdateBankAccount")
    /* The user-defined transaction “UpdateBankAccount” starts*/
try {
wlHttp.ExpectNavigation("http://…")
wlHttp.Navigate("www.....com")
wlHttp.SyncDOM(1);

   /* The body of the transaction*/
```

```
<any valid JavaScript statement>

EndTransaction("UpdateBankAccount",
               UpdateBankAccount_VerificationFunction())
function UpdateBankAccount_VerificationFunction()
{
   return WLSuccess;
}
    /* The user-defined transaction “UpdateBankAccount” ends*/
```

Note: The user-supplied `UpdateBankAccount_VerificationFunction()` is called to determine whether the transaction succeeded or failed. This function can be defined either in the Agenda or in an included file.

If during the test the above transaction succeeded 15 times and failed 5 times, and it took an average of 1.2 seconds for the transaction to execute, the Statistics Report displays as follows:

Total UpdateBankAccount transactions	20
Successful UpdateBankAccount transactions	15
Failed UpdateBankAccount transactions	5
UpdateBankAccount timer	1.2 sec

Note: The transaction timer measures the time it took to execute the code between the `BeginTransaction` statement and the `EndTransaction` statement.

Verification Function Return Values

You define the pass/fail criterion for user-defined transactions. For each verification test you must also specify the severity level of a failure. The severity level determines the execution path when the main Agenda resumes control. Thus, depending on the severity level, you can program the test to ignore the failure, jump to the next navigation block or stop the test. Extreme failures with high severity levels can be set to abort the test, whereas less severe failures can be ignored.

You set the severity level in the return statement of the verification function. The assigned severity levels and their results are handled in the same manner as other test failure results in WebLOAD. All failures are logged and are displayed in the WebLOAD log window. Refer to Appendix A *(see "WebLOAD Error Messages" on page 397)* for more details on severity levels.

Transactions may be assigned one of the following return values:

Severity Level	WebLOAD Behavior
Success	The transaction completed successfully.

Severity Level	WebLOAD Behavior
MinorError	The specific transaction failed, but the Agenda continues to execute from the following statement.
Error	The specific transaction failed. The Agenda continues from the next navigation block.
SevereError	The specific transaction failed. The load generator on which the error occurred is stopped.

Note: If a return value (`return()`) is not specified, the default value is Success, indicating that the transaction completed successfully.

Reporting Events

WebLOAD enables you to record specific events and transaction instances as they occur. This information is very useful when analyzing Web site performance with Data Drilling, described in Data Drilling and Table Compare Reports (on page 335).

Record transaction events with the following syntax:

```
ReportEvent (“<event-name>”, [“<description>”])
```

Where:

- `event-name` is a string that identifies the specific event.
- `description` is an optional string providing more information about the specific event.

Tracking Reasons for Transaction Failure

WebLOAD enables you to specify possible reasons for the failure of a given transaction. WebLOAD tracks the reasons for failure individually, in the Statistics Reports. The reasons for the transaction failure must be specified in the verification function.

The default reason for failure of basic transactions is General-Failure. Unless another reason for failure is specified, this is set automatically whenever the transaction fails.

Set the reason for a transaction failure using the following syntax:

```
SetFailureReason (“<failure-reason>”)
```

For example:

Continuing the `UpdateBankAccount` transaction example described in User-Defined Transactions and Verification (see "User-Defined Transactions and Verification Syntax" on page 265), the `UpdateBankAccount` verification function might appear as follows:

```
function UpdateBankAccount_VerificationFunction()
```

```
{
   VerifyUpdateBankAccount () {
      if <Condition-success> then
         return(WLSuccess)
      else
      if (document. URL == "../no-login") then {
         SetFailureReason("User Not Logged")
         return(WLError) }
      else
      if (document.URL == "...wrong-   password") then {
         SetFailureReason("Wrong Password")
         return(WLError); }
      else
         return(WLSevereError)
}
```

Explanation:

- `Return(WLSuccess)` - the actual transaction succeeded, it is added to the `SuccessfulUpdateBankAccount` transactions counter.
- The `SetFailureReason()` function accepts a string as a parameter. This string is used to identify the cause of the transaction failure ("User Not Logged", "Wrong Password", etc.). This data enables tracking of the number of failures that occur for a specific reason, as well as the time that the failures occurred.
- When the reason for failure is "User Not Logged" or "Wrong Password", the severity level is Error. Consequently, when the main Agenda resumes control the current round aborts and a new round commences.
- If you mark the transaction as failed, but do not specify any failure reason, the system registers a "General Failure" which is the default failure severity.

If during the execution of a test the above transaction succeeded 15 times and failed 5, and the `SetFailureReason()` function was called 3 times with the string "Wrong Password" and 2 times with the string "User Not Logged", the Statistics Report appears as follows:

Total UpdateBankAccount transactions	20
Successful UpdateBankAccount transactions	15
Failed UpdateBankAccount transactions	5
Failed UpdateBankAccount transactions: Wrong Password	3

Failed UpdateBankAccount transactions: User Not Logged	2
UpdateBankAccount timer	1.2 sec

All of the selected tests are performed on all pages in the Agenda when you run a WebLOAD test session. Verification test failures are automatically added to the Data Drilling reports. All verification test failures are also displayed in the Event Viewer that highlights the "expected" and "actual" results, described in Viewing Verification Test Results (on page 269).

Viewing Verification Test Results

After running your Agenda, WebLOAD provides information on all major events that occurred during runtime such as verification failures and error messages.

When a verification test fails:

- The failure is logged in the Log Window, described in Viewing Error Information in the Log Window (on page 269).
- The failure is added to the Data Drilling report, described in Data Drilling (on page 271).

All verification failures can be viewed in the Event Viewer, which displays the "expected" results, the "actual" results, and a description of the error. The Event View can be accessed through either the Log Window or the Data Drilling reports, as described here.

Viewing Error Information in the Log Window

If an error occurs at any time during the test session, an error message displays in the Log Window. A browser error or a minor error will not cause the test session to stop. A higher level of severity (Error or Severe Error) ends the session upon completion of the current navigation block.

A description of each error is listed in the Log Window. You can toggle between displaying or hiding the error log by selecting View | Log Window.

The Log Window displays the following additional information about your test session results:

- Message Status - the result and severity of each message.

  Messages fall into three categories:

  - Success / Info Message
  - Minor Error
  - Error

- Attachments - double-click on the paperclip adjacent to the message to view additional information.
- Generator Name - The generator that originated the message.
- Agenda Name - The Agenda that originated the message.
- Message - The runtime action and information about failed actions.

To view more information on an error:

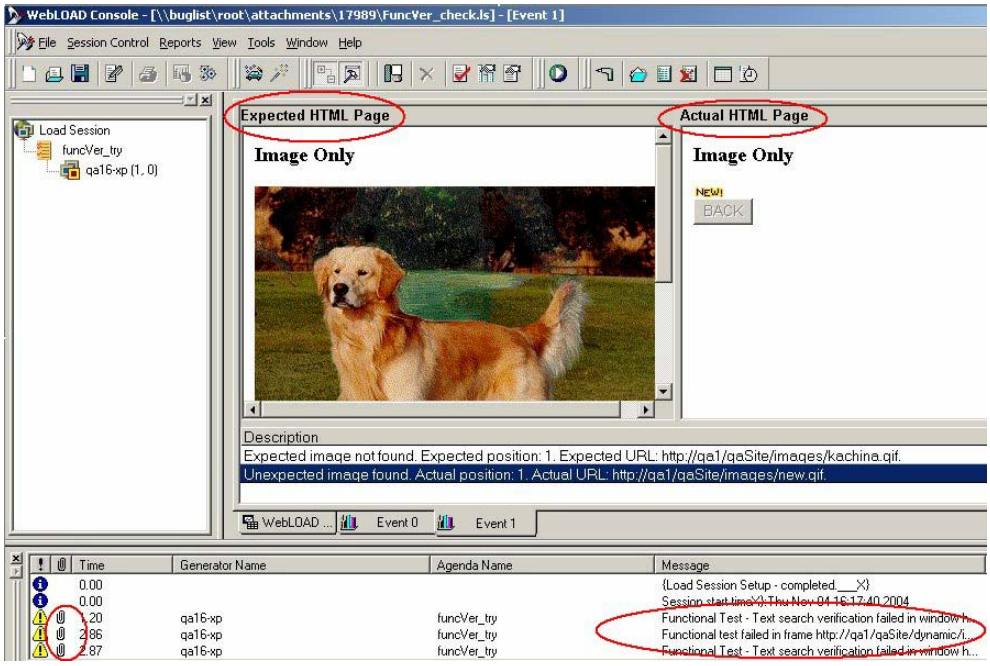
- Double-click on the paperclip adjacent to the verification failure in the Log Window.

  The Event Viewer opens.

Note: The Event Viewer is only available after the test session is complete.

Viewing Error Information in the Event Viewer

When you double-click on a verification failure in the Log Window, the Events Viewer opens. The Event Viewer displays the Expected and Actual views of the object that triggered the failure, with the object highlighted. A description of any errors that occurred during the test run appears below.



Double click on the description below the viewer to highlight any changes relating to the error. If the object you are testing for is not found, it is highlighted on the expected page only. If the object is changed, it is highlighted on both pages.

Printing the Contents of the Log Window

To print the contents of the Log Window:

1. Right-click inside the log window.
2. Select Print from the right-click menu.

   The Print Setup dialog displays.
3. Select a printer and press OK.

Saving the Contents of the Log Window

To save the contents of the Log Window:

1. Right-click inside the log window.
2. Select Save from the right-click menu.

   The Save As dialog displays.
3. In the File Name field, type in the name for the file.
4. Press Save.

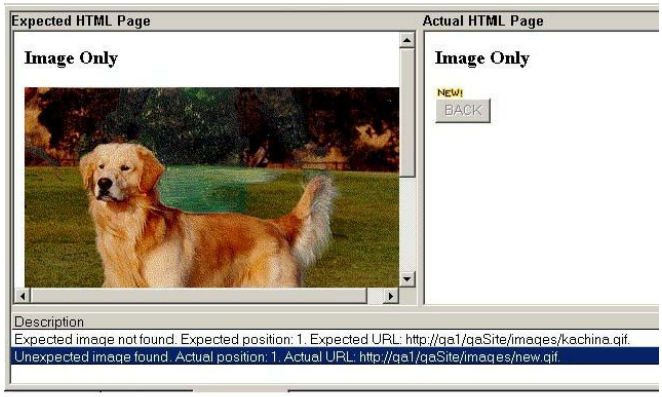
   The file is saved with the extension *.log.

You can view the saved log file with any text editor.

Data Drilling

When testing the integrity of your Web application, it is important to see not only if a given transaction succeeded or failed, but also the reasons for the failure to pinpoint the transaction's exact weak points and bottlenecks. Data Drilling enables you to display a detailed description of all user-defined and named transactions to the instance level.

With Data Drilling you can view all transaction failures and the reasons for them. Data Drilling reports provide a complete breakdown of all information about a selected event. Click on an event entry to bring up detailed information about that event in the Event Viewer. The Event Viewer provides a graphic illustration of the verification failure event, displaying the “expected” results, the “actual” results, and a description of the error.



Data Drilling is described in detail in Data Drilling and Table Compare Reports (on page 335).

Note: Data Drilling is only available to you if:

You initiated user-defined transactions in the scripting for functional testing.

You configured WebLOAD to Collect Instances in the Tools | Functional Testing tab.

You are using Internet Explorer 4 or higher.

C H A P T E R 1 8

Running Tests and Analyzing Test Results

Running test and analyzing the results introduces you to WebLOAD's performance reports.

In This Chapter

Explanation of Creation and Analysis of Performance Reports . 273

Explanation of Creation and Analysis of Performance Reports

Performance Reports are the product of your WebLOAD testing. WebLOAD collects all of the data for a wide variety of measurements. You can view and configure these reports through the Console, while tests are running, or after the Load Session is completed.

This material is divided into six chapters:

- Running a Load Session (on page 275) provides instructions for running a WebLOAD test session.
- Viewing the Log Window (on page 287) describes the WebLOAD Log window.
- WebLOAD Performance Reports (on page 293) provides instructions for creating and viewing WebLOAD Integrated Reports.
- WebLOAD Statistics Reports (on page 313) describes the statistics reports provided by WebLOAD.
- Data Drilling and Table Compare Reports (on page 335) describes how to view all transactions failures and the reasons for them.
- Exporting Performance Reports (on page 345) describes how to export all of the data displayed in the Integrated Reports and Statistics to HTML, Microsoft Excel and to Tab files.
- Performance Measurements Manager (on page 365) describes how to configure the Performance Measurements Manager to collect data from different sources during the test.

After completing this material, you will have the skills necessary to create and analyze all of the performance reports provided by WebLOAD.

C H A P T E R 1 9

Running a Load Session

After configuring a Load Session, you can run your Load Session. This chapter describes the procedure for running your Load Session, and the way WebLOAD behaves while a Load Session is in progress.

WebLOAD collects a complete set of test data while a Load Session is running. You can configure reports that include only the data needed. Reports can be configured during or after a test.

WebLOAD run commands can start, pause, and stop your Load Session while it is in progress. These commands are detailed in this chapter.

In This Chapter

Starting a Load Session 275
Establishing Communication 278
The Console Screen in Session Mode 281
WebLOAD Operating Commands 282
Throttle Control - Changing the Load Size Dynamically During Runtime 284

Starting a Load Session

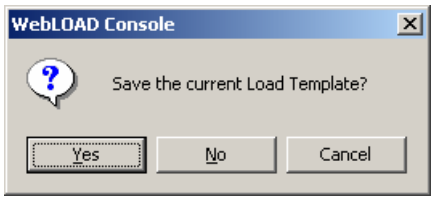
You can start a Load Session in one of the following ways:

- Click the Start icon in the Console toolbar.
- Select Session Control | Start Session.
- From the WebLOAD Wizard - Finish dialog box, select the checkbox to run the Load Session immediately and click Finish.
- From the Cruise Control - Finish dialog box, select the checkbox to run the Load Session immediately and click Finish.

Running a Load Session from the System Memory

You can run a test from the configuration parameters in your system memory, without saving the Load Session. If you start running a Load Session without saving the Load Session configuration, you are prompted to save the Load Template.

The following message box appears.



Saving the Current Load Template

To save the current load template:

- Click Yes.

  The test session definition is saved to a Load Template (*.TPL) file.

Scheduling WebLOAD Tests with TestView Scheduler

Using TestView Scheduler you can easily create a schedule to automatically run WebLOAD load sessions at specified times.

Note: WebLOAD must be closed before a scheduled test can run.

For information on scheduling WebLOAD tests with TestView Scheduler, see the *TestView Suite User's Guide*.

Launching WebLOAD through the Command Line Interface

You can initiate testing through a command line interface. You can enter the WebLOAD launch command into a batch file or into an external script and WebLOAD will run directly, without user intervention, using the parameters specified. This will enable you to perform unattended WebLOAD testing at prescribed times.

To launch WebLOAD through the command line interface:

- Enter the command:

  `WebLOAD.exe`

  together with a series of optional parameters (described below) into your external script to automatically launch a WebLOAD test. When your script runs, the executable file will invoke WebLOAD and run the specified test according to the specified parameters.

Note: Verify that the Agenda used with the template specified, and any included files, are accessible to the Load Template or Load Session file that will be run.

Parameters

When running a test invoked by the executable, you can specify the following parameters:

Parameter	Description
File name to open	The name of the TPL or LS file (Load Template or Load Session file) to run.
File name to save	The name of the LS file containing the test data. This file will be saved in the current directory unless otherwise specified.
Flags	/ar - auto run Automatically run the WebLOAD test without waiting for user input. If this flag is not specified, the Console is called up with the specified Load Template/Load Session but the test is not automatically run. The system waits for user input.
Time	The length of time (in seconds) that the test is to run.

Parameters are all optional. If no parameters are entered, the executable launches the Console and does not run a test. If the autorun flag </ar> flag is not set, the <file name to save> and the <time> parameters are ignored.

Note: If there is a conflict between the time defined in this command and the time defined in the TestView Scheduler, the load test runs for the shorter of the two periods defined.

Syntax

Use the following syntax to define the parameters for running a WebLOAD test through a command line interface:

```
WebLOAD.exe [<file name to open>][<file name to
save>][<flags>][<time>]
```

Examples

Example 1:

```
WebLOAD.exe test1.tpl
```

This command opens the Console and the `test1.tpl` template. The Console waits for user input.

Example 2:

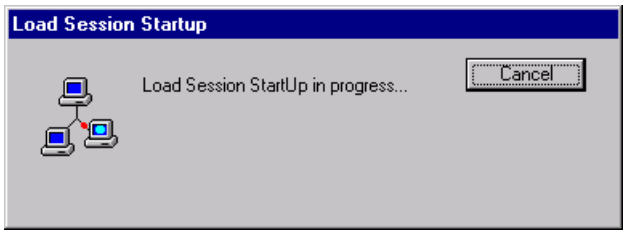
```
WebLOAD.exe test1.tpl march9.ls /ar 30
```

This command opens the Console and automatically run a test using the `test1.tpl` template file. The test results are saved in the Load Session file `march9.ls`, which includes all of the test data and results. This file is saved in the current directory, unless otherwise specified. The autorun flag is set, meaning that the test runs without user intervention. The test will run for 30 seconds.

Establishing Communication

WebLOAD begins executing a test by verifying the test parameters and attempting to communicate with the hosts participating in the test session.

While WebLOAD is preparing to run the test, the following message appears:



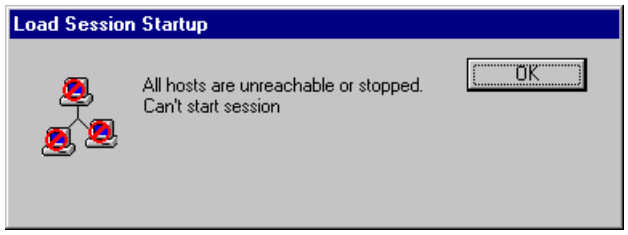
Errors in Communication

The following errors may occur during Load Session startup:

- All hosts are unreachable or stopped
- Load Session setup not completed
- Load Session setup timed out

Correcting the All Hosts are Unreachable or Stopped Error

When the All hosts are unreachable or stopped error occurs and the session cannot be started, the following error message appears:

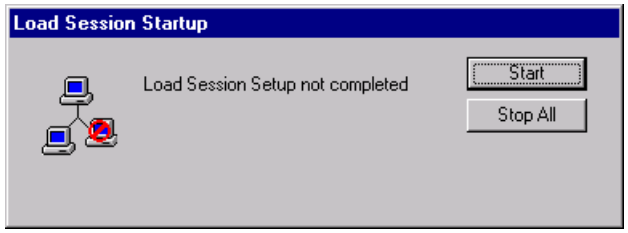


To correct the all hosts are unreachable or stopped error:

1. Click OK.
2. Test your connection. Verify that the host systems participating in the test are all up and running. If hosts are still unreachable, inform your system administrator.

Correcting the Load Session Setup Not Completed Error

When only some of the requested hosts are active, the 'Load Session setup not completed" error message appears as follows:



To correct the Load Session Setup not completed error:

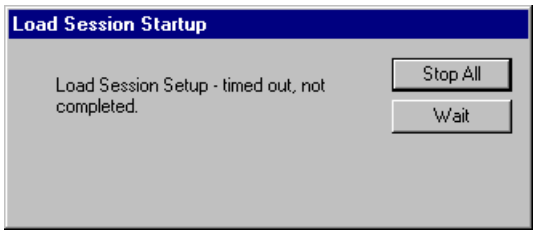
- Click Start to begin testing with Hosts that have been reached,

  -Or-

  Select Stop All to stop testing.

Correcting the Load Session Setup Timed Out Error

When the Load Session Setup has timed out, the following error message appears:



To correct the Load Session Setup timed out error:

- Select Stop All to stop testing,

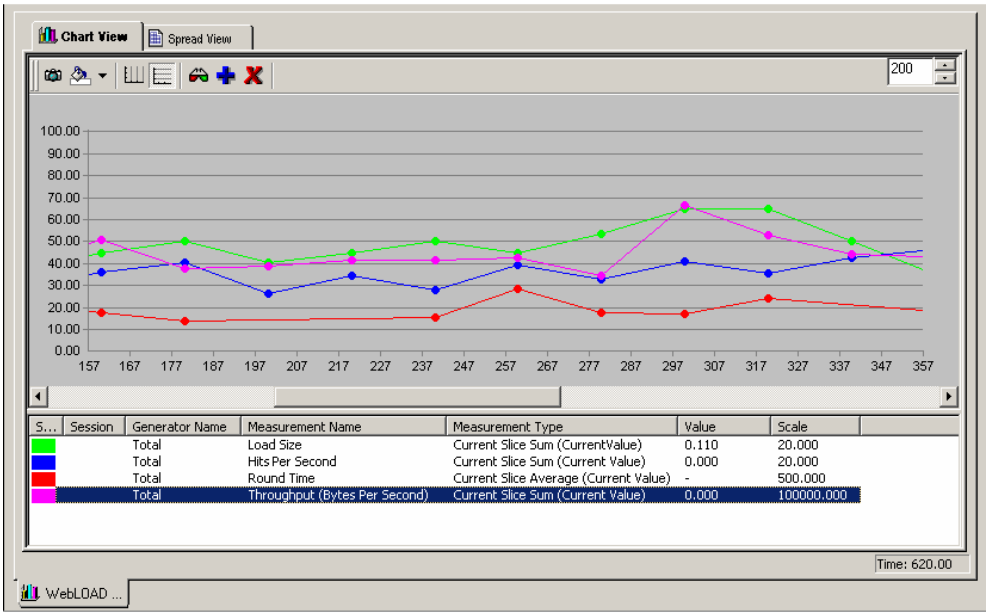
  -Or-

  Select Wait to give WebLOAD more time to connect to the unreachable Hosts. WebLOAD allows another interval of time equal to the timeout interval, for the systems to connect.

WebLOAD has a built-in start session timeout value, which is the length of time WebLOAD waits while TestTalk attempts to make contact with the hosts. You can change the timeout values through the Tools | Global Options | General tab, see The General Tab (on page 205) for details.

The Console Screen in Session Mode

When you start running your Load Session, the Results window appears.



The following table describes the segments of the Console in session mode:

Segment	Function
Session Run Toolbar	Displays only during a test run. It contains buttons for controlling the test session.
Session Tree	Presents a graphic display of your test session. See Viewing the Session Tree in Session Mode (on page 282) for a complete discussion of the Session Tree.
Results Window	Displays all of the reports opened during a test session. Use the tabs located at the top and the bottom of the window to view different reports.
Log Window	Displays all of the error messages recorded during a test session. You can toggle the Log Window display on/off through the View menu.
Status Bar	Indicates the program status. You can toggle the Status Bar display on/off through the View menu. When running the WebLOAD Workstation, the Standalone Workstation icon or the Connected Workstation icon displays in the lower right-hand corner.

Viewing the Session Tree in Session Mode

The Session Tree displays the complete configuration of the current Load Session. Status icons reflect the current status of your Load Session.

Through the Session Tree you can see:

- The Agendas running
- The hosts running each Agenda
- Each hosts and Agendas current operating status

Icons display on each line of the tree, making it easy to view your test activity.

When a host is connected, the word “CONNECTED” appears to the right of the host name. While the Load Session is running, status icons reflect the current status for each script. The status icons are described in the above table.

To the right of each Agenda two numbers are displayed in parentheses, for example Agenda1[45, 75]. These display the maximum Load Size (number of Virtual Clients) that Agenda is scheduled to generate, followed by the Load Size currently being generated. The current Load Size number changes according to the Load Size schedule that is defined in the Load Generator Schedule dialog box. When a test session has not yet begun, the 'current' Load Size being generated is 0. In most modes of operation, the current Load Size never exceeds the maximum Load Size scheduled for that Agenda listed to the left of the current value.

Note: Using Throttle Control, you can increase the current Load Size to exceed the maximum Load Size - but only if you are not using Cruise Control.

WebLOAD Operating Commands

WebLOAD offers five commands for controlling a test session:

- Start
- Pause
- Go
- Stop
- Stop All

You access them through the Console toolbar, the Session Tree popup menus, or through the commands listed in the Session Control menu.

Starting a Test Session

To start a test session:

- Click the Start Session toolbar button,

  -Or-

  Select Session Control | Start Session.

Pausing a Test Session

To pause a test session:

- Click the Pause toolbar button,

  -Or-

  Right-click on a component in the test session tree and select Pause from the pop-up menu,

  -Or-

  Select Session Control | Pause.

When you pause an Agenda it temporarily stops the execution of that Agenda. The remainder of the components in the current test session (other Agendas running simultaneously) will continue running as scheduled.

When a paused Agenda is continued (using the Go / Resume command), the Agenda skips any actions scheduled for the duration of the pause. Agenda execution continues with whatever load was scheduled for the current time.

Pausing a Host that is generating a heavy load causes an immediate jump in system response time because the system load drops immediately.

Resuming a Test Session

To resume a test session:

- Click the Go toolbar button,

  -Or-

  Right-click on a component in the test session Tree and select Go,

  -Or-

  Select Session Control | Resume.

Resuming a Host that is generating a heavy load causes an immediate drop in system response time.

Stopping a Load Session Component

To stop a load session component:

- Right-click on a component in the test session Tree and select Stop,

  -Or-

  Select Session Control | Stop.

The selected component will stop. Once an individual component is stopped, it cannot be restarted unless the whole Load Session is restarted.

You may stop some of the Load Session components without affecting the remaining ones.

Stopping a Load Session component affects system performance by causing a sudden drop in system load, which leads to a corresponding jump in system performance.

Stopping a Test Session

To stop a test session:

- Click the Stop toolbar button,

  -Or-

  Right-click on a component in the Load Session Tree and select Stop All,

  -Or-

  Select Session Control | Stop All.

Stopping a complete Load Session affects system performance because the system load immediately drops to zero.

Throttle Control - Changing the Load Size Dynamically During Runtime

Using Throttle Control utility, you can override the test schedule and dynamically control the Load Size during runtime. You can select a component (from the Session Tree) and modify Load Size - the number of Virtual Clients participating in the test.

The selected component can be:

- The root of the Session Tree
- A Load Machine

When using Throttle Control consider the following:

- You cannot modify an Agenda (test script) using Throttle Control.
- You cannot use Throttle Control with Cruise Control.
- You cannot use Throttle Control to create a load of more than 20 users in Standalone Workstation mode.
- If you select the root from the Session Tree and you are testing with more than one Load Machine, WebLOAD distributes the load among the Load Machines. During this process, the original schedule is "suspended" for the duration of Throttle Control.

Opening Throttle Control

To open Throttle Control:

- Click the Throttle Control toolbar button,

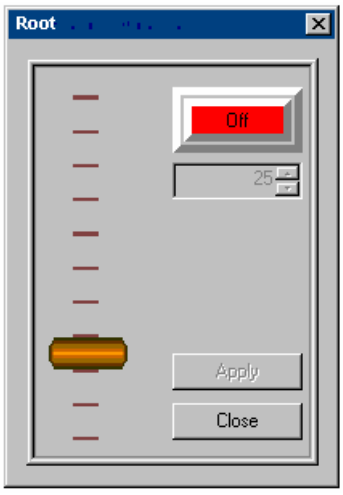
  -Or-

  Right-click on a Session component and select Throttle Control from the pop-up menu,

  -Or-

  Select Session Control | Throttle Control.

  The Throttle Control dialog box appears.



The Throttle Control dialog box displays:

- The current Load Size
- The currently selected component. (In the figure above, the selected component is the Root of the Session Tree).

Activating Throttle Control

To activate Throttle Control:

1. Toggle the Off button to On.
2. Adjust the Throttle by moving the lever up or down and set the desired Load Size.
3. Click Apply.

   The Load Size changes dynamically while the test is running.

Deactivating Throttle Control

To deactivate Throttle Control:

1. Toggle the button On to Off.
2. Click Close.

   The Throttle Control dialog box is closed. Once Throttle Control is turned off, the Load Session resumes the manual schedule and continues running from the point at which Throttle Control was de-activated.

C H A P T E R 2 0

Viewing the Log Window

The Log Window displays log messages detected by WebLOAD in run time which are generated by the Console, the JavaScript compiler and any user messages programmed in the test script.

In This Chapter

Setting the Log Window Options 287
Opening the Log Window 288

Setting the Log Window Options

Before the Load Session begins, you can set the options that affect the way the data is displayed in the Log Window.

To set the log window options:

1. Select Tools | Global Options | General tab.
2. Click one of the following radio buttons:
   - Stop Session after n log message
   - Stop sending log messages after n log messages
3. Click OK.

the following table describes the log window options:

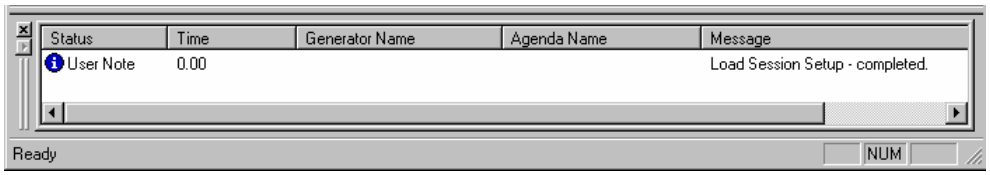
Field Name	Description
Stop Session after n log messages	Stop the Load Session after a defined number of log messages have been received. Once the maximum defined number of log messages is received, a message box displays stating that the Load Session is stopped.
Stop sending log messages after n log messages	Stop displaying log messages in the log window after a defined number of log messages have been received. Once the maximum defined number of log messages is received, a message box displays stating that no more log messages are displayed.

Opening the Log Window

To open the log window:

- Select View | Log Window.

  The Log Window appears.



The Log Window displays:

- A list of the log messages received during the test session.
- The status of each log message.
- A paper-clip symbol which when you double-click on it opens the Table Compare Viewer.
- The time each log message was generated (in seconds) from the start of the current Load Session.
- The Load Generator and Agenda for which each log message is generated.

Viewing the Entire Log Message

To view the entire log message:

- Double-click on any of the messages displayed in the Log Window.

  A Log Message message box appears displaying the complete message. The text displayed in the Log Message message box can be selected and copied to a text editor.

Sorting Error Messages

To sort error messages:

- Click on the column heading.

  The first time you click on a column heading, the error messages list in ascending order. A second click on the column heading reverses the order. For example, to sort the error messages by Agenda Name, click on the Agenda Name column heading.

Opening the Log Window Menu

To open the log window menu:

- Right-click inside the Log Window.

  The Log Window pop-up menu appears.

The following table describes the Log Window menu commands:

Command	Description
Allow Docking	Enables/disables moving the Log Window.
Hide	Does not display the Log Window.

Command	Description
Clear	Erases data in the Log Window.
Show last message	Displays the final messages in the logged in the Log Window.
Print	Prints the data in the Log Window.
Save / Save As	Saves the data in the Log Window.
Report Defect	Opens the Submit Defect dialog box with the selected log message displayed in the Description field.
Float in Main Window	Floats the Log Window in the main WebLOAD Window.

Printing Error Messages Recorded in the Log Window

To print error messages recorded in the log window:

1. Right-click inside the Log Window.

   The Log Window pop-up menu appears.
2. Click Print.

Saving Error Messages Recorded in the Log Window

To save error messages recorded in the log window:

1. Right-click inside the Log Window.

   The Log Window pop-up menu appears.
2. Select Save or Save As.

   The Windows Save As dialog box appears.
3. Type the name of the file in the File Name box.
4. Click Save.

   The file is saved as a *.log ASCII file. The file can be opened in any editor.

Viewing the Message Status

Messages fall into four categories:

- User Note
- Warning
- Error
- Severe

The log messages include messages generated by the Console, the JavaScript compiler and any human errors or user messages detected by WebLOAD at run-time. You can also add log messages to the Agenda.

User Note

User Notes are messages added to the Agenda by the user. User Notes can be used in order to facilitate debugging.

They do not stop an active Load Session.

Warning

A Warning indicates a suspicious condition. The source of the problem may be generated by the application being tested, the connection to the server, networking, etc.

Warnings do not stop an active Load Session.

Error

When an Error is detected WebLOAD will stop the round and restart the Agenda for the Virtual Client that encountered the error.

Errors do not stop an active Load Session.

Severe

Severe errors cause the Load Generator, on which the error was generated, to stop immediately.

An error message indicating that the process was stopped is displayed.

C H A P T E R 2 1

WebLOAD Performance Reports

The WebLOAD Dashboard lets you view a summary of performance statistics generated during the test session. Integrated Reports enable you to configure a report that combines

- Performance data from the Load Machines and Probing Clients.
- Performance data on the ABT, gathered from Performance Monitors.
- Existing performance data from a previously saved Load Session with data from the currently running Load Session.

Combining several different measurements into a single report gives you a more complete picture of your Web application's performance.

In This Chapter

Viewing Test Results on the WebLOAD Dashboard 294
Opening Reports 295
Opening an Existing Report 298
Editing a Report 298
Renaming a Report 299
Deleting a Report 300
Configuring an Integrated Report 300
Regression Testing 305
Integrated Report Chart View 306
Integrated Report in Spread(sheet) View 310

Viewing Test Results on the WebLOAD Dashboard

The WebLOAD Dashboard displays real-time statistical information about the test session for the following categories:

Category	Description
Time	The time during the test that the values displayed on the Dashboard were sampled.
Running Virtual Clients	The total number of Virtual Clients currently running.
Rounds Executed	The total number of times the Agenda has been executed by all Virtual Clients.
Failed Rounds	The total number of rounds with errors.
Failed Hits	The total number of hits with errors.
Hits Per Second	The number of HTTP requests for information (pages, individual images, objects, frames) made by Virtual Clients to the ABT per second.
Calculated Hits Per Day	The number of HTTP request for information made by Virtual Clients to the ABT per day.
Pages Per Second	The number of upper level request for information made by a Virtual Client to the application being tested (the ABT) per second.
Calculated Pages Per Day	The number of upper level request for information made by a Virtual Client to the application being tested (the ABT) per second.
Throughput Per Second	The average number of bytes per second transmitted from the ABT to the Virtual Clients running this Agenda.
Calculated Throughput Per Day	The average number of bytes per day transmitted from the ABT to the Virtual Clients running this Agenda.

To open the Dashboard:

- Click the Dashboard toolbar button.
- Select Reports | Dashboard.

Opening Reports

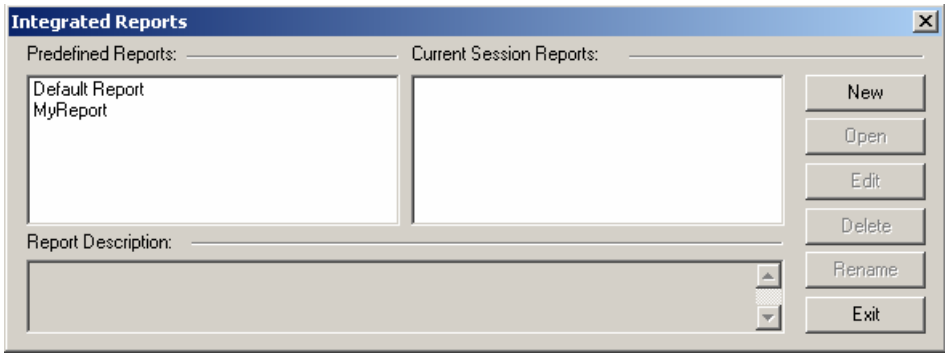
To open reports:

1. Select Reports | Report Manager,

   -Or-

   Click the Open Reports Manager toolbar button.

   The Integrated Report dialog box appears.

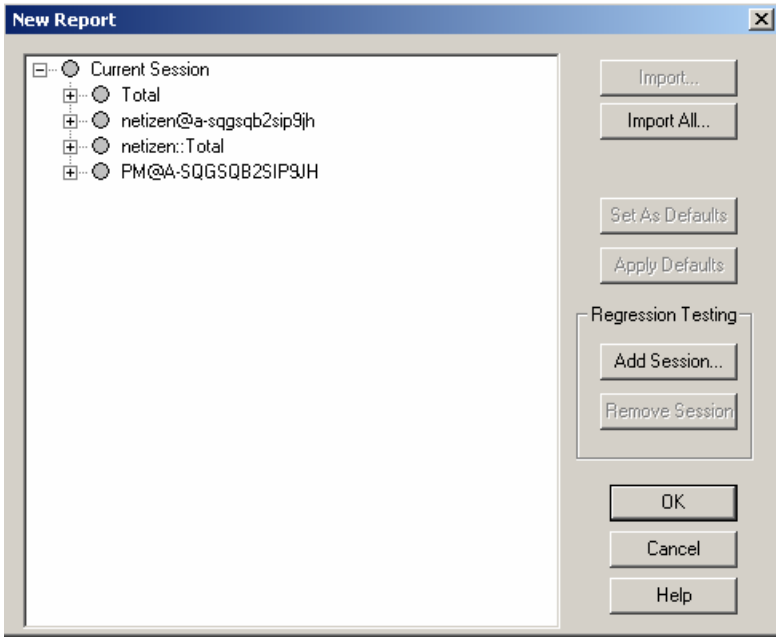


2. Click New.

   -Or-

   Select a report and click Edit.

The Report dialog box appears.



Report data is organized in a tree structure. Report fields are organized into different "report tree branches", illustrated in the preceding figure.

Report tree branches include:

- Total-data collected for the session as a whole, including total load size, total number of rounds, and total number of pages.
- AgendaName@LoadGenerator-data collected for all instances of the named Agenda, running under the named Load Generator. Multiple branches may appear for a single Agenda name, each one linked to a different Load Generator.
- AgendaName:: Total-data collected for all instances of the named Agenda, running under all Load Generators participating in this test session.

Using the Report dialog box you can:

- Configure the items displayed in the test report.
- Rename the report.
- Import an existing report configuration.
- Set a default report configuration.
- Apply a default report configuration.
- Configure a report that compares existing performance data from a previously saved Load Session with data from the currently running Load Session.

The following table describes the fields in the Report dialog box.

Button	Functionality
Import	Imports a report configuration for the selected component in the configuration tree, from a previously saved Load Template. See Importing a Report Configuration for a Selected Component (on page 304). Note: Select an upper level component in the configuration tree to enable this button.
Import All	Imports a report configuration from a previously saved Load Template, see Importing a Complete Report Configuration (on page 303).
Set as Defaults	Sets the current Integrated Report configuration as the default. This overrides the built-in default configuration. Note: Select an upper level component in the configuration tree to enable this button.
Apply Defaults	Apples the default configuration. The defaults are set using the Set as Default option. Selecting this option configures the Integrated Reports to the defaults. Note: Select an upper level component in the configuration tree to enable this button.
Regression Testing	
Add Session	Imports performance data from a previously saved (existing) Load Session, see Importing Report Parameters from a Saved Load Template. (see "Importing Report Parameters from a Saved Load Template" on page 303)
Remove Session	Removes the selected Load Session from the performance data available for display in the report. Note: Select an upper level component in the configuration tree to enable this button.
OK	Confirms the selections made in this dialog box. The selected items are included in the Integrated Report.
Cancel	Closes this dialog box without accepting changes.
Help	Opens the online-help dialog box.

For information on selecting the measurements included in an Integrated Report, see Selecting Measurements for the Integrated Report (on page 302).

Opening an Existing Report

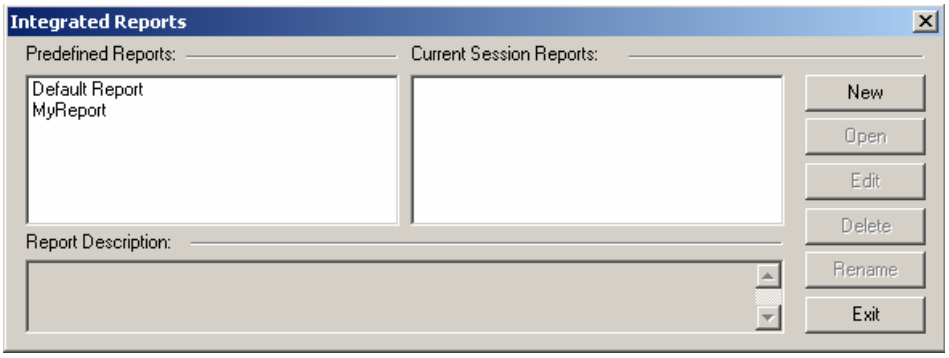
To open an existing report:

1. Select Reports | Report Manager,

   -Or-

   Click the Open Reports Manager toolbar button.

   The Integrated Report dialog box appears.



2. Select a report.

   The Open button is enabled and a description of the measurements included in the report appears at the bottom of the dialog box.

3. Click Open.

   The Integrated Report displays in the Console Results Window. WebLOAD displays the results of your test in both graphical and tabular format. Click the Chart View and Spread View tabs at the top of the Results Window to switch between the two views.

Editing a Report

To edit a report:

1. Select Reports | Report Manager,

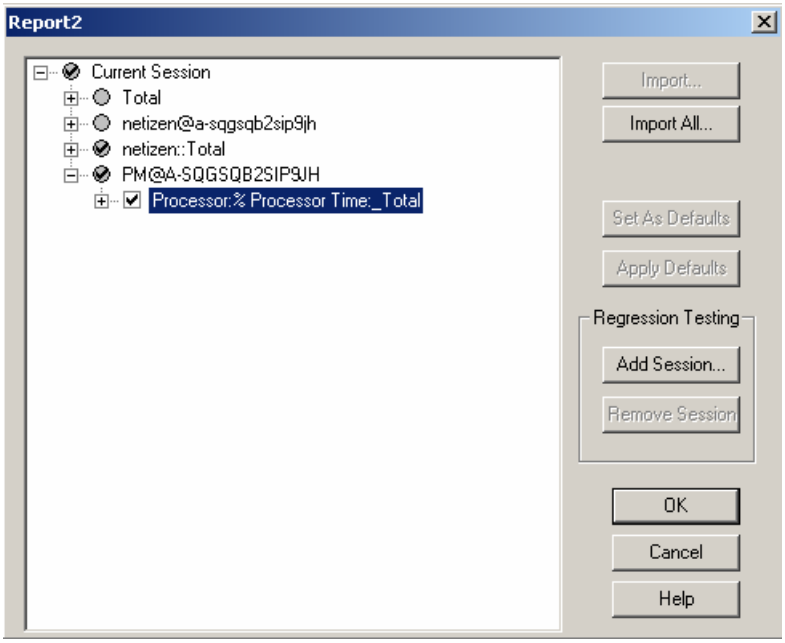
   -Or-

   Click the Open Reports Manager toolbar button.

   The Integrated Report dialog box appears.

2. Select a report and click Edit.

   The Report dialog box appears.



3. Make the necessary changes, and click OK.

Renaming a Report

You can assign a meaningful name to a report. The default name for reports is "ReportX" where X = a number that is incremented with each generated report.

To rename a report:

1. Select Reports | Report Manager,

   -Or-

   Click the Open Reports Manager toolbar button.

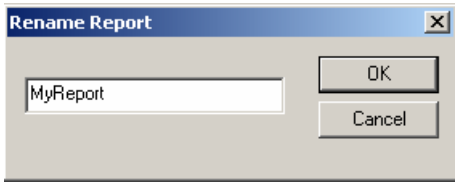
   The Integrated Report dialog box appears.

2. Select a report to rename.

   The Rename button is enabled and a description of the measurements included in the report appears at the bottom of the dialog box.

3. Click Rename.

   The Rename dialog box appears.



4. Type a new name for the report.
5. Click OK.

   The selected report is renamed.

Deleting a Report

To delete a report:

1. Select Reports | Report Manager,

   -Or-

   Click the Open Reports Manager toolbar button.

   The Integrated Report dialog box appears.
2. Select a report to delete.

   A description of the measurements included in the report appears at the bottom of the dialog box.
3. Click OK.

   The selected report is deleted.

Configuring an Integrated Report

You can configure an Integrated Report while the test is running. To view data from the WebLOAD Performance Monitor in your Integrated Report, see Performance Measurements Manager (on page 365).

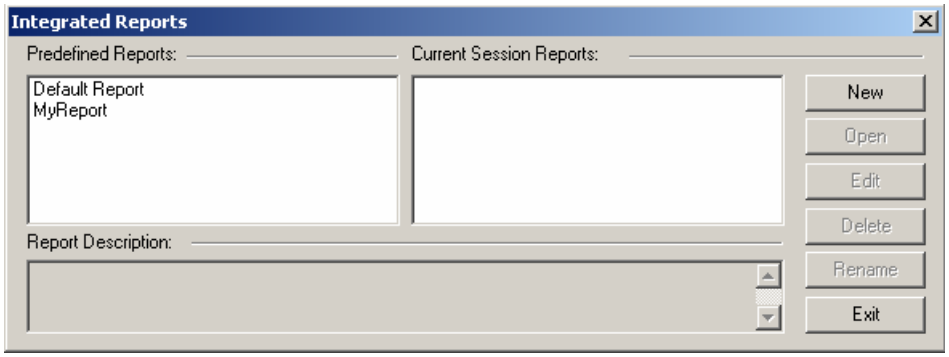
To configure an integrated report:

1. Select Reports | Report Manager,

   -Or-

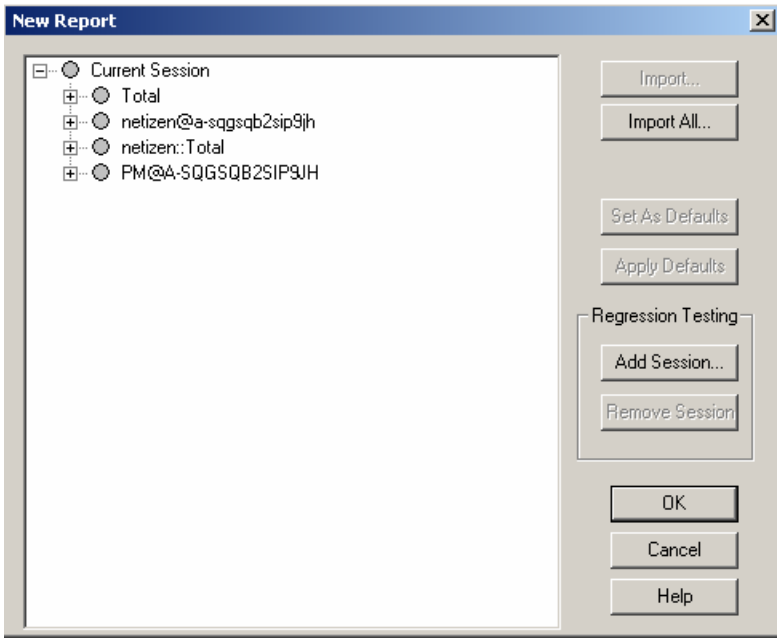
Click the Open Reports Manager toolbar button.

The Integrated Report dialog box appears.



2. Click New.

The Report dialog box appears.



The full tree of measurements, including all of the items that can be included in the report, are displayed in the Report dialog box. In addition, items configured to be collected in the Server Monitor Definition are also displayed. Data for all of the items is accumulated during the test run.

Select the parameters to view in the Integrated Report from the tree structure displayed in the Report dialog box. Only the results for the items that are checked are display in the Integrated Report.

Selecting Measurements for the Integrated Report

The first time the Report dialog box displays, none of the items are checked. If you do not select any of the checkboxes, the Integrated Report is blank.

You can configure the Integrated Report in one of several ways:

- Check the items to be included in the report manually.

  Mark the checkbox adjacent to the measurement that you want to include in the Integrated Report. The options display in tree format. You can expand or compress the tree using the +/- buttons. If an upper level component is selected, all of the default subcomponents in that tree will be selected.

  Note: A gray circle next to a measurement indicates that no default subcomponents are defined (the upper level component cannot be selected). Right-click on a measurement to be included in the report to prompt the pop-up menu that enables you to select the item.

- Select a WebLOAD default configuration by clicking the Apply Defaults button. WebLOAD selects the set of default measurements.
- Select Import All to apply a previously configured Integrated Reports test set, saved as a Load Template to your current test.

  Note: The configuration is imported, even if it was designed for testing a different application.

Changing the Reports Parameters

You can edit the timer options listed in the configuration tree in real time.

While a test is running, you can:

- Add a user-defined timer that was not previously configured.
- Change the scale of the report.

Timer information from Probing Clients and Load Machines are added to the configuration tree automatically in runtime, when the Console receives the timer data.

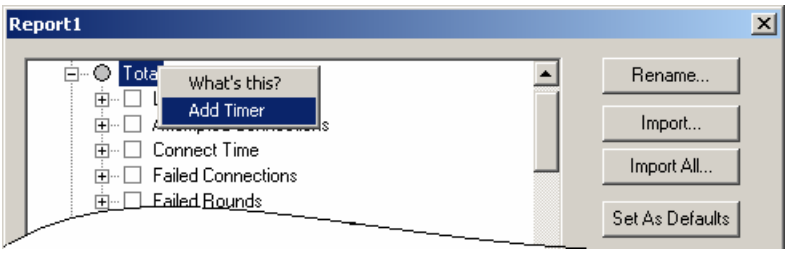
You can define user-defined timers in the Agenda to measure specific activity. For more information on Timers see the TestView *Programmer's Guide.*

Adding a User-Defined Timer

To add a user-defined timer:

1. Right-click on the upper level item in the configuration tree to which you would like to add a timer.

   A pop-up menu displays.



2. Click Add Timer.
3. Enter the name of the timer in the text box.
4. Select the timer attributes you would like to include in the report.
5. Click OK.

   The new timer is included in the Integrated Report.

Importing Report Parameters from a Saved Load Template

WebLOAD enables you to import a complete report configuration or to import a report configuration for a selected component in the Load Session.

Importing a Complete Report Configuration

To import a complete report configuration:

1. In the Reports dialog box, click Import All.

   The Open dialog box appears.
2. Select the Load Template to import.
3. Click Open.

   The Import dialog box displays a list of reports contained in the selected Load Template.

4. Select the report that contains the desired configuration.
5. Click OK.

   The report is configured to display the same measurements.

   Note: If the imported configuration contains a measurement not configured in the current report, WebLOAD ignores the measurement in the Load Template file. It is not added to the report.

This function imports the Integrated Report test configuration from the selected Load Template file. None of the other information in the Load Template file is imported.

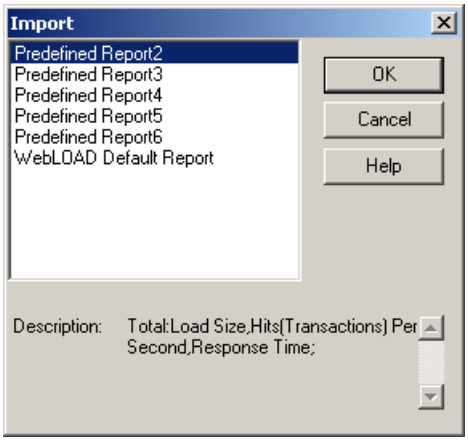
Importing a Report Configuration for a Selected Component

To import a report configuration for a selected component:

1. Select the upper level component in the configuration tree for which you want to import a report configuration.
2. Click Import.

   The Open dialog box appears.
3. Select the Load Template to import and click Open.

   The Import dialog box displays a list of reports contained in the selected Load Template.



4. Select the report that contains the desired configuration.

   WebLOAD displays the list of components (Load Machines and Probing Clients) included in that report.
5. Select the component that contains the desired configuration.

   The current component is configured to display the same measurements.

Note: If the imported component contains a measurement that is not configured in the current report, WebLOAD ignores the measurement in the Load Template file. It does not add it to the report.

This function imports the Integrated Report test configuration from the selected Load Template file. None of the other information in the Load Template file is imported.

Regression Testing

Regression testing enables you to compare Load Sessions while tests are running, or after the test session is completed. WebLOAD regression testing reports enable you to visualize, verify and compare results over time to confirm that results only vary in ways that you expect them to.

Regression testing enables you to configure a report that compares:

- Existing performance data from a previously saved or "expected" Load Session with data from the currently running test session.
- Existing performance data from two previously saved Load Sessions.

Combining performance results from several different Load Sessions into a single report enables you to view, verify and compare results over time and ensure that results only vary in ways that you expect them to.

Configuring an Integrated Report to Include Data from Previously Saved Load Sessions

To configure an Integrated Report t include data from previously saved Load Sessions:

1. Select Reports | Report Manager,

   -Or-

   Click the Open Reports Manager toolbar button.

   The Integrated Report dialog box appears.
2. Click New.

   The Reports dialog box appears.
3. In the Reports dialog box, click Add Session.

   The Open dialog box appears.
4. Select the Load Session (*.LS) file to add and click Open.

   The Load Session is added to the Report dialog box.
5. Configure the Integrated Report to include the measurements you want to view, see Configuring an Integrated Report (on page 300). If a measurement is already selected in the

current Load Session it is automatically selected in the Load Session selected to be included in the report.

6. Click OK.

   The report displays in the Console Results window.

Removing a Previously Saved Load Sessions from the Integrated Report Configuration

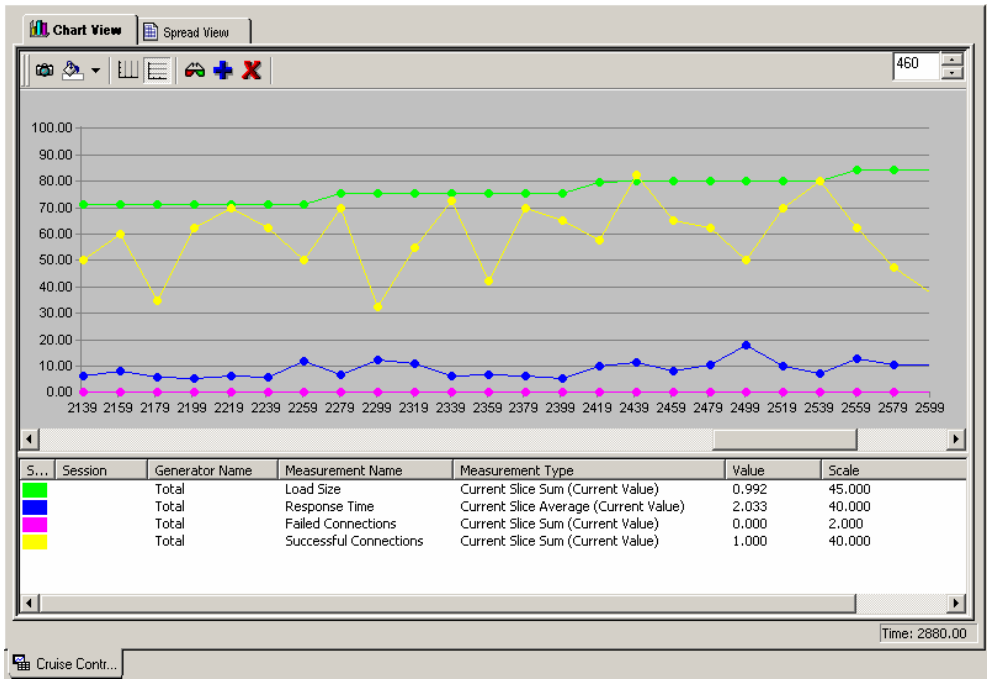
To remove a previously saved Load Sessions from the Integrated Report configuration:

1. Select the Load Session you want to remove from the Integrated Report configuration in the Reports dialog box.
2. Click Remove Session.

   The Load Session is removed from the Report dialog box, and any measurements displayed for that Load Session in the Console Results window are removed.

Integrated Report Chart View

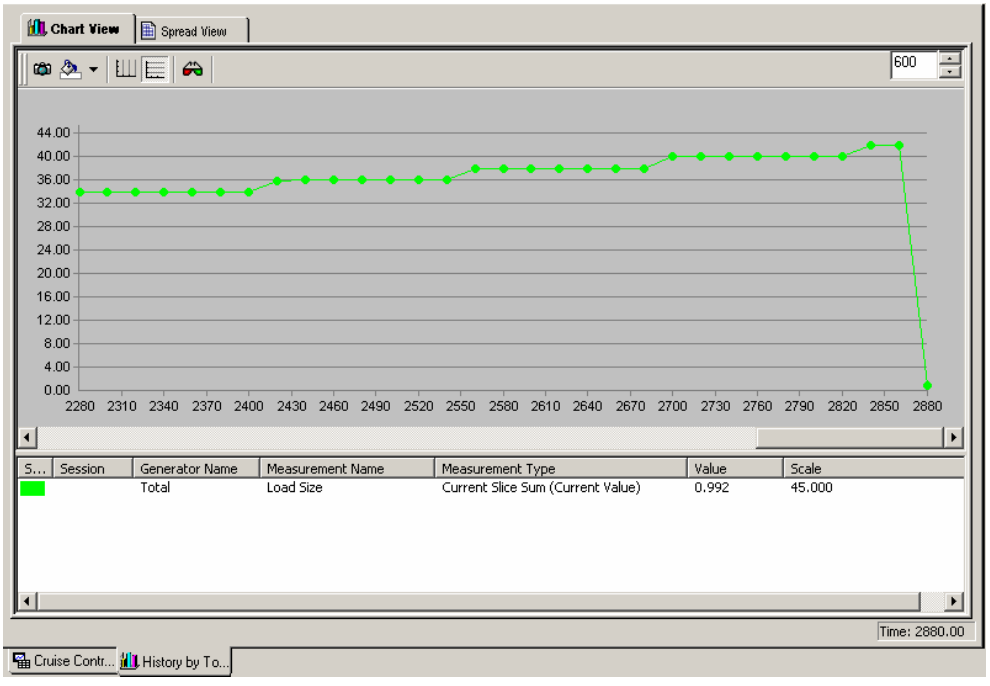
In Chart View, WebLOAD displays the results of your test in a line graph. Separate lines track each measurement selected in the report configuration.



The report above shows an Integrated Report that includes four measurements. Each is tracked on a different line graph. The line graphs are color-coded to match the measurements on the bottom half of the report.

Features of this graph:

- The upper part of the report displays a line graph for each measurement tracked in the report. The X-Axis represents the elapsed time (in seconds) for this test session. The Y-Axis represents the actual value for each measurement, displayed as a percentage relative to the defined maximum Scale Value.
- Right-click on a colored line graph to view the exact value for the selected point.
- The scroll bar at the center of the screen provides you with a quick and easy way to access points in the test session currently not in focus.
- The lower part of the report displays a list of the measurements being tracked for each host and each session, the current value of the measurement and, in the last column on the right, the maximum Scale Value set for each measurement.
- The Time Range box in the top right-hand corner displays the time scale for the X-Axis. In this graph, the X-Axis displays a window of 200 seconds out of the total time span of the Load Session. Using the arrows on the Time Range box, you can change the number of seconds to achieve the report display you desire.
- If you are tracking more than one measurement, the top graph shows multiple graph lines. To bring up a history focusing on a specific measurement, double click on a single measurement line in the Integrated Report.



See Integrated Report Toolbar in Chart View (on page 309) for information on the toolbar at the top of the Integrated Report screen in Chart View.

Changing the Scale of a Report for a Selected Measurement

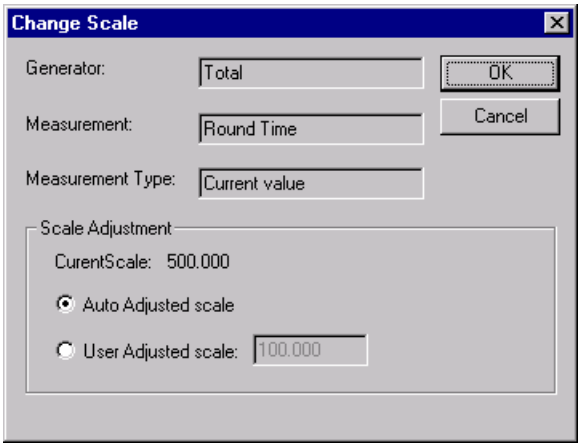
To change the scale of a report for a selected measurement:

1. Right click on a specific measurement and select Change Scale from the pop-up menu,

   -Or-

   Double click on a specific measurement.

   The Change Scale dialog box appears.



2. Select User Adjusted Scale.
3. Enter the new Scale Value.
4. Click OK.

   The scale of the report for that measurement changes in real time. The Scale Value for a measurement can also be changed after the test has run.

Viewing the True Value vs. Relative Value of a Measurement

Each measurement displayed in the Integrated Reports has its own expression or value of measurement. For example, Timer results are measured in seconds. CPU is measured in terms of percentage of usage. Throughput is measured in kilobytes per second. All measurements display as a percentage relative to the defined scale.

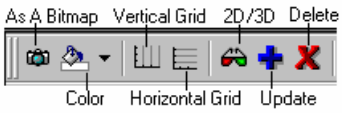
To change the Scale Value of a measurement, see Changing the Scale of a Report for a Selected Measurement (on page 308). When you change the max value, WebLOAD displays the relative

value of that measurement rather than the true value of a measurement. View the real value in the display at the bottom of the screen.

Integrated Report Toolbar in Chart View

At the top of the Integrated Report in Chart View, find the Integrated Report toolbar. This toolbar offers a variety of options for customizing the Integrated Report display, tailoring the colors, perspective, and dimension as needed.

The following illustration shows the Integrated Report toolbar in Chart View and the function of each toolbar button.



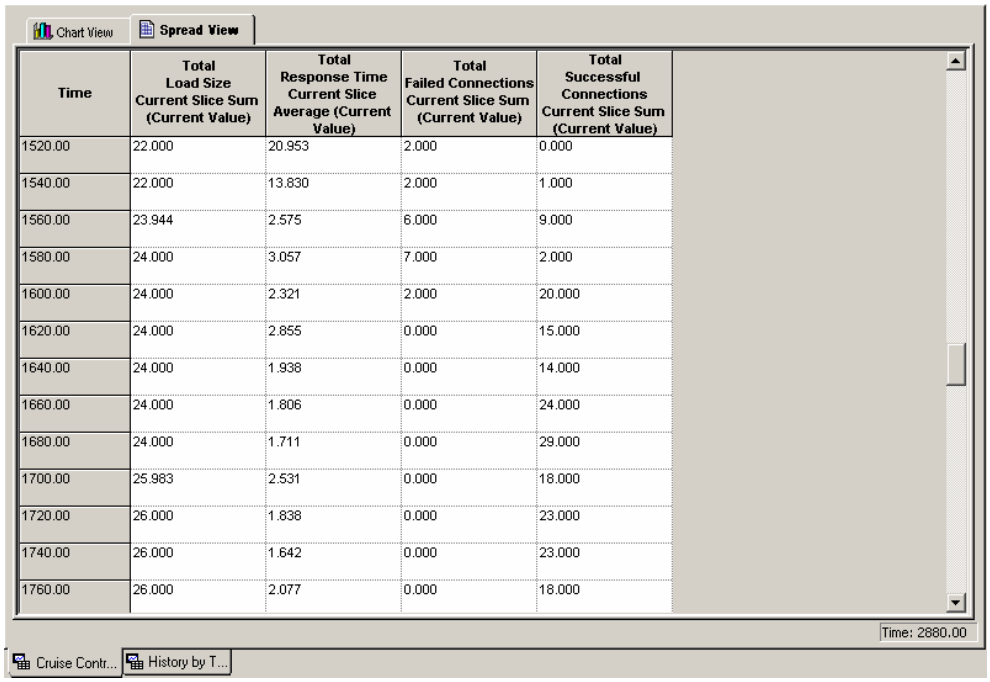
Icon	Toolbar Button	Function
	As A Bitmap	Captures a screen shot of the currently displayed chart to the paste buffer.
	Change Color	Changes the color of an item. ◆ Click on the arrow to display a drop-down palette bar. ◆ Select one of the colors from the palette bar. ◆ Click and drag on the new color from the Color Selection icon to the item to be re-colored.
	Vertical Grid	Toggles the Vertical Grid on and off.
	Horizontal Grid	Toggles the Horizontal Grid on and off.
	2D/3D Toggle	Toggles between 2D and 3D view.
	Update	Updates items displayed in the Integrated Report while the test is in progress. Click the Update button to bring up the Report Configuration dialog box. Using this option, you can select the measurement to be displayed in the report.
	Delete	Deletes items from the Integrated Report while the test is in progress. Select the item you want to delete by clicking on the Series indicator in the list appearing at the bottom and then click on the Delete button. The item is immediately deleted from the report.

Integrated Report in Spread(sheet) View

The Spread View displays the data accumulated in the test session and presented in the Integrated Report Chart View in tabular format.

Note: The data displayed in Spread View tabular format is the same data displayed graphically in the Integrated Report Chart View.

The data displays as follows:



Time	Total Load Size Current Slice Sum (Current Value)	Total Response Time Current Slice Average (Current Value)	Total Failed Connections Current Slice Sum (Current Value)	Total Successful Connections Current Slice Sum (Current Value)
1520.00	22.000	20.953	2.000	0.000
1540.00	22.000	13.830	2.000	1.000
1560.00	23.944	2.575	6.000	9.000
1580.00	24.000	3.057	7.000	2.000
1600.00	24.000	2.321	2.000	20.000
1620.00	24.000	2.855	0.000	15.000
1640.00	24.000	1.938	0.000	14.000
1660.00	24.000	1.806	0.000	24.000
1680.00	24.000	1.711	0.000	29.000
1700.00	25.983	2.531	0.000	18.000
1720.00	26.000	1.838	0.000	23.000
1740.00	26.000	1.642	0.000	23.000
1760.00	26.000	2.077	0.000	18.000

Features of this report:

- The names of all of the measurements and their attributes tracked in the Integrated Report display in the header at the top of the table.
- The first column at the far left lists the times the measurement Attributes were recorded from the start of the current Load Session.
- The data accumulated during the Load Session displays in tabular format in the table fields.

Example:

The last column of this table displays all Total Successful Connections Current Slice Sum (Current Value) values recorded in the current test session. The report resolution is set to 20 seconds. Therefore values are recorded starting at 20 seconds after the test session began and gathered every 20 seconds after that. This is called a reporting interval.

In the first entry in the illustration above:

Time = 1520 - the number of seconds from the start of the test.

Total Successful Connections Current Slice Sum (Current Value) = 0.000 - the sum for this measurement since the last Statistics reporting (over the last reporting interval).

In the second entry:

Time = 1540 - the number of seconds from the start of the test.

Total Successful Connections Current Slice Sum (Current Value) = 1.000 - the sum for this measurement over the second reporting interval.

C H A P T E R 2 2

WebLOAD Statistics Reports

The Statistics Report is both a real-time monitor that displays the results of your test while you run it and a report that you can save and export.

WebLOAD collects approximately 30 different statistics during a test. The Statistics Report displays the values for all of them. You can also create Integrated Reports that display only the statistics that you choose. For information about Integrated Reports, see WebLOAD Performance Reports (on page 293).

In This Chapter

Statistical Measurements and Their Values 313
Summary Reports 323

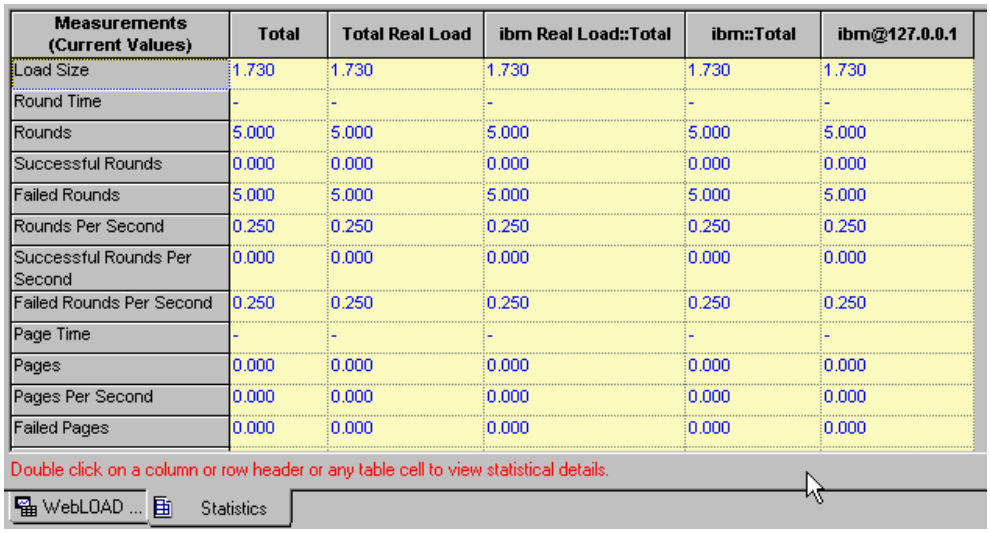
Statistical Measurements and Their Values

The Statistics Report is presented as a table.

Row headings (the labels in the first column on the left) are the names of the statistical measurements.

Column headings categorize the statistics as follows:

- Total-sums the values for each statistic across all the Agendas running during the test.
- AgendaName::Total-sums the values for each statistic across all instances of the specified Agenda running during the test.
- AgendaName@LoadGenerator-sums the values for each statistic across all instances of the specified Agenda running under the specified Load Generator during the test.



Measurements (Current Values)	Total	Total Real Load	ibm Real Load::Total	ibm::Total	ibm@127.0.0.1
Load Size	1.730	1.730	1.730	1.730	1.730
Round Time	-	-	-	-	-
Rounds	5.000	5.000	5.000	5.000	5.000
Successful Rounds	0.000	0.000	0.000	0.000	0.000
Failed Rounds	5.000	5.000	5.000	5.000	5.000
Rounds Per Second	0.250	0.250	0.250	0.250	0.250
Successful Rounds Per Second	0.000	0.000	0.000	0.000	0.000
Failed Rounds Per Second	0.250	0.250	0.250	0.250	0.250
Page Time	-	-	-	-	-
Pages	0.000	0.000	0.000	0.000	0.000
Pages Per Second	0.000	0.000	0.000	0.000	0.000
Failed Pages	0.000	0.000	0.000	0.000	0.000

The statistics displayed in each row are grouped into the following categories:

- Per time unit statistics are ratios that calculate an average value for an action or process. For example: Hits Per Second, Rounds Per Second.
- Timers track the amount of time it took the Virtual Clients to complete an action or process. For example: Connect Time, Round Time.
- Counters track the number of times the Virtual Clients completed an action or process. For example: Hits, Rounds.

By default, WebLOAD displays values for its statistics at 20-second intervals. The reported value at each reporting interval is called the current value.

Changing the Length of the Reporting Interval

To change the length of the reporting interval:

1. Select Tools | Global Options | General tab.
2. Change the value in the Statistics reporting resolution field to any number of seconds between 10 and 3,600.

Current Value Computation

WebLOAD calculates the current value as follows:

- For per time unit and timer statistics, WebLOAD averages the values collected during the last reporting interval (current slice). If the reporting interval is 20 seconds (the default), the value for Hits Per Second is the average number of hits per second over the last 20 seconds. The value for Hit Time is the average time it took for a successful hit to be completed during the last 20 seconds.
- For counters, WebLOAD reports the number of times the item being counted occurred over the last reporting interval. If the reporting interval is 20 seconds, the value for Hits is the actual number of hits that occurred during the last 20 seconds.

Per Time-Unit Statistics Definitions

The following table presents definitions of the per time-unit statistics:

Statistic	Definition
Hits Per Second	A hit is a request for information made by a Virtual Client to the application being tested (the ABT). In other words, an HTTP request. For example, a Get statement for a URL retrieves a page. The page can include any number of graphics and contents files. Each request for each gif, jpeg, html file, etc., is a single hit. Hits Per Second is the number of times the Virtual Clients made an HTTP request divided by the elapsed time, in seconds. The value posted in the Current Value column is the number (sum) of HTTP requests per second during the last reporting interval.
Successful Hits Per Second	The number of times the Virtual Clients obtained the correct HTTP response to their HTTP requests divided by the elapsed time, in seconds. The value posted in the Current Value column is the number (sum) of successful HTTP requests per second during the last reporting interval.
Failed Hits Per Second	The number of times the Virtual Clients did not obtain the correct HTTP response divided by the elapsed time, in seconds. The value posted in the Current Value column is the number (sum) of unsuccessful HTTP requests per second during the last reporting interval.

Statistic	Definition
Pages Per Second	A page is an upper level request for information made by a Virtual Client to the application being tested (the ABT). For example, a Get statement for a URL retrieves a page. Pages Per Second is the number of times the Virtual Clients made upper level requests divided by the elapsed time, in seconds. The value posted in the Current Value column is the number (sum) of requests per second during the last reporting interval.
Successful Pages Per Second	The number of times the Virtual Clients obtained the correct HTTP response to their upper level requests divided by the elapsed time, in seconds. The value posted in the Current Value column is the number (sum) of successful requests per second during the last reporting interval.
Failed Pages Per Second	The number of times the Virtual Clients did not obtain the correct response to an upper level request divided by the elapsed time, in seconds. The value posted in the Current Value column is the number (sum) of unsuccessful requests per second during the last reporting interval.
Rounds Per Second	The number of times the Virtual Clients attempted to run the Agenda divided by the elapsed time, in seconds. The value posted in the Current Value column is the number (sum) of attempts (both successful and unsuccessful) per second during the last reporting interval.
Successful Rounds Per Second	The number of times the Virtual Clients completed an entire iteration of the Agenda divided by the elapsed time, in seconds. The value posted in the Current Value column is the number (sum) of successful iterations of the Agenda per second during the last reporting interval.
Failed Rounds Per Second	The number of times the Virtual Clients started but did not complete an iteration of this Agenda divided by the elapsed time, in seconds. The value posted in the Current Value column is the number (sum) of failed iterations of the Agenda per second during the last reporting interval.

Statistic	Definition
Throughput (bytes per second)	The average number of bytes per second transmitted from the ABT to the Virtual Clients running this Agenda during the last reporting interval. In other words, this is the amount of the Response Data Size (sum) divided by the number of seconds in the reporting interval.
Connection Speed (Bits Per Second)	The number of bits transmitted back and forth between the Virtual Clients and the ABT divided by the time it took to transmit those bits, in seconds. You can set the Virtual Clients to emulate a particular connection speed during the test either by using the Variable Connection Speed settings (Tools | Default Options | Connection tab) or by coding the connection speed in the Agenda. If a connection speed is specified for the test, WebLOAD reports it in the Statistics Report. The value posted in the Current Value column is the number (sum) of bits passed per second during the last reporting interval. It should match, very closely, the connection speed you specified for the test.

Timer Definitions

The following table presents definitions of the timer statistics:

Statistic	Definition
Round Time	The time it takes one Virtual Client to finish one complete iteration of an Agenda, in seconds. The value posted in the Current Value column is the average time it took the Virtual Clients to finish one complete iteration of the Agenda during the last reporting interval.
Hit Time	The time it takes to complete a successful HTTP request, in seconds. (Each request for each gif, jpeg, html file, etc. is a single hit.) The time of a hit is the sum of the Connect Time, Send Time, Response Time, and Process Time. The value posted in the Current Value column is the average time it took to make an HTTP request and process its response during the last reporting interval.

Statistic	Definition
Connect Time	The time it takes for a Virtual Client to connect to the application being tested (the ABT), in seconds. In other words, the time it takes from the beginning of the HTTP request to the TCP/IP connection. The value posted in the Current Value column is the average time it took a Virtual Client to connect to the ABT during the last reporting interval. If the Persistent Connection option is enabled (Tools | Default Options | Browser Parameters tab), there may not be a value for Connect Time because the HTTP connection remains open between successive HTTP requests.
Send Time	The time it takes the Virtual Clients to write an HTTP request to the ABT, in seconds. The value posted in the Current Value column is the average time it took the Virtual Clients to write a request to the ABT during the last reporting interval.
Response Time	The time it takes the ABT to send the object of an HTTP request back to a Virtual Client, in seconds. In other words, the time from the end of the HTTP request until the Virtual Client has received the complete item it requested (Time to first byte + Receive Time - Process Time). The value posted in the Current Value column is the average time it took the ABT to respond to an HTTP request during the last reporting interval.
Process Time	The time it takes WebLOAD to parse an HTTP response from the ABT and then populate the document-object model (the DOM), in seconds. The value posted in the Current Value column is the average time it took WebLOAD to parse an HTTP response during the last reporting interval.
Automatic Data Collection timers	If you have Automatic Data Collection enabled (Tools | Default Options | Functional Testing tab, Automatic Data Collection) WebLOAD creates a timer for each GET and POST statement in the Agenda. The row heading is the URL of the page. The value posted is the average time it took a Virtual Client to complete the GET or POST statement, in seconds, during the last reporting interval.
Time to First Byte	The time it takes from when a request is sent until the Virtual Client receives the first byte of data.

Statistic	Definition
Receive Time	The elapsed time between receiving the first byte and the last byte.
<User-Defined> timer	You can add timers to Agendas to keep track of the amount of time it takes to complete specific actions. (See the *JavaScript Programming for WebLOAD Test Agendas* guide.) If there are any timers in the Agendas that you are running, WebLOAD reports their values in the Statistics Report. The row heading is the name (argument) of the timer, that is, the string in parenthesis in the SetTimer() function call. The timer represents the time it takes to complete all the actions between the SetTimer() call and its corresponding SendTimer() call, in seconds. The value posted is the average time it took a Virtual Client to complete the actions between the pair of timer calls, in seconds, during the last reporting interval.
<User-Defined Transaction> timers	You can add user-defined transaction functions to Agendas for functional tests. (See the J*avaScript Programming for WebLOAD Test Agendas* guide.) If there is a user-defined transaction function in the Agenda that you are running, WebLOAD reports a timer for it in the Statistics Report. The row heading is the name (argument) of the user-defined transaction, that is, the string in parentheses in the BeginTransaction() function call. The timer represents the average time it took to complete all the actions between the BeginTransaction() call and its corresponding EndTransaction() call, in seconds, during the last reporting interval.
Java and ActiveX timers	You can add function calls to your Agendas that allow you to instantiate and call methods and properties in Java and ActiveX components. (See the J*avaScript Programming for WebLOAD Test Agendas* guide.) If there are ActiveX or Java function calls in the Agenda you are running, WebLOAD reports timers for them in the Statistics Report. The row heading is the name of the function call. The value posted is the average amount of time it took to complete the function call, in seconds, during the last reporting interval.

Statistic	Definition
Server Performance Measurements (For example: Sessions Errored Out, % Processor Time, Current Disk Queue Length, and so on)	If you selected Performance Monitor statistics for the report, WebLOAD creates a row for them and reports their values in the Statistics Report. For definitions of the statistics, see the Server Monitor Definition dialog box (click the Pencil icon so that you are in edit mode, then select Tools | Performance Measurements Manager). Be selective when choosing server performance measurements or the system resources required to manage the data might affect the Console.

Counter Definitions

The following table presents definitions of the counter statistics:

Statistic	Definition
Load Size	The number of Virtual Clients running during the last reporting interval.
Rounds	The total number of times the Virtual Clients attempted to run the Agenda during the last reporting interval.
Successful Rounds	The total number of times the Virtual Clients completed one iteration of the Agenda during the last reporting interval.
Failed Rounds	The total number of times the Virtual Clients started but did not complete the Agenda during the last reporting interval.
Hits	The total number of times the Virtual Clients made an HTTP request to the application being tested (ABT) during the last reporting interval. (Each request for each gif, jpeg, html file, etc. is a single hit.)
Successful Hits	The total number of times the Virtual Clients made an HTTP request and received the correct HTTP response from the ABT during the last reporting interval. (Each request for each gif, jpeg, html file, etc. is a single hit.)
Failed Hits	The total number of times the Virtual Clients made an HTTP request but did not receive the correct HTTP response from the ABT during the last reporting interval. (Each request for each gif, jpeg, html file, etc. is a single hit.)
Attempted Connections	The total number of times the Virtual Clients attempted to connect to the ABT during the last reporting interval.

Statistic	Definition
Successful Connections	The total number of times the Virtual Clients were able to connect to the ABT during the last reporting interval. This number is always less than or equal to the number of successful hits because several hits might use the same HTTP connection if the Persistent Connection option is enabled (Tools | Default Options | Browser Parameters tab).
Failed Connections	The total number of times the Virtual Clients tried to connect to the ABT but was unable to, during the last reporting interval. This number is always less than or equal to the number of failed hits because hits can fail for reasons other than a failed connection.
Responses	The number of times the ABT responded to an HTTP request during the last reporting interval. This number should match the number of successful hits.
Return Status	WebLOAD creates a row in the Statistics Report for each kind of HTTP status code it receives as an HTTP response from the ABT (redirection codes, success codes, server error or client error codes). The value posted is the number of times the Virtual Clients received that status code during the last reporting interval.
Response Data Size	The size, in bytes, of all the HTTP responses sent by the ABT during the last reporting interval. WebLOAD uses this value to calculate Throughput (bytes per second).
Automatic data collection counters	If you have Automatic Data Collection enabled (Tools | Default Options | Functional Testing tab, Automatic Data Collection) WebLOAD creates three counters for each GET and POST statement in the Agenda. The counters are the total number of times it occurred, the number of times it succeeded, and the number of times it failed during the last reporting interval.

Statistic	Definition
User-defined counters	You can add your own counters to Agendas with the SendCounter() and the SendMeasurement() functions. (See the *JavaScript Programming for WebLOAD Test Agendas* guide.) If there is a user-defined counter in the Agenda that you are running, WebLOAD reports their values in the Statistics Report. The row heading is the name (argument) of the counter, that is, the string in parenthesis in the SendCounter() or SendMeasurement() function call. The value reported is the number of times the counter was incremented during the last reporting interval.
<User-Defined Transaction> counters	You can add user-defined transaction functions to Agendas for functional tests. (See the J*avaScript Programming for WebLOAD Test Agendas* guide.) If there is a user-defined transaction function in the Agenda that you are running, WebLOAD reports three counters for it in the Statistics Report. The row heading is the name (argument) of the user-defined transaction, that is, the string in parenthesis in the BeginTransaction() function call. The counters are the total number of times it occurred, the number of times it succeeded, and the number of times it failed during the last reporting interval.
Java and ActiveX counters	You can add function calls to your Agendas that allow you to instantiate and call methods and properties in Java and ActiveX components. (See the J*avaScript Programming for WebLOAD Test Agendas* guide.) If there are ActiveX or Java function calls in the Agenda that you are running, WebLOAD reports three counters for them in the Statistics Report. The row heading is the name of the function call. The counters are the total number of times it occurred, the number of times it succeeded, and the number of times it failed during the last reporting interval.

Statistics for Virtual Clients and Probing Clients

There are differences in the statistics reported for Virtual Clients and Probing Clients:

- The values reported for Virtual Clients are average values based on the data reported by the total number of Virtual Clients running that Agenda.
- The values reported for a Probing Client are the specific values for that single, representative user.

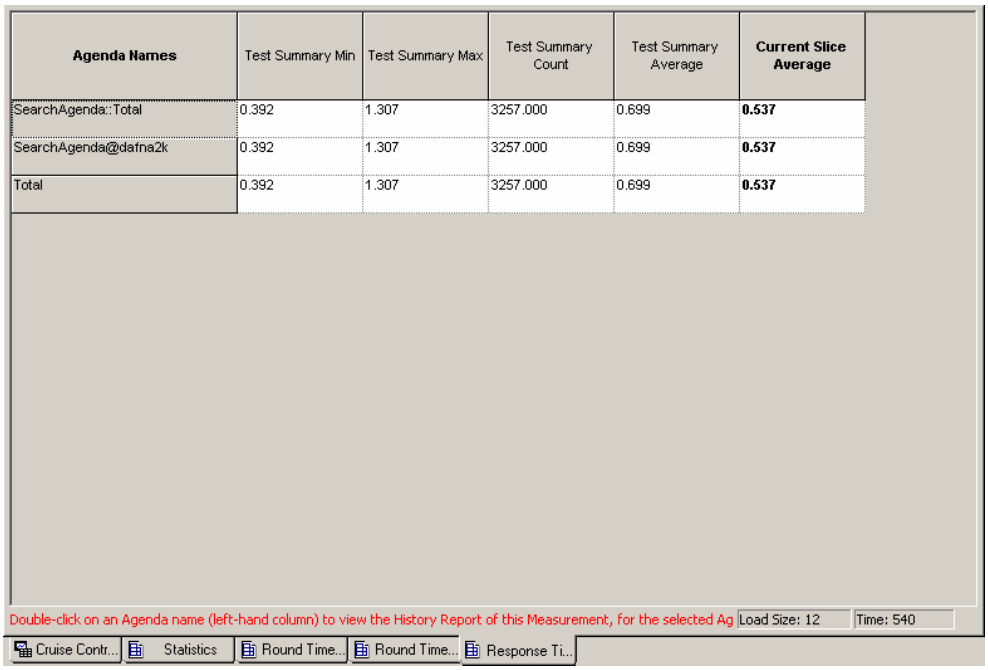
By default, Virtual Clients are single-threaded so the load machines can support more Virtual Clients overall. Probing Clients run with four threads in order to emulate a single user as closely as possible.

Therefore, you should use the statistics that you obtain from the different types of emulated users in different ways. For example, if you are interested in the exact experience of a typical user of your site, examine the timers like Round Time, Hit Time, Connect Time, and so on from a Probing Client. For performance or load issues, however, examine statistics like Successful and Failed hits, Successful and Failed Rounds, Rounds Per Second, Hits Per Second, and so on from Virtual Clients.

Summary Reports

Each cell in the Statistics Report is an access point for another report:

- Double-click on a row heading to display a Summary by Measurement Report.
- Double-click on a column heading to display a Summary by Agenda or Summary by Total Report.
- Double-click on a cell to display a summary report for that statistical measurement.



Summary reports provide the following values:

Value	Definition
Current Slice/Current Value	The value posted for this reporting interval - current slice.
Min	The lowest value reported for this statistic since the beginning of the test.
Max	The highest value reported for this statistic since the beginning of the test.
Count	(For timers only.) The total number of readings (the number of times the item being timed occurred) for the timed statistic since the beginning of the test. For example, for Hit Time, Count shows the number of hits completed.
Average	For per time unit statistics and counters, average is the total of all of the current values divided by the number of reporting intervals. For timers, average is the total amount of time counted by the timer (not the elapsed time) divided by the Count (the total number of readings). For example, the average for Hit Time is the amount of time it took to complete all the successful hits divided by the number of successful hits (the Count).
Sum	The aggregate or total value for this statistic in this Agenda since the beginning of the test.
STDev	The average amount this statistic varies from the average number in this Agenda.

The Statistics Report displays all of the statistics that WebLOAD collects unless there were no readings for a statistic during a test. For example, if there were no unsuccessful rounds during a test, the table would not display a row for Failed Rounds.

Additionally, if a measurement has no meaning for a specific statistic, WebLOAD displays a dash (-) in the cell. For example, if there were two Agendas running but only one of them returned an HTTP Response Status code, the table would include a row for that Response Status and would display a dash in the cell for the Agenda that did not return the code.

Using the Statistics Report

Open specific statistical values by double-clicking on a cell. You can click on any Agenda Name or on Total in the column headers, any Measurement Name in the left-hand column, or any cell at the intersection of any measurement row and Agenda or Total column.

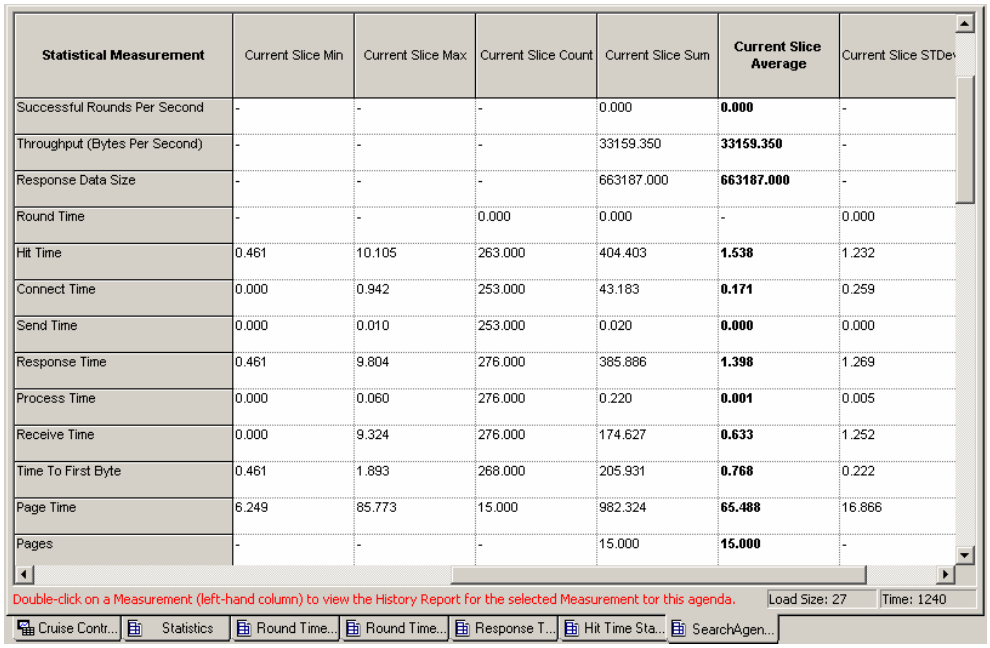
Double click on	To
Agenda Name	Displays a Summary Report table containing all measurement attribute values for the selected Agenda.

Double click on	To
Total	Displays a Summary Report table containing the sum of all measurement attribute values for all current Agendas.
Measurement Name	Displays a Summary Report table containing the attribute values for this measurement for all current Agendas
Intersection, Measurement & Agenda	Displays a History Report containing all of the Attribute values recorded from the start of the current test session. These values are specific to the selected measurement and Agenda. The complete History Report can also be accessed via the Summary Reports.
Intersection, Measurement & Total	Displays a History Report containing all of the Attribute values recorded for the selected Measurement totaled for all current Agendas. The complete History Report can also be accessed via the Summary Reports.

Statistics Report-Summary by Agenda

Double-click on a specific Agenda name in the Statistics Report main window to call up a summary table. When you select an Agenda, the report contains the values for all the attributes collected for that Agenda.

The following example contains all of the values collected for the Agenda, SearchAgenda:



Statistical Measurement	Current Slice Min	Current Slice Max	Current Slice Count	Current Slice Sum	Current Slice Average	Current Slice STDev
Successful Rounds Per Second	-	-	-	0.000	**0.000**	-
Throughput (Bytes Per Second)	-	-	-	33159.350	**33159.350**	-
Response Data Size	-	-	-	663187.000	**663187.000**	-
Round Time	-	-	0.000	0.000	-	0.000
Hit Time	0.461	10.105	263.000	404.403	**1.538**	1.232
Connect Time	0.000	0.942	253.000	43.183	**0.171**	0.259
Send Time	0.000	0.010	253.000	0.020	**0.000**	0.000
Response Time	0.461	9.804	276.000	385.886	**1.398**	1.269
Process Time	0.000	0.060	276.000	0.220	**0.001**	0.005
Receive Time	0.000	9.324	276.000	174.627	**0.633**	1.252
Time To First Byte	0.461	1.893	268.000	205.931	**0.768**	0.222
Page Time	6.249	85.773	15.000	982.324	**65.488**	16.866
Pages	-	-	-	15.000	**15.000**	-

The names of all the measurements tracked during the current Load Session are listed in the column at the far left. The names of all the Attributes totaled for the relevant measurements are displayed in the header at the top of the table.

Example:

The screen above displays the Response Time measurement values for the Agenda, SearchAgenda.

Note: The MINIMUM time needed for the ABT to respond to requests from the Agenda during the current test session is 1.246 seconds. The MAXIMUM time needed for the ABT to respond to the Agenda run by Generator1 during the current slice is 0.461 seconds. The AVERAGE Response Time to the Agenda in the current slice is 1.398 seconds.

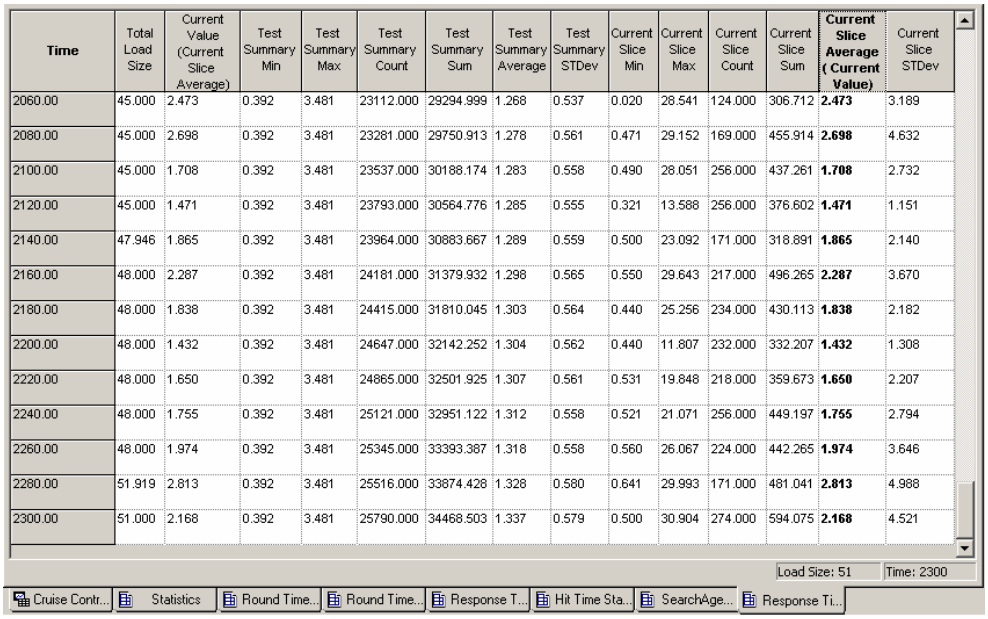
Getting More Detailed Information

Double-click on a measurement name in the left-hand column to call up a History Report for this measurement in the Agenda. This History Report can also be accessed from the Statistics Report main window by double clicking on the button located at the intersection of the selected measurement and Agenda. This History Report is described in the section Statistics Report-History by Measurement and Agenda (on page 331).

Example:

Double-click on Response Time in the left-hand column to call up a History Report. This History Report lists all of the Response Time Attribute values recorded for the Agenda run by Load Generator1 from the start of the current test session. This History Report can also be

accessed from the Statistics Report main window by double-clicking on the button located at the intersection of the Response Time row and SearchAgenda column.

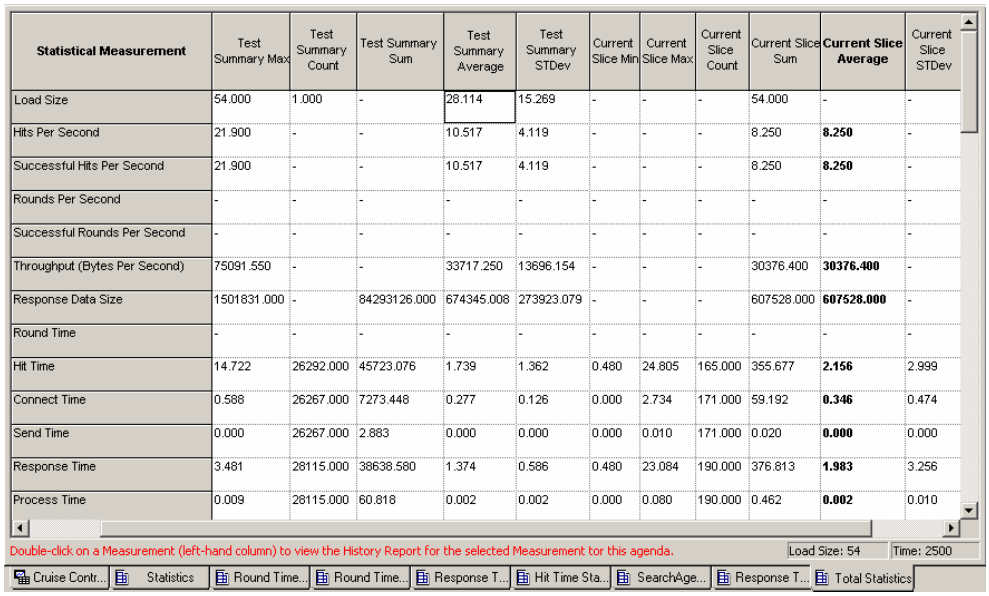


Time	Total Load Size	Current Value (Current Slice Average)	Test Summary Min	Test Summary Max	Test Summary Count	Test Summary Sum	Test Summary Average	Test Summary STDev	Current Slice Min	Current Slice Max	Current Slice Count	Current Slice Sum	**Current Slice Average (Current Value)**	Current Slice STDev
2060.00	45.000	2.473	0.392	3.481	23112.000	29294.999	1.268	0.537	0.020	28.541	124.000	306.712	**2.473**	3.189
2080.00	45.000	2.698	0.392	3.481	23281.000	29750.913	1.278	0.561	0.471	29.152	169.000	455.914	**2.698**	4.632
2100.00	45.000	1.708	0.392	3.481	23537.000	30188.174	1.283	0.558	0.490	28.051	256.000	437.261	**1.708**	2.732
2120.00	45.000	1.471	0.392	3.481	23793.000	30564.776	1.285	0.555	0.321	13.588	256.000	376.602	**1.471**	1.151
2140.00	47.946	1.865	0.392	3.481	23964.000	30883.667	1.289	0.559	0.500	23.092	171.000	318.891	**1.865**	2.140
2160.00	48.000	2.287	0.392	3.481	24181.000	31379.932	1.298	0.565	0.550	29.643	217.000	496.265	**2.287**	3.670
2180.00	48.000	1.838	0.392	3.481	24415.000	31810.045	1.303	0.564	0.440	25.256	234.000	430.113	**1.838**	2.182
2200.00	48.000	1.432	0.392	3.481	24647.000	32142.252	1.304	0.562	0.440	11.807	232.000	332.207	**1.432**	1.308
2220.00	48.000	1.650	0.392	3.481	24865.000	32501.925	1.307	0.561	0.531	19.848	218.000	359.673	**1.650**	2.207
2240.00	48.000	1.755	0.392	3.481	25121.000	32951.122	1.312	0.558	0.521	21.071	256.000	449.197	**1.755**	2.794
2260.00	48.000	1.974	0.392	3.481	25345.000	33393.387	1.318	0.558	0.560	26.067	224.000	442.265	**1.974**	3.646
2280.00	51.919	2.813	0.392	3.481	25516.000	33874.428	1.328	0.580	0.641	29.993	171.000	481.041	**2.813**	4.988
2300.00	51.000	2.168	0.392	3.481	25790.000	34468.503	1.337	0.579	0.500	30.904	274.000	594.075	**2.168**	4.521

Statistics Report-Summary by Total

Double-click on the heading Total at the top of the Total column in the Statistics Report main window to call up the Summary by Total Report.

This table displays the sum of all measurement values for all current Agendas.



Statistical Measurement	Test Summary Max	Test Summary Count	Test Summary Sum	Test Summary Average	Test Summary STDev	Current Slice Min	Current Slice Max	Current Slice Count	Current Slice Sum	**Current Slice Average**	Current Slice STDev
Load Size	54.000	1.000	-	28.114	15.269	-	-	-	54.000	-	-
Hits Per Second	21.900	-	-	10.517	4.119	-	-	-	8.250	**8.250**	-
Successful Hits Per Second	21.900	-	-	10.517	4.119	-	-	-	8.250	**8.250**	-
Rounds Per Second	-	-	-	-	-	-	-	-	-	-	-
Successful Rounds Per Second	-	-	-	-	-	-	-	-	-	-	-
Throughput (Bytes Per Second)	75091.550	-	-	33717.250	13696.154	-	-	-	30376.400	**30376.400**	-
Response Data Size	1501831.000	-	84293126.000	674345.008	273923.079	-	-	-	607528.000	**607528.000**	-
Round Time	-	-	-	-	-	-	-	-	-	-	-
Hit Time	14.722	26292.000	45723.076	1.739	1.362	0.480	24.805	165.000	355.677	**2.156**	2.999
Connect Time	0.588	26267.000	7273.448	0.277	0.126	0.000	2.734	171.000	59.192	**0.346**	0.474
Send Time	0.000	26267.000	2.883	0.000	0.000	0.000	0.010	171.000	0.020	**0.000**	0.000
Response Time	3.481	28115.000	38638.580	1.374	0.586	0.480	23.084	190.000	376.813	**1.983**	3.256
Process Time	0.009	28115.000	60.818	0.002	0.002	0.000	0.080	190.000	0.462	**0.002**	0.010

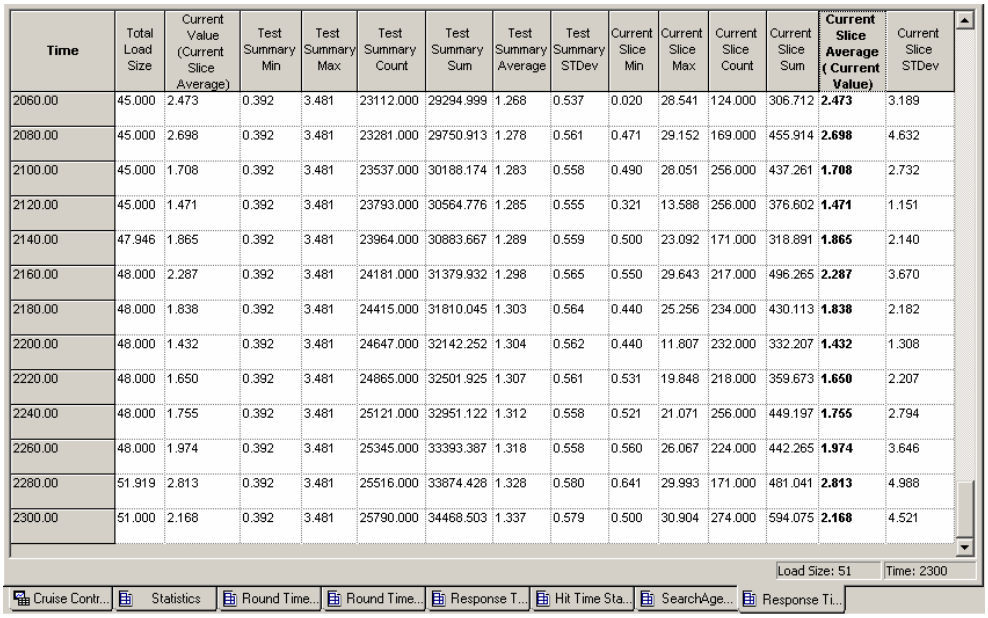
The names of all the measurements tracked during the current Load Session are listed in the column at the far left. The names of all the attributes totaled for the relevant measurements for all current Agendas are displayed in the header at the top of the table.

Getting More Detailed Information

Double-click on a measurement name in the left-hand column to call up a History Report. This History Report can also be accessed from the Statistics Report main window by double clicking on the button located at the Intersection of the selected Measurement row and Total column. This History Report is described in the Statistics Report-History by Measurement and Total (on page 332).

Example:

Double-click on Response Time in the left-hand column to call up a History Report. This History Report lists all of the Response Time Attribute values recorded for all Agendas from the start of the current test session. This is the same Summary Report accessed from the Statistics Report main window by double-clicking on the button located at the intersection of the Response Time row and Total column.

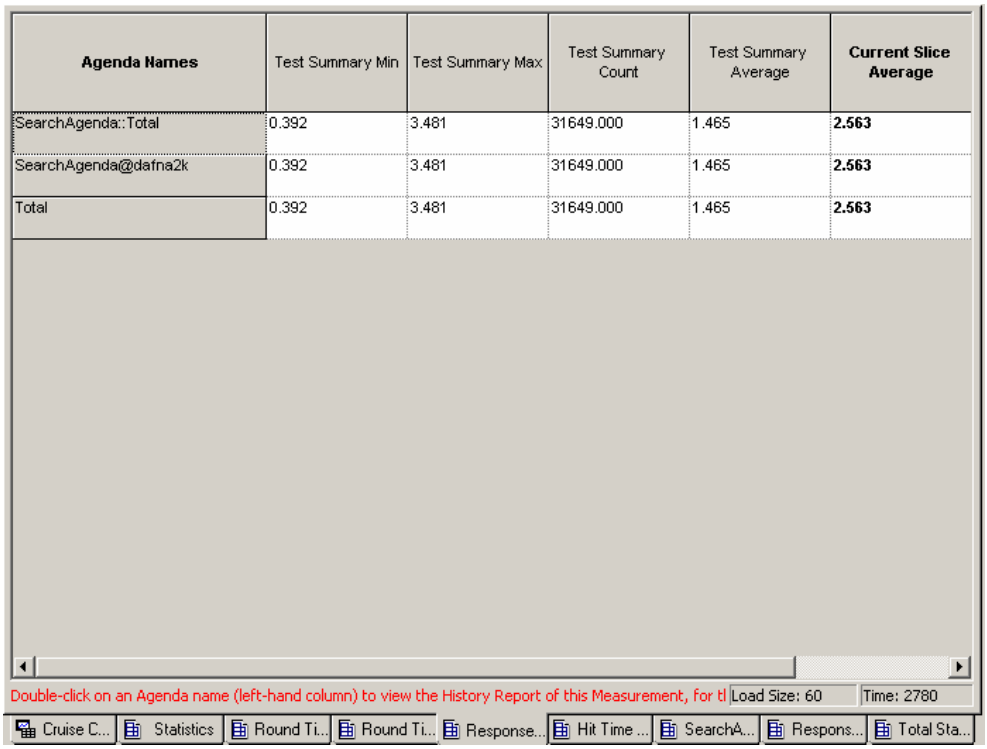


Time	Total Load Size	Current Value (Current Slice Average)	Test Summary Min	Test Summary Max	Test Summary Count	Test Summary Sum	Test Summary Average	Test Summary STDev	Current Slice Min	Current Slice Max	Current Slice Count	Current Slice Sum	**Current Slice Average (Current Value)**	Current Slice STDev
2060.00	45.000	2.473	0.392	3.481	23112.000	29294.999	1.268	0.537	0.020	28.541	124.000	306.712	**2.473**	3.189
2080.00	45.000	2.698	0.392	3.481	23281.000	29750.913	1.278	0.561	0.471	29.152	169.000	455.914	**2.698**	4.632
2100.00	45.000	1.708	0.392	3.481	23537.000	30188.174	1.283	0.558	0.490	28.051	256.000	437.261	**1.708**	2.732
2120.00	45.000	1.471	0.392	3.481	23793.000	30564.776	1.285	0.555	0.321	13.588	256.000	376.602	**1.471**	1.151
2140.00	47.946	1.865	0.392	3.481	23964.000	30883.667	1.289	0.559	0.500	23.092	171.000	318.891	**1.865**	2.140
2160.00	48.000	2.287	0.392	3.481	24181.000	31379.932	1.298	0.565	0.550	29.643	217.000	496.265	**2.287**	3.670
2180.00	48.000	1.838	0.392	3.481	24415.000	31810.045	1.303	0.564	0.440	25.256	234.000	430.113	**1.838**	2.182
2200.00	48.000	1.432	0.392	3.481	24647.000	32142.252	1.304	0.562	0.440	11.807	232.000	332.207	**1.432**	1.308
2220.00	48.000	1.650	0.392	3.481	24865.000	32501.925	1.307	0.561	0.531	19.848	218.000	359.673	**1.650**	2.207
2240.00	48.000	1.755	0.392	3.481	25121.000	32951.122	1.312	0.558	0.521	21.071	256.000	449.197	**1.755**	2.794
2260.00	48.000	1.974	0.392	3.481	25345.000	33393.387	1.318	0.558	0.560	26.067	224.000	442.265	**1.974**	3.646
2280.00	51.919	2.813	0.392	3.481	25516.000	33874.428	1.328	0.580	0.641	29.993	171.000	481.041	**2.813**	4.988
2300.00	51.000	2.168	0.392	3.481	25790.000	34468.503	1.337	0.579	0.500	30.904	274.000	594.075	**2.168**	4.521

Statistics Report-Summary by Measurement

Double-click on a specific measurement name in the Statistics Report main window to call up the following summary table.

This table displays all attribute values for the selected measurement for the current Agendas.



Agenda Names	Test Summary Min	Test Summary Max	Test Summary Count	Test Summary Average	Current Slice Average
SearchAgenda::Total	0.392	3.481	31649.000	1.465	**2.563**
SearchAgenda@dafna2k	0.392	3.481	31649.000	1.465	**2.563**
Total	0.392	3.481	31649.000	1.465	**2.563**

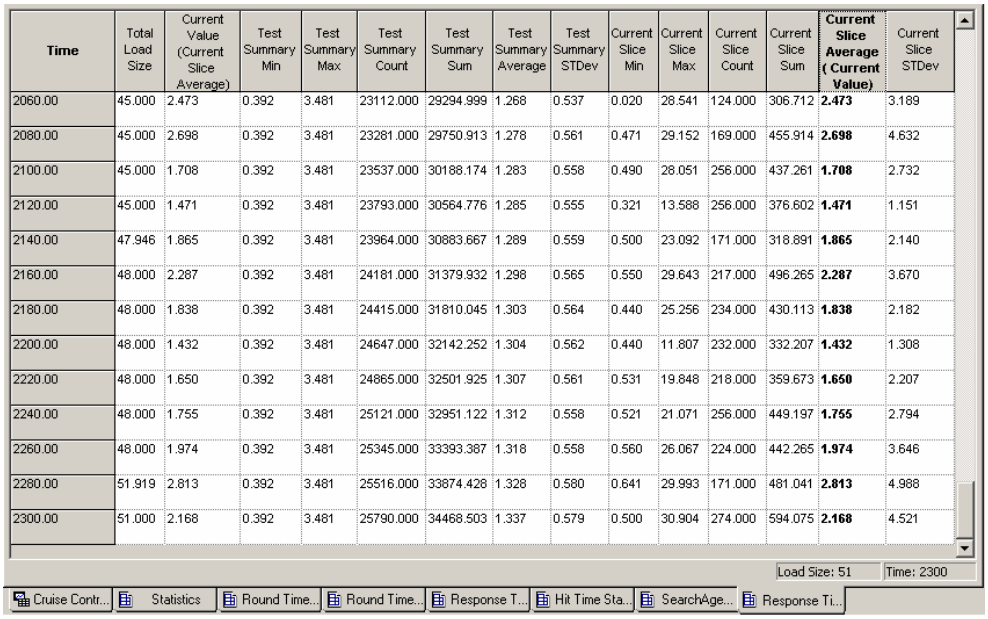
The screen above shows the Response Time values for Agenda, SearchAgenda. The attributes displayed are for Current Slice, Min, Max Count and Average. The third line displays the total values for these attributes for all running Agendas.

Getting More Detailed Information

Double-click on an Agenda name in the left-hand column to call up a History Report. This History Report can also be accessed from the Statistics Report main window by double-clicking on the button located at the Intersection of the selected measurement row and Agenda column. This History Report is described in Statistics Report-History by Measurement and Agenda (on page 331).

Example:

Double-click on SearchAgenda in the left-hand column to call up a History Report. This History Report lists all of the Response Time Attribute values recorded for all Agendas from the start of the current test session. This is the same History Report accessed from the Statistics Report main window by double-clicking on the button located at the intersection of the Response Time row and Agenda column.



Time	Total Load Size	Current Value (Current Slice Average)	Test Summary Min	Test Summary Max	Test Summary Count	Test Summary Sum	Test Summary Average	Test Summary STDev	Current Slice Min	Current Slice Max	Current Slice Count	Current Slice Sum	**Current Slice Average (Current Value)**	Current Slice STDev
2060.00	45.000	2.473	0.392	3.481	23112.000	29294.999	1.268	0.537	0.020	28.541	124.000	306.712	**2.473**	3.189
2080.00	45.000	2.698	0.392	3.481	23281.000	29750.913	1.278	0.561	0.471	29.152	169.000	455.914	**2.698**	4.632
2100.00	45.000	1.708	0.392	3.481	23537.000	30188.174	1.283	0.558	0.490	28.051	256.000	437.261	**1.708**	2.732
2120.00	45.000	1.471	0.392	3.481	23793.000	30564.776	1.285	0.555	0.321	13.588	256.000	376.602	**1.471**	1.151
2140.00	47.946	1.865	0.392	3.481	23964.000	30883.667	1.289	0.559	0.500	23.092	171.000	318.891	**1.865**	2.140
2160.00	48.000	2.287	0.392	3.481	24181.000	31379.932	1.298	0.565	0.550	29.643	217.000	496.265	**2.287**	3.670
2180.00	48.000	1.838	0.392	3.481	24415.000	31810.045	1.303	0.564	0.440	25.256	234.000	430.113	**1.838**	2.182
2200.00	48.000	1.432	0.392	3.481	24647.000	32142.252	1.304	0.562	0.440	11.807	232.000	332.207	**1.432**	1.308
2220.00	48.000	1.650	0.392	3.481	24865.000	32501.925	1.307	0.561	0.531	19.848	218.000	359.673	**1.650**	2.207
2240.00	48.000	1.755	0.392	3.481	25121.000	32951.122	1.312	0.558	0.521	21.071	256.000	449.197	**1.755**	2.794
2260.00	48.000	1.974	0.392	3.481	25345.000	33393.387	1.318	0.558	0.560	26.067	224.000	442.265	**1.974**	3.646
2280.00	51.919	2.813	0.392	3.481	25516.000	33874.428	1.328	0.580	0.641	29.993	171.000	481.041	**2.813**	4.988
2300.00	51.000	2.168	0.392	3.481	25790.000	34468.503	1.337	0.579	0.500	30.904	274.000	594.075	**2.168**	4.521

Statistics Report-History by Measurement and Agenda

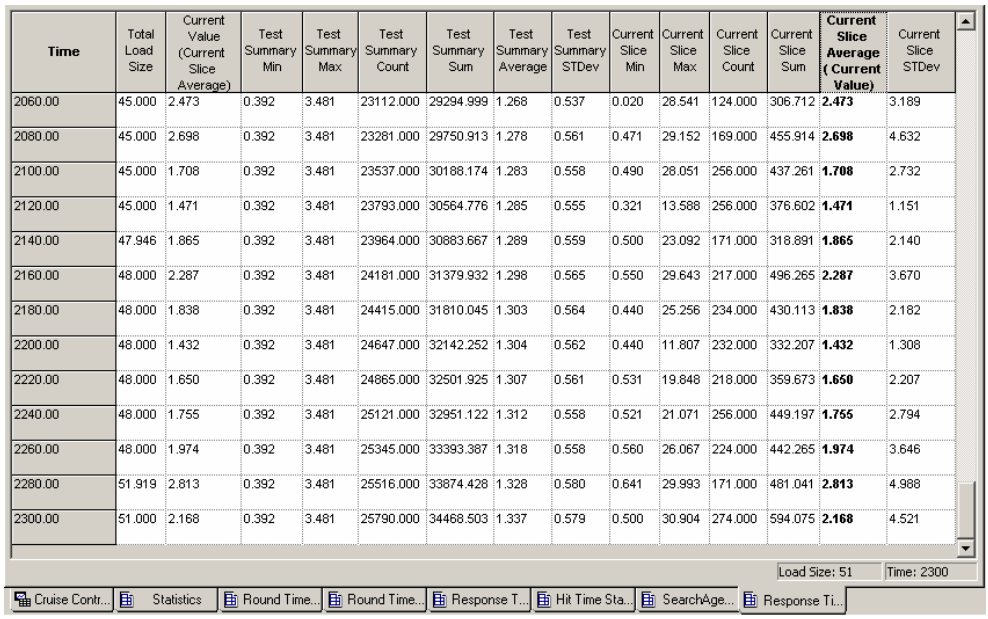
To open the History by Measurement and Agenda Report:

- Double-click on a button located at the intersection of a specific measurement row and Agenda column in the Statistics Report main window.
- Double-click on a selected Agenda name in the left hand column of a Summary by Measurement Report.
- Double-click on a selected measurement name in the far left column of a Summary by Agenda Report.

The History by Measurement and Agenda Report displays all the Attribute values recorded from the start of the current Load Session for the selected measurement and Agenda.

The following table displays all of the Response Time Attribute values for the Agenda, SearchAgenda.

Time	Total Load Size	Current Value (Current Slice Average)	Test Summary Min	Test Summary Max	Test Summary Count	Test Summary Sum	Test Summary Average	Test Summary STDev	Current Slice Min	Current Slice Max	Current Slice Count	Current Slice Sum	**Current Slice Average (Current Value)**	Current Slice STDev
2060.00	45.000	2.473	0.392	3.481	23112.000	29294.999	1.268	0.537	0.020	28.541	124.000	306.712	**2.473**	3.189
2080.00	45.000	2.698	0.392	3.481	23281.000	29750.913	1.278	0.561	0.471	29.152	169.000	455.914	**2.698**	4.632
2100.00	45.000	1.708	0.392	3.481	23537.000	30188.174	1.283	0.558	0.490	28.051	256.000	437.261	**1.708**	2.732
2120.00	45.000	1.471	0.392	3.481	23793.000	30564.776	1.285	0.555	0.321	13.588	256.000	376.602	**1.471**	1.151
2140.00	47.946	1.865	0.392	3.481	23964.000	30883.667	1.289	0.559	0.500	23.092	171.000	318.891	**1.865**	2.140
2160.00	48.000	2.287	0.392	3.481	24181.000	31379.932	1.298	0.565	0.550	29.643	217.000	496.265	**2.287**	3.670
2180.00	48.000	1.838	0.392	3.481	24415.000	31810.045	1.303	0.564	0.440	25.256	234.000	430.113	**1.838**	2.182
2200.00	48.000	1.432	0.392	3.481	24647.000	32142.252	1.304	0.562	0.440	11.807	232.000	332.207	**1.432**	1.308
2220.00	48.000	1.650	0.392	3.481	24865.000	32501.925	1.307	0.561	0.531	19.848	218.000	359.673	**1.650**	2.207
2240.00	48.000	1.755	0.392	3.481	25121.000	32951.122	1.312	0.558	0.521	21.071	256.000	449.197	**1.755**	2.794
2260.00	48.000	1.974	0.392	3.481	25345.000	33393.387	1.318	0.558	0.560	26.067	224.000	442.265	**1.974**	3.646
2280.00	51.919	2.813	0.392	3.481	25516.000	33874.428	1.328	0.580	0.641	29.993	171.000	481.041	**2.813**	4.988
2300.00	51.000	2.168	0.392	3.481	25790.000	34468.503	1.337	0.579	0.500	30.904	274.000	594.075	**2.168**	4.521



The times that the attribute values were recorded since the start of the current Load Session are listed in the column at the far left. The names of all the attributes totaled for the selected measurement and Agenda are displayed in the header at the top of the table.

Example:

This table displays all Total Load Size, Current Value, Min, Max, count and average values recorded for Response Time for SearchAgenda. Values were recorded starting at 20 seconds after the Load Session began.

Statistics Report-History by Measurement and Total

To open the History by Measurement and Total Report:

- Double-click on a button located at the intersection of a specific measurement row and the Total column in the Statistics Report main window,

  -Or-

  Double-click on the specific measurement name in the far left column of a Summary by Total Report.

The History by Measurement and Total Report displays all the Attribute values recorded from the start of the current Load Session for the selected measurement. These values are summed for all Agendas:

Time	Total Load Size	Current Value (Current Slice Average)	Test Summary Min	Test Summary Max	Test Summary Count	Test Summary Average	Current Slice Average (Current Value)
3260.00	69.000	1.958	0.392	3.481	37750.000	1.572	**1.958**
3280.00	69.000	2.899	0.392	3.481	37884.000	1.576	**2.899**
3300.00	69.000	2.553	0.392	3.481	38161.000	1.583	**2.553**
3320.00	71.959	1.768	0.392	3.481	38434.000	1.585	**1.768**
3340.00	72.000	2.483	0.392	3.481	38696.000	1.591	**2.483**
3360.00	72.000	2.132	0.392	3.481	39021.000	1.595	**2.132**
3380.00	72.000	2.093	0.392	3.481	39255.000	1.598	**2.093**
3400.00	72.000	4.293	0.392	4.293	39403.000	1.608	**4.293**
3420.00	72.000	2.481	0.392	4.293	39639.000	1.614	**2.481**
3440.00	72.000	1.916	0.392	4.293	39816.000	1.615	**1.916**
3460.00	74.877	1.949	0.392	4.293	40138.000	1.618	**1.949**
3480.00	75.000	2.466	0.392	4.293	40409.000	1.623	**2.466**
3500.00	75.000	2.466	0.392	4.293	40611.000	1.627	**2.466**

Load Size: 75 Time: 3500

Cruise... Statis... Roun... Roun... Resp... Hit Ti... Searc... Resp... Total ... Respo... Resp...

Listed in the far left-hand column are the times that attribute values were recorded since the beginning of the current Load Session. The names of all the attributes totaled for the selected measurement are displayed in the header at the top of the table.

C H A P T E R 2 3

Data Drilling and Table Compare Reports

When testing the integrity of your Web application, it is important to see not only if a given hit succeeded or failed, but also the reasons for the failure to pinpoint the hits exact weak points and bottlenecks. Data Drilling enables you to display a detailed description of all user-defined and named hits to the instance level. With Data Drilling you can view all hit failures and the reasons for them.

The WebLOAD Data Drilling reports and the Table Compare Viewer provide a graphic illustration of the problematic tables, making it simple to locate and correct the source of the error.

Data Drilling and Table Compare reports are available after the Load Session has completed.

In This Chapter

Data Drilling Reports 335
Table Compare Reports 341

Data Drilling Reports

Data Drilling provides both a global and detailed account of hit successes and failures allowing you to verify the functional integrity of your Web application at the per-client, per-transactions and per-instance level. The Data Drilling reports provide an extremely detailed yet easily accessible summary of all the statistical, timing and performance information collected over the course of the test session, which includes:

- A named entry for each HTTP action in the Agenda. Information is automatically saved for all actions tested using the Functional Verification Wizard. Information is also saved for all user-defined or manually named transactions.

  Click on a specific entry to zoom in on the detailed information available for that action. For each HTTP action tested and verified by your test Agenda, WebLOAD provides a complete breakdown of all relevant information, including:

  - A hierarchical tree of all the 'sub-steps' needed to complete the specified action. For example, a single Get of a simple Web page may include multiple 'sub-Gets' or hits of the various frames found on that page, each with its own pictures and tables, each of which is listed with its own 'sub-sub-Get' as well.

  Note: WebLOAD works with the 'visual set' of Web page elements associated with each action. Only the targeted frames that were actually accessed during a specific action will be saved as part of the DOM for that action, and not necessarily every single peripheral frame associated with the parent Web page.

  - Detailed information about the time required for each action and sub-transaction, including low-performance flags for actions that complete correctly, but take longer than a user-set time threshold to complete.
  - A hierarchical tree of all the objects accessed or otherwise associated with the specified action, together with detailed information about the time required to access each object. The timing information is broken down into the time required for each HTTP activity associated with that object, including connect time, send time, wait time (time to first byte), and final receive time.

- Entries for every timer triggered during the test session, including user-defined timers, WebLOAD-defined built-in timers, and automatic timers triggered each time ActiveX or Java object methods are activated. Click on a timer entry to bring up detailed information about that timer.
- Entries for every error event triggered during the test session by both WebLOAD and user-defined verification functions. Click on an event entry to bring up detailed information about that event.

Note: Data Drilling is only available to you if:

You initiated user-defined transactions in the scripting for functional testing.

You configured WebLOAD to Collect Instances in the Tools | Functional Testing tab.

You are using Internet Explorer 4 or higher.

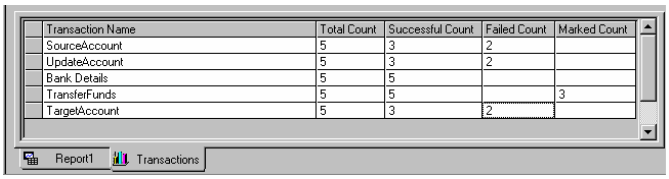
Opening a Data Drilling Report

To open a Data Drilling report:

- Select Reports | Data Drilling.

Viewing the Transaction Grid

The Transaction Grid is the entry point for accessing the Data Drilling reports provided by WebLOAD.



Transaction Name	Total Count	Successful Count	Failed Count	Marked Count
SourceAccount	5	3	2	
UpdateAccount	5	3	2	
Bank Details	5	5		
TransferFunds	5	5		3
TargetAccount	5	3	2	

Report1 Transactions

The Transactions Grid displays summary information for all user-defined and named transactions incorporated in the Agenda including:

Attribute	Explanation
Transaction Name	The name of the transaction.
Total Count	The total number of times the transaction was executed.
Successful Count	The number of successful executions of the transaction.
Failed Transactions	The number of failed executions of the transaction.
Marked Count	The number of times a lower level transaction, meaning a transaction nested within a higher level transaction, failed within the current transaction.

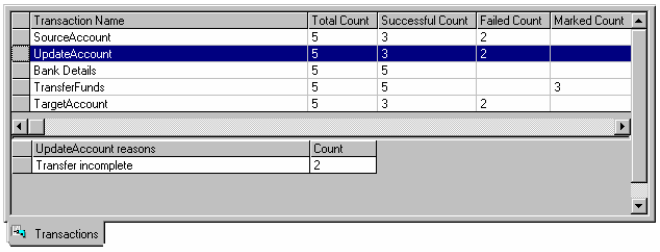
Note: Some of the transactions include both successes and failures.

Getting More Detailed Information

To see why a particular transaction failed select the Transaction Name on the Transaction Grid to display a Transaction Reason Failure Grid.

Viewing the Transaction Reason Failure Grid

The Transaction Reason Failure grid lists the reasons for the transaction failure, together with the total failure count for that transaction.



Transaction Name	Total Count	Successful Count	Failed Count	Marked Count
SourceAccount	5	3	2	
UpdateAccount	5	3	2	
Bank Details	5	5		
TransferFunds	5	5		3
TargetAccount	5	3	2	

UpdateAccount reasons	Count
Transfer incomplete	2

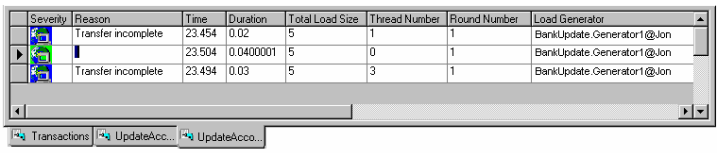
Transactions

In this example the UpdateAccount transaction is highlighted in the Transaction Grid. The Reason for the UpdateAccount transaction failure, Transfer incomplete displays immediately below the Transaction Grid. The Transaction Failure Reason Grid displays why two of the UpdateAccount transactions failed.

Getting More Detailed Information

To track down and correct the underlying cause of the failure you want to discover which transactions failed. Double-click on the reason failure row to open the Instance Grid.

Viewing the Instance Grid



The Instance Grid lists the individual instances of transaction failure, together with specific details about each instance.

The following information is included for each transaction instance displayed in the grid:

Attribute	Explanation
Severity	The severity level of this Transaction Instance failure, based on the return code set by the Agenda in the `Verification function`.
Reason	The reason that the Transaction Instance failed based on information supplied by the Agenda in the `SetFailureReason` function.
Time	The time that this Transaction Instance began execution, relative to the beginning of the current test session.
Duration	The time it took to complete the Transaction instance. If the Transaction Instance failed, the duration is the time until the transaction failed.
Total Load Size	The total number of Virtual Clients running on the Load Generator during this Transaction Instance.
Thread Number	The number of the current thread running during this Transaction Instance.
Round Number	The number of the current round running during this Transaction Instance.
Load Generator	The name of the Load Generator that executed the Transaction Instance.

The Instance Grid incorporates functional testing with performance testing and provides you with a detailed description of exactly which Transaction Instance failed, why it failed, and what else occurred at that point in the Agenda execution for that Load Generator.

Getting More Detailed Information

To track down and correct the underlying cause of the failure you might still need help understanding how all of this information fits together. Double-click on one of the Transaction Instances to display a complete Parent Transaction Instance Tree.

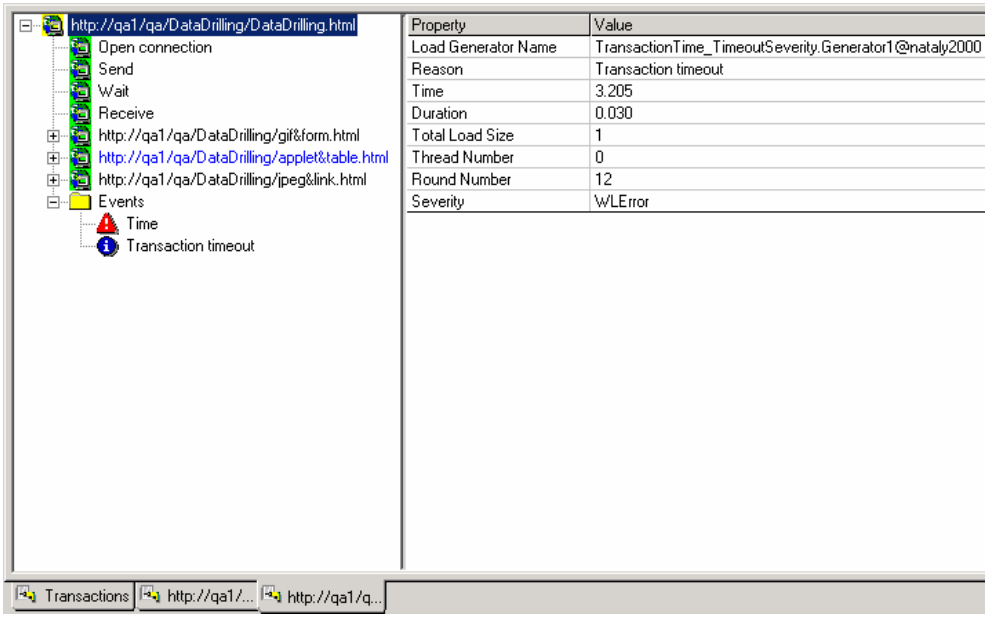
Viewing the Parent Transaction Instance Tree

The Parent Transaction Instance Tree:

- Provides a graphic presentation of the relationships between all of the parent and children transactions, providing a detailed breakdown of all sub-events that occurred as part of the selected Transaction. For example, a transaction may involve a series of steps, including opening a connection, sending a signal, waiting, receiving a response, etc.
- Lists all of the details about the selected Instance provided in the Instance Grid.

A transaction is selected in the tree at the left. The fact that this transaction triggered an error is indicated by the blue flag icon next to the transaction and the blue color of the transaction itself.

The properties for the selected Transaction Instance are displayed in a table on the right side of the window. The Transaction Instance Tree provides a broad understanding of how the failed Transaction Instance fits into the general Agenda execution.



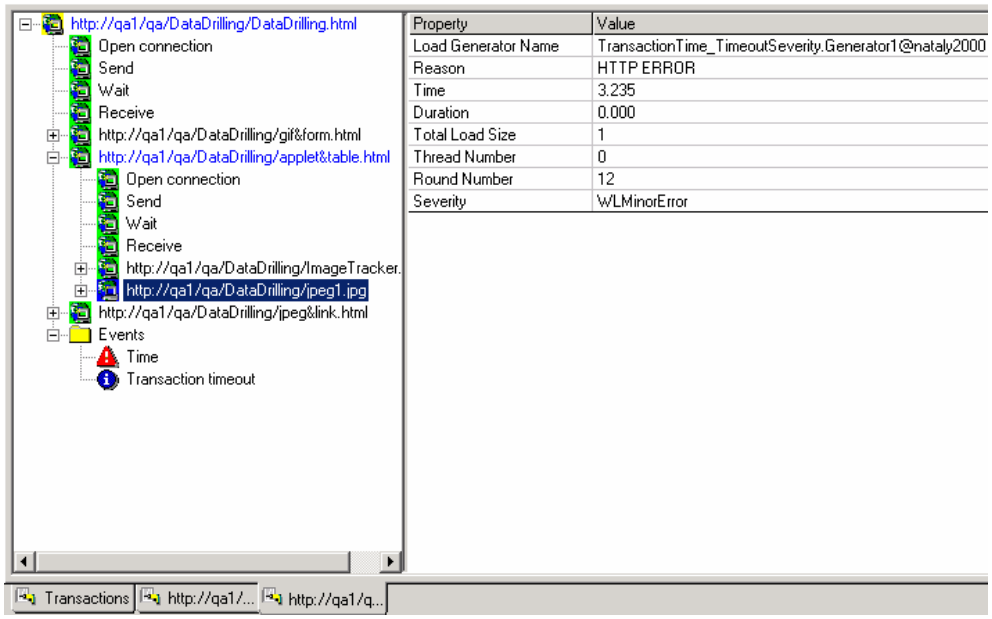
Getting More Detailed Information

For even more detailed information you may wish to see a breakdown of the transaction steps that were part of the parent transaction. For example, one of the child branches in the proceeding Transaction Instance Tree is also marked in blue, indicating that an error occurred at that point. Click on the error-flagged child branch in the Transaction Instance Tree to see a detailed breakdown of that specific event.

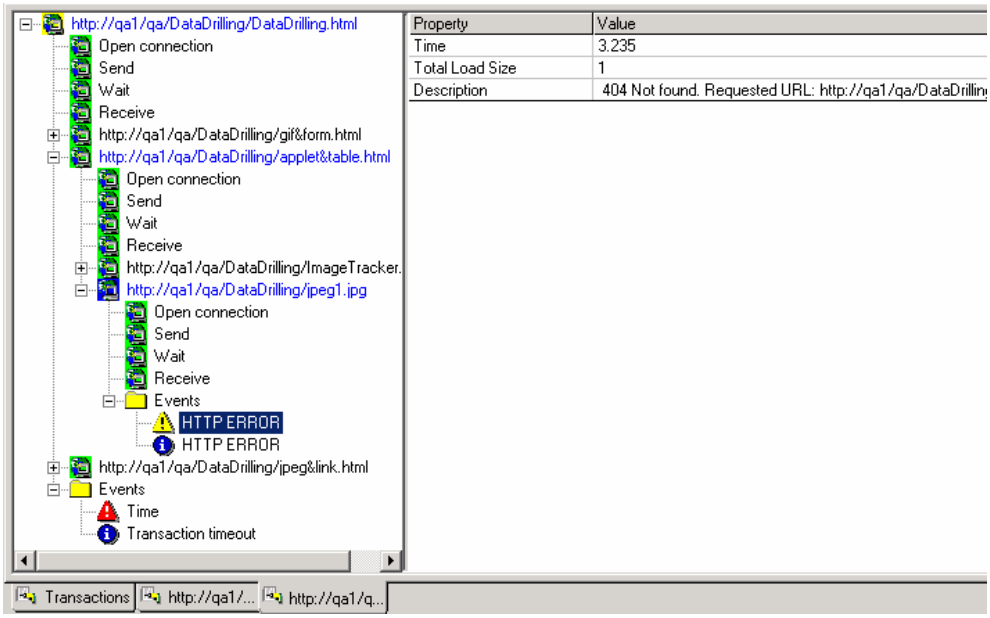
Viewing the Child Transaction Instance Tree

Child transactions are selected on the tree at the left. The fact that the transaction triggered an error is indicated by the blue flag on the icon next to the transaction and the blue color of the transaction itself in the tree. Details of the error are provided in the table at the right.

In the example below, the selected child transaction triggered an HTTP ERROR of severity Minor Error. The corresponding error entry in the Log Window is highlighted.



To see more information about the actual event that triggered this error, double click on the child transaction branch in the tree to expand its own sub tree, including its own Events sub-branch. Double click on more entries under the Events sub-branch to display more information about each selected item on the table at the right.

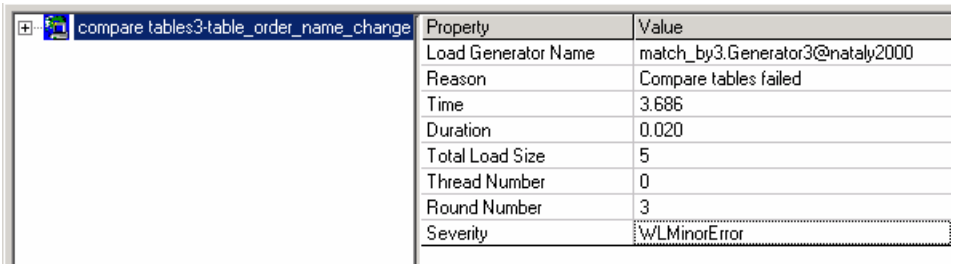


By working your way through the multiple levels of the Data Drilling reports you can achieve a complete picture of the exact point at which your Agenda failed, along with a complete history of the general system and events that led up to the failure. This level of detailed information simplifies your work when testing and debugging your Web site. Table Compare Events display graphically in the Table Compare Viewer. For more information on viewing the results of Table Compare Events graphically see Table Compare Reports (on page 341).

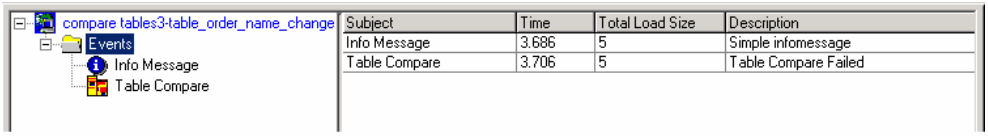
Table Compare Reports

When you use the WebLOAD Functional Testing feature, you select specific Agenda events to be tracked and verified as they occur. Your verification function records information about the specified transaction event and exactly what caused any errors or failures. The Table Compare Viewer provides a graphic representation of all Table Compare Events that failed during the Load Session.

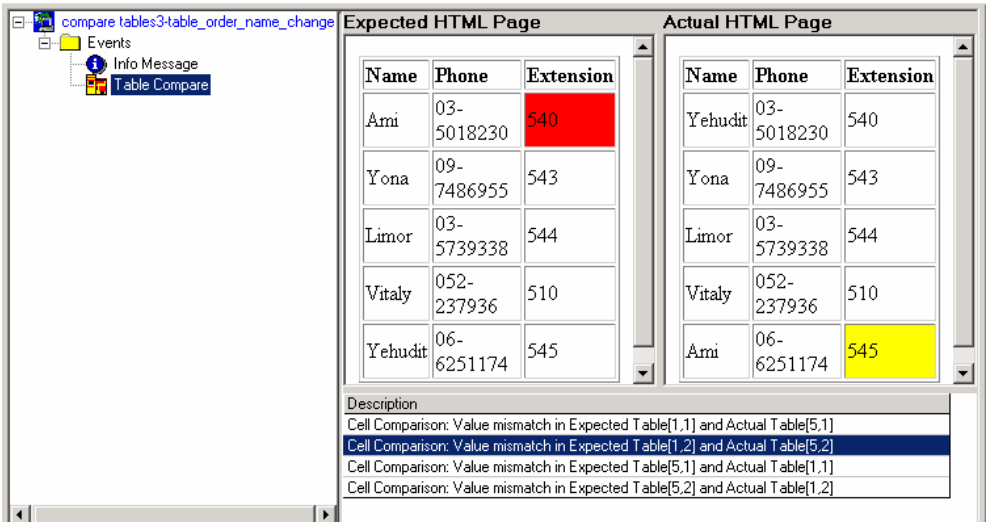
Note: The Table Compare Viewer is only available with Internet Explorer 5.



The first level indicates that a minor error caused the table comparison to fail. For more information, double click on the parent branch in the tree to enter the second report level, expanding the Events branch under the selected transaction:



The second level tree lists the events that triggered the error. For an exact illustration of the problem, click on the Table Compare branch:



The screen now displays the source of the error, showing both the expected and actual tables with the mismatched table cells marked and the corresponding information highlighted in the Description and Log windows.

Opening the Table Compare Viewer

To open the Table Compare Viewer:

1. Double click on a Table Compare Failed User Note,

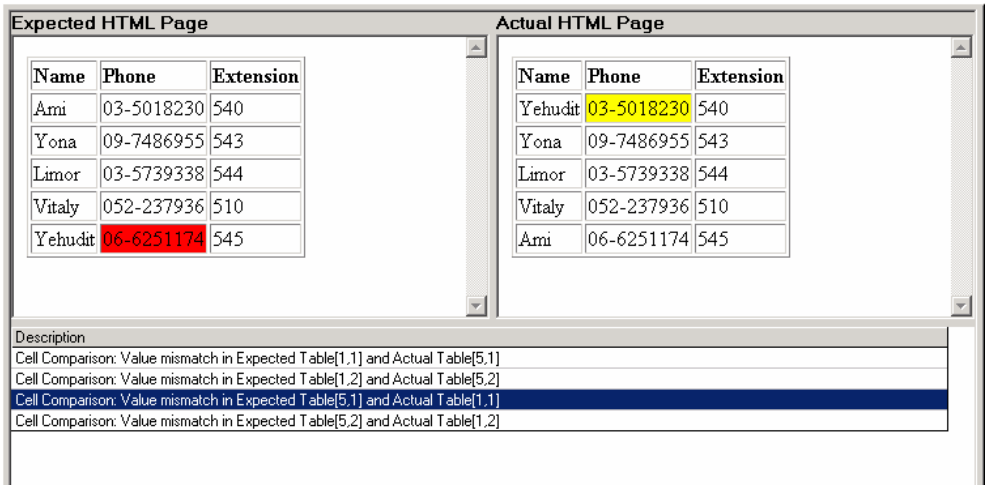
   -Or-

   Double click on the paperclip adjacent to the User Note in the Log Window.

2. Click on a specific Table Compare Event in the Transaction Instance Tree.

   Note: The Transaction Instance Tree is available only if User-Defined Transactions were initiated in the Agenda for functional testing and WebLOAD is configured to Collect Instances in the Tools | Default Options | Functional Testing tab.

   The Table Compare Viewer appears in the Results Window.



Features of this report

- The upper part of the viewer graphically displays the tables, which were compared in each particular Event. The table on the left represents the Expected Table and the table on the right displays the Actual Table.
- The lower part of the viewer displays the list of failed Table Compare transactions that occurred during the test session for the particular Event, with a detailed description of why the transaction failed.

Click one of the descriptions displayed in the lower half of the Results Window to highlight where the transaction failed in the Tables. The cells in which the discrepancy occurred are highlighted in both the Expected Table and the Actual Table.

C H A P T E R 2 4

Exporting Performance Reports

Performance Reports are the product of your test Session. Performance Reports enable you to combine several different measurements into a single report. You are able to view the data accumulated in the NT and Unix Performance Monitors together with the performance data from the Load Generators and Probing Clients. All of the data displayed in the Integrated Reports and Statistics can be exported to HTML, Microsoft Excel and to Tab files.

In This Chapter

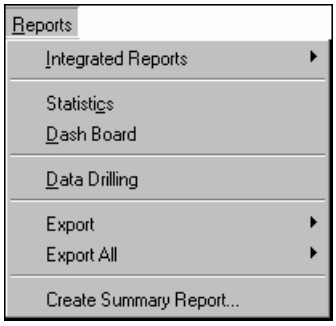
Report Export Options 345
Exporting Selective WebLOAD Report Data 347
Exporting All Statistical Data 353
Customizing the Export File 357
Creating a Summary Report 359

Report Export Options

WebLOAD enables you export three different types of reports:

- Reports containing only the data specified.
- General reports containing all the data from your test Session.
- Short summary reports that provide an overview of your Load Session testing.

Access the export options through the Reports menu.

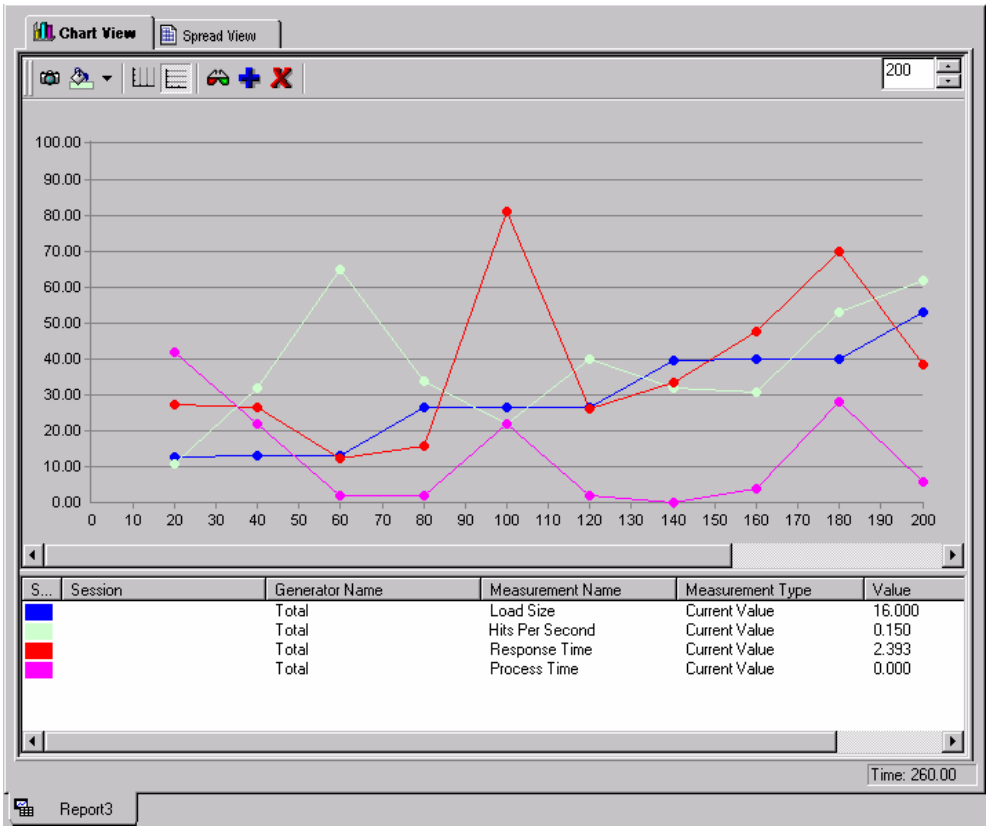


The following table describes each export option.

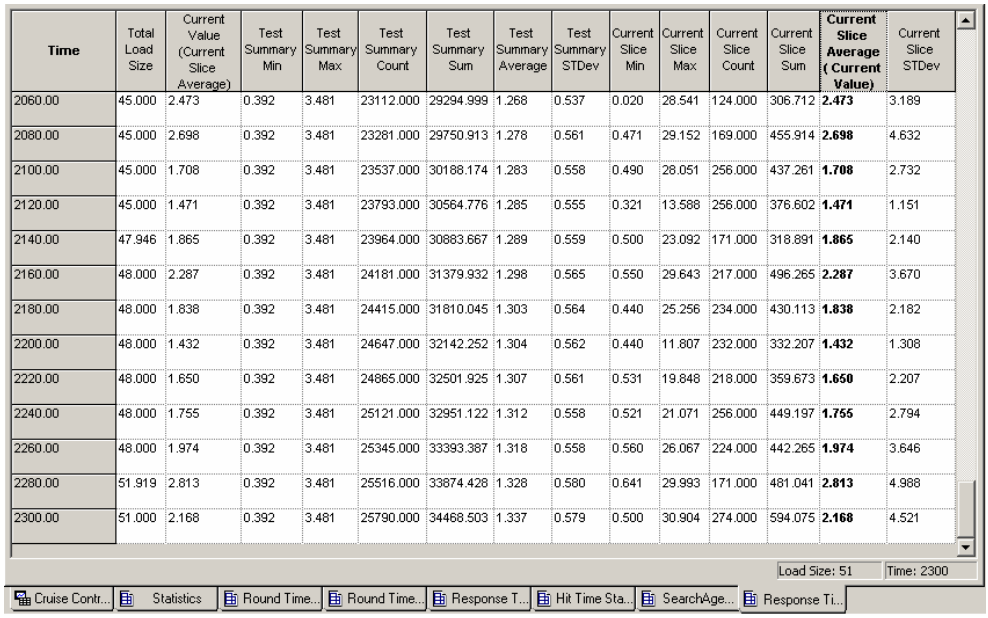
Category	Description
Export	Exports selective WebLOAD report data.
Export All	Exports all of the statistical data gathered by WebLOAD throughout the test Session.
Create Summary Report	Creates and exports a report that summarizes the results of your WebLOAD testing.

Exporting Selective WebLOAD Report Data

The following is a graphic presentation of an Integrated Report. This report displays the data for LOAD SIZE, HITS PER SECOND, RESPONSE TIME, and PROCESS TIME.



The screen below displays the data for the Response Time in tabular format.



Time	Total Load Size	Current Value (Current Slice Average)	Test Summary Min	Test Summary Max	Test Summary Count	Test Summary Sum	Test Summary Average	Test Summary STDev	Current Slice Min	Current Slice Max	Current Slice Count	Current Slice Sum	**Current Slice Average (Current Value)**	Current Slice STDev
2060.00	45.000	2.473	0.392	3.481	23112.000	29294.999	1.268	0.537	0.020	28.541	124.000	306.712	**2.473**	3.189
2080.00	45.000	2.698	0.392	3.481	23281.000	29750.913	1.278	0.561	0.471	29.152	169.000	455.914	**2.698**	4.632
2100.00	45.000	1.708	0.392	3.481	23537.000	30188.174	1.283	0.558	0.490	28.051	256.000	437.261	**1.708**	2.732
2120.00	45.000	1.471	0.392	3.481	23793.000	30564.776	1.285	0.555	0.321	13.588	256.000	376.602	**1.471**	1.151
2140.00	47.946	1.865	0.392	3.481	23964.000	30883.667	1.289	0.559	0.500	23.092	171.000	318.891	**1.865**	2.140
2160.00	48.000	2.287	0.392	3.481	24181.000	31379.932	1.298	0.565	0.550	29.643	217.000	496.265	**2.287**	3.670
2180.00	48.000	1.838	0.392	3.481	24415.000	31810.045	1.303	0.564	0.440	25.256	234.000	430.113	**1.838**	2.182
2200.00	48.000	1.432	0.392	3.481	24647.000	32142.252	1.304	0.562	0.440	11.807	232.000	332.207	**1.432**	1.308
2220.00	48.000	1.650	0.392	3.481	24865.000	32501.925	1.307	0.561	0.531	19.848	218.000	359.673	**1.650**	2.207
2240.00	48.000	1.755	0.392	3.481	25121.000	32951.122	1.312	0.558	0.521	21.071	256.000	449.197	**1.755**	2.794
2260.00	48.000	1.974	0.392	3.481	25345.000	33393.387	1.318	0.558	0.560	26.067	224.000	442.265	**1.974**	3.646
2280.00	51.919	2.813	0.392	3.481	25516.000	33874.428	1.328	0.580	0.641	29.993	171.000	481.041	**2.813**	4.988
2300.00	51.000	2.168	0.392	3.481	25790.000	34468.503	1.337	0.579	0.500	30.904	274.000	594.075	**2.168**	4.521

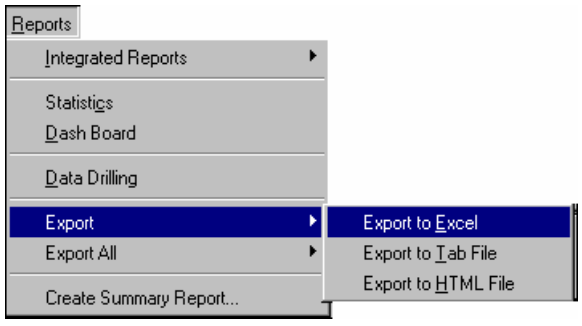
All of the examples in this section refer to these reports (EXAMPLE 1 and EXAMPLE 2)

Exporting Performance Reports to Excel

You can view all of the data available in the Performance Reports from a single Excel spreadsheet. After you have exported a Performance Report to Excel, you work with the exported spreadsheet as you would with any other Excel spreadsheet.

To export Performance Reports to Excel:

1. Display the Integrated Report you want to export. If there is more than one report open, focus on the report you want to export.
2. Select Reports | Export | Export to Excel,



-Or-

Click the Export to Excel toolbar button.

The report is sent to Excel, which opens automatically and displays the Integrated Report.

Notes:

If Excel is not installed, you receive the following error: Cannot Open Excel.

Exporting a large amount of data to Excel may take some time. Wait until the export to Excel has completed before continuing work with the Console.

3. Save the Excel file.

The following screen shows an Integrated Report after it is exported to an Excel spreadsheet. This screen shows a tabular presentation for a graphic screen displayed in WebLOAD (EXAMPLE 1).

Microsoft Excel - Book1

File Edit View Insert Format Tools Data Window Help

Arial

A1 = Time

	A	B	C	D	E	F
1	Time	Total/ Load Size/ Current Value	Total/ Hits Per Second/ Current Value	Total/ Response Time/ Current Value	Total/ Process Time/ Current Value	
2	20	3.855	0.55	2.746	0.021	
3	40	4	1.6	2.667	0.011	
4	60	4	3.25	1.246	0.001	
5	80	7.955	1.7	1.587	0.001	
6	100	8	1.1	8.101	0.011	
7	120	8	2	2.638	0.001	
8	140	11.92	1.6	3.345	0	
9	160	12	1.55	4.773	0.002	
10	180	12	2.65	6.977	0.014	
11	200	15.96	3.1	3.867	0.003	
12	220	16	3.35	2.746	0.001	
13	240	16	2.75	5.613	0.005	
14	260	16	0.15	2.393	0	
15						

Sheet1 Sheet2 Sheet3

Ready NUM

Features of this Report

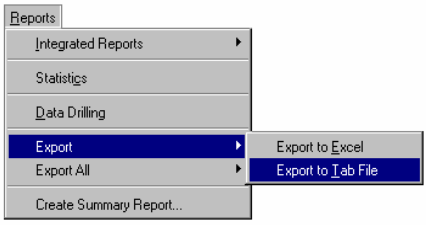
- The first column contains the times (in seconds) that the measurement values were recorded.
- The top row contains the name of each measurement in the following format: Agenda / Measurement / Attribute. The names of the measurements are color-coded to match the measurements in the Chart View of the report.
- Integrated Reports displayed in Chart View are exported to Excel in tabular format.

See Customizing the Export File (on page 357), for information on configuring the sampling frequency and the delimiters of your exported report.

Exporting Performance Reports to a Tab File

To export Performance Reports to a Tab file:

1. Display the report you want to export. If there is more than one report open, focus on the report you want to export.
2. Select Reports | Export | Export to Tab File,



-Or-

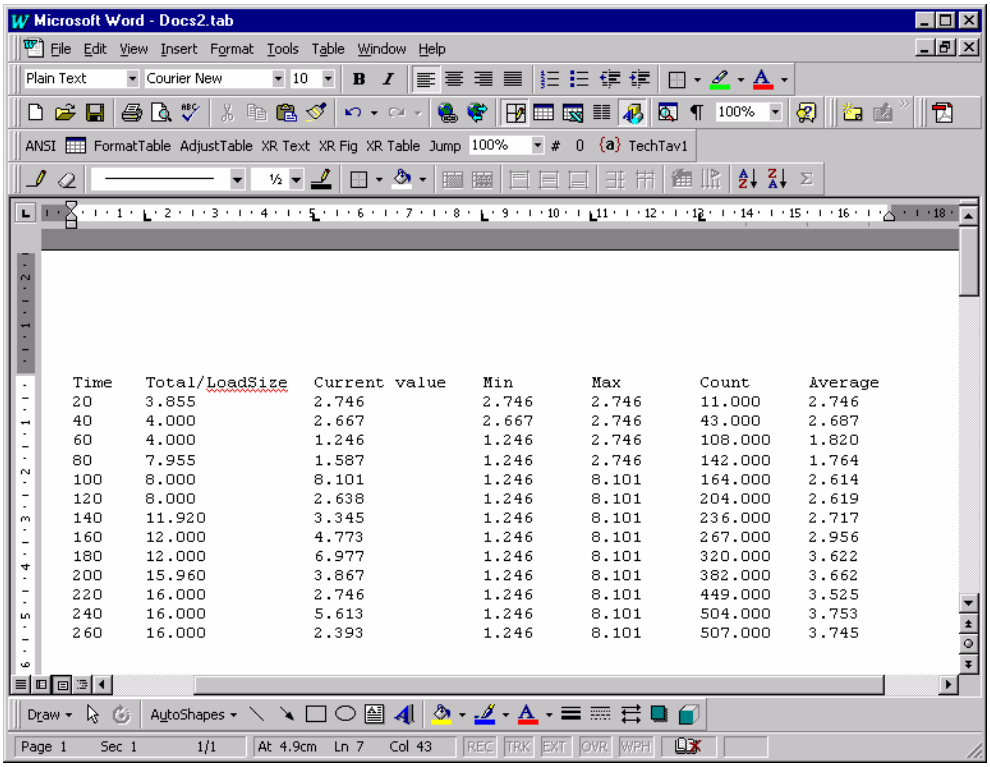
Click on the Export to Tab File toolbar button.

The Windows Save As dialog box displays.

3. Enter the name of your new Tab file. The file has a `*.Tab` extension.
4. Click Save.

Once the file is saved, it is displayed. Configure the viewer and the view options using the Export options dialog box, see Customizing the Export File (on page 357).

The following screen shows an Integrated Report after it is exported to a Tab file and open in Microsoft Word (using tab as a delimiter). You can change the presentation of this file using the Export options dialog box.



Time	Total/LoadSize	Current value	Min	Max	Count	Average
20	3.855	2.746	2.746	2.746	11.000	2.746
40	4.000	2.667	2.667	2.746	43.000	2.687
60	4.000	1.246	1.246	2.746	108.000	1.820
80	7.955	1.587	1.246	2.746	142.000	1.764
100	8.000	8.101	1.246	8.101	164.000	2.614
120	8.000	2.638	1.246	8.101	204.000	2.619
140	11.920	3.345	1.246	8.101	236.000	2.717
160	12.000	4.773	1.246	8.101	267.000	2.956
180	12.000	6.977	1.246	8.101	320.000	3.622
200	15.960	3.867	1.246	8.101	382.000	3.662
220	16.000	2.746	1.246	8.101	449.000	3.525
240	16.000	5.613	1.246	8.101	504.000	3.753
260	16.000	2.393	1.246	8.101	507.000	3.745

Features of this Report

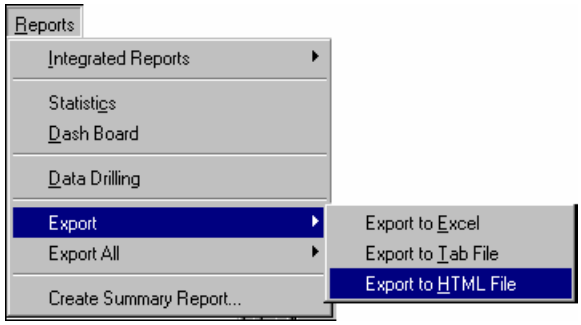
- The first column contains the times (in seconds) that the measurement values were recorded.
- The top row contains of the name of each measurement in the following format: Agenda / Measurement / Attribute.
- Integrated Reports displayed in Chart View is exported to Tab files in tabular format.

See Customizing the Export File (on page 357), for information on configuring the sampling frequency, the delimiters of your exported table, the viewer, and the view options.

Exporting Performance Reports to HTML

To export Performance Reports to HTML:

1. Display the report you want to export. If there is more than one report open, focus on the report you want to export.
2. Select Reports | Export | Export to HTML,

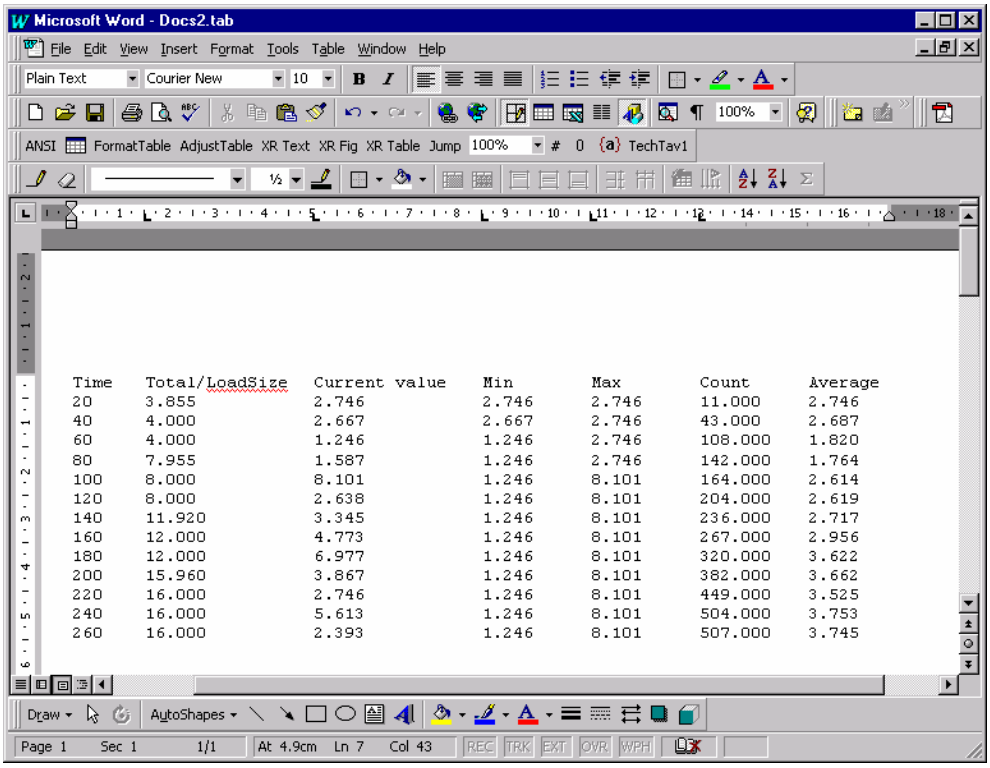


-Or-

Click on the Export to HTML toolbar button.

The report is displayed in HTML.

The following screen shows an Integrated Report after it is exported to HTML.



Time	Total/LoadSize	Current value	Min	Max	Count	Average
20	3.855	2.746	2.746	2.746	11.000	2.746
40	4.000	2.667	2.667	2.746	43.000	2.687
60	4.000	1.246	1.246	2.746	108.000	1.820
80	7.955	1.587	1.246	2.746	142.000	1.764
100	8.000	8.101	1.246	8.101	164.000	2.614
120	8.000	2.638	1.246	8.101	204.000	2.619
140	11.920	3.345	1.246	8.101	236.000	2.717
160	12.000	4.773	1.246	8.101	267.000	2.956
180	12.000	6.977	1.246	8.101	320.000	3.622
200	15.960	3.867	1.246	8.101	382.000	3.662
220	16.000	2.746	1.246	8.101	449.000	3.525
240	16.000	5.613	1.246	8.101	504.000	3.753
260	16.000	2.393	1.246	8.101	507.000	3.745

Features of this Report

- The first column contains the times (in seconds) that the measurement values were recorded.
- The top row contains of the name of each measurement in the following format: Agenda / Measurement / Attribute.
- Integrated Reports displayed in Chart View is exported to HTML in tabular format.

See Customizing the Export File (on page 357), for information on configuring the sampling frequency, the delimiters of your exported table, the viewer, and the view options.

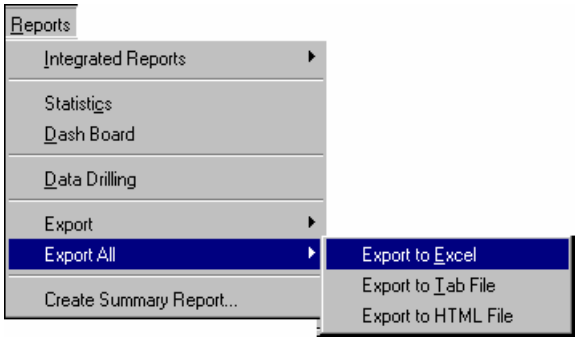
Exporting All Statistical Data

WebLOAD provides a wealth of statistical information about the performance of the Web application being tested. All of the statistical data gathered by WebLOAD can be exported to Microsoft Excel, Tab files and HTML. Exporting the WebLOAD statistical data allows you to view and manipulate your data and perform cross report comparisons.

Exporting All Statistics to Excel

To export all Statistics to Excel:

- Select Report | Export All | Export to Excel,

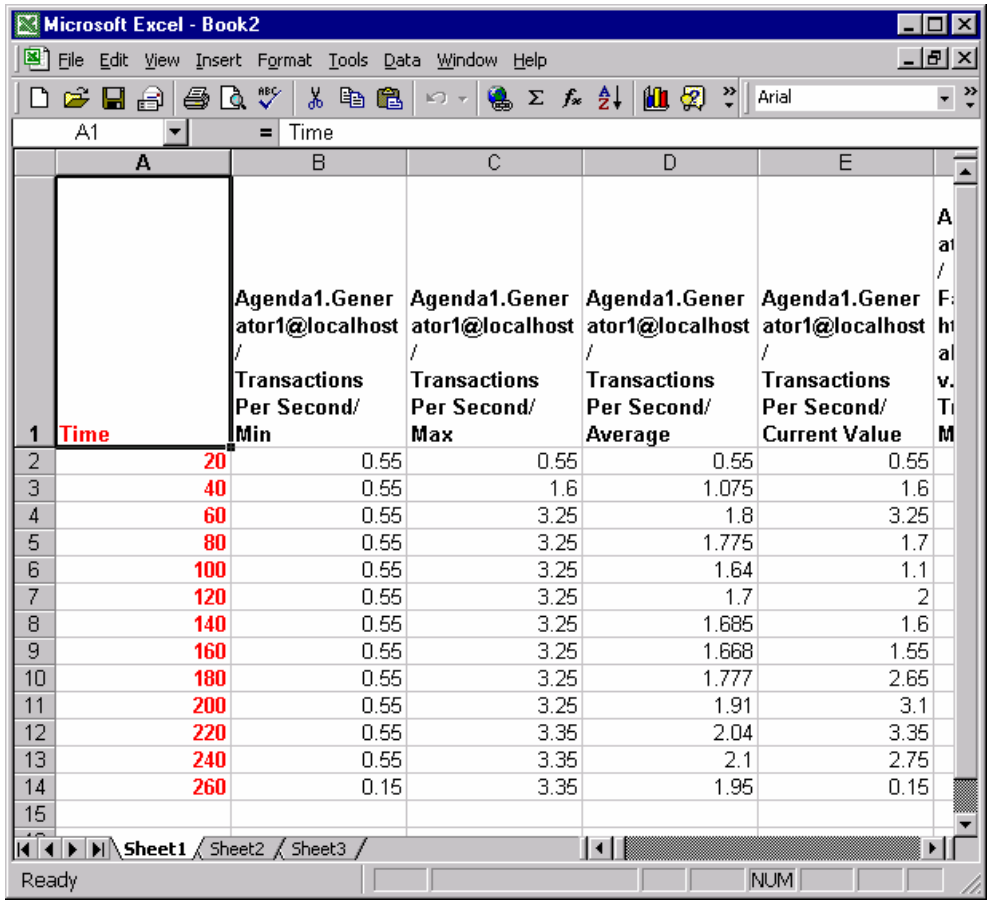


-Or-

Click on the Export All to Excel toolbar button.

Excel opens automatically and displays all of the Statistics gathered by WebLOAD.

The following screen shows the Statistics as they appear when exported to Excel.



1	A Time	B Agenda1.Generator1@localhost / Transactions Per Second/ Min	C Agenda1.Generator1@localhost / Transactions Per Second/ Max	D Agenda1.Generator1@localhost / Transactions Per Second/ Average	E Agenda1.Generator1@localhost / Transactions Per Second/ Current Value	A at / F: ht al v. T: M
2	20	0.55	0.55	0.55	0.55	
3	40	0.55	1.6	1.075	1.6	
4	60	0.55	3.25	1.8	3.25	
5	80	0.55	3.25	1.775	1.7	
6	100	0.55	3.25	1.64	1.1	
7	120	0.55	3.25	1.7	2	
8	140	0.55	3.25	1.685	1.6	
9	160	0.55	3.25	1.668	1.55	
10	180	0.55	3.25	1.777	2.65	
11	200	0.55	3.25	1.91	3.1	
12	220	0.55	3.35	2.04	3.35	
13	240	0.55	3.35	2.1	2.75	
14	260	0.15	3.35	1.95	0.15	
15						

Features of this Report

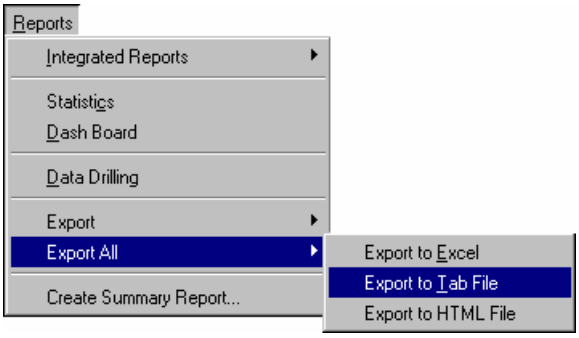
- The first column contains the times (in seconds) that the measurement values were recorded.
- The top row contains the name of each measurement in the following format: Agenda / Measurement / Attribute.
- Data is exported to Excel by columns. When the end of a Sheet is reached, a new Excel Sheet is opened.

See Customizing the Export File (on page 357), for information on configuring the sampling frequency and the delimiters of your exported table.

Exporting All Statistics to a Tab File

To export all statistics to a Tab file:

1. Select Reports | Export All | Export to Tab File,



   -Or-

   Click on the Export All to Tab File toolbar button.

   The Windows Save As dialog box is displayed.

2. Enter the name of your new Tab file. The file has a `*.Tab` extension.
3. Click Save.

   Once the file is saved, the data is displayed.

Configure the viewer and the view options in the Export options dialog box, see C*ustomizing the Export File (on page* 357*).*

Change the presentation of this file using the Export options dialog box.

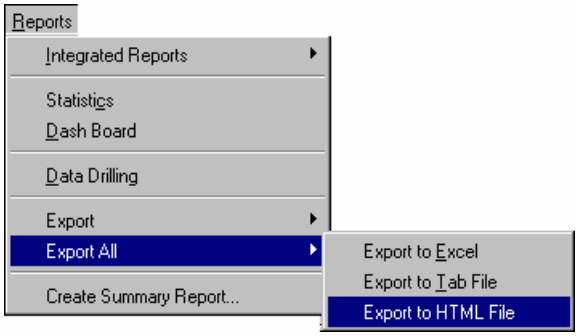
- The left column contains the times (in seconds) that the measurement values were recorded.
- The top row contains the name of each measurement in the following format: Agenda / Measurement / Attribute.

See Customizing the Export File (on page 357), for information on configuring the sampling frequency, the delimiters of your exported table, the viewer, and the view options.

Exporting All Statistics to HTML

To export all statistics to HTML:

- Select Reports | Export All | Export to HTML,



-Or-

Click on the Export All to HTML toolbar button.

Change the presentation of this file using the Export options dialog box.

- The left column contains the times (in seconds) that the measurement values were recorded.
- The top row contains the name of each measurement in the following format: Agenda / Measurement / Attribute.

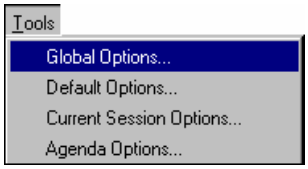
See Customizing the Export File (on page 357), for information on configuring the sampling frequency, the delimiters of your exported table, the viewer, and the view options.

Customizing the Export File

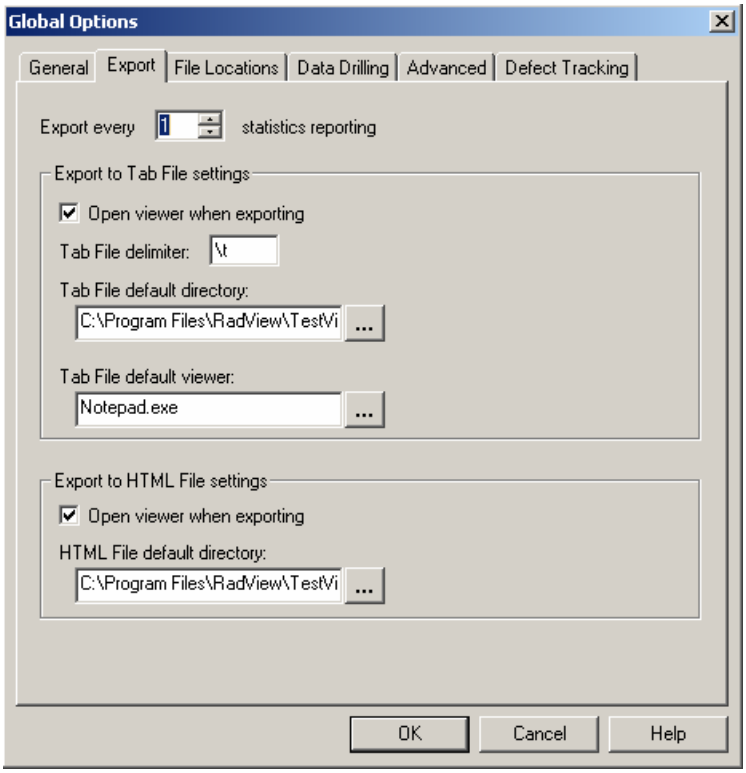
Before data is exported you can define the way the data is displayed in the exported file.

To customize the export file:

1. Select Tools | Global Options.



The Global Options dialog box opens:



2. Select the Export tab.

The Export dialog box contains the following fields:

Field Name	Description
Export Every Statistics Reporting	The frequency of the number of samples exported. For example, if “1” is entered for this value, WebLOAD exports every sample taken. If “2” is entered for this value, WebLOAD exports every second sample. When configuring this option, you must consider the setting selected for sampling. If for example, the Statistics Reporting Resolution is set to 20 seconds and Export Every Statistics Reporting is set to 2, the measurement exported reflects a sampling every 40 seconds. The measurements are the measurements taken at the intervals: 20, 60, 100, 140… This option is useful in cases where there is a large amount of data due to a long test Session.
Open Viewer When Exporting	Check to launch the Tab File and HTML Default Viewer after the Tab or HTML file is exported. This is checked by default

Field Name	Description
Tab File Delimiter	Definition of the delimiter used when exporting the statistical data to a tab file. All strings can be used as delimiters by entering the string in the Tab File Delimiter field. ◆ To use tab characters as delimiters enter \t. ◆ To use a new line as a delimiter, enter \r or \n. ◆ To use a space as a delimiter, use \s.
Tab \ HTML File Default Directory	The default directory for saving Tab and HTML files.
Tab File Default Viewer	The default viewer to open after the Tab file is exported.

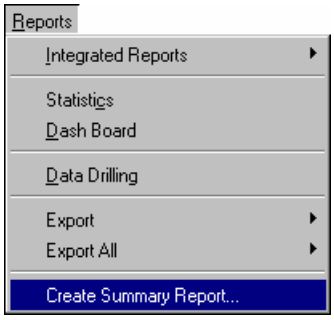
1. Click OK.

Creating a Summary Report

WebLOAD allows you to create a report that summarizes the results of your WebLOAD testing. Summary reports are useful as a clear and easy way to present the results of your WebLOAD testing to executives in the company. The Summary Report Wizard helps you create an easy to read report that summarizes the data of interest to you so that it may be summarized.

To create a Summary Report:

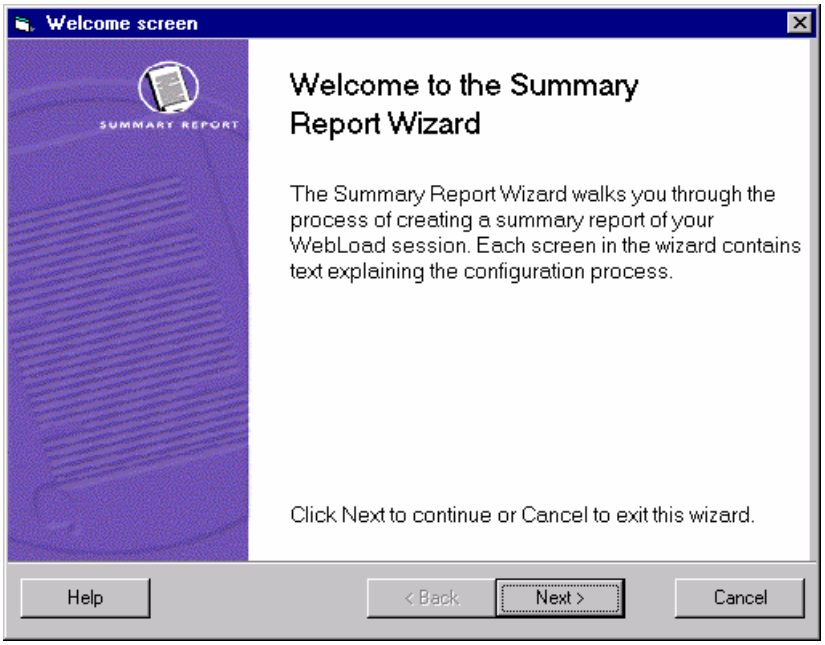
1. Select Reports | Create Summary Report,



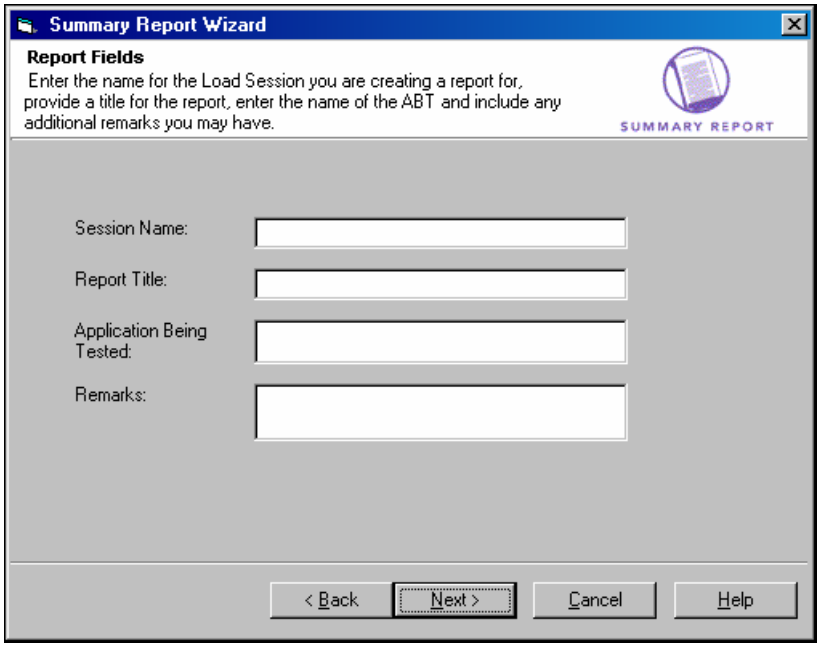
-Or-

Click the Create Summary Report toolbar button.

The Summary Report Wizard opens.



Customizing the Report Fields

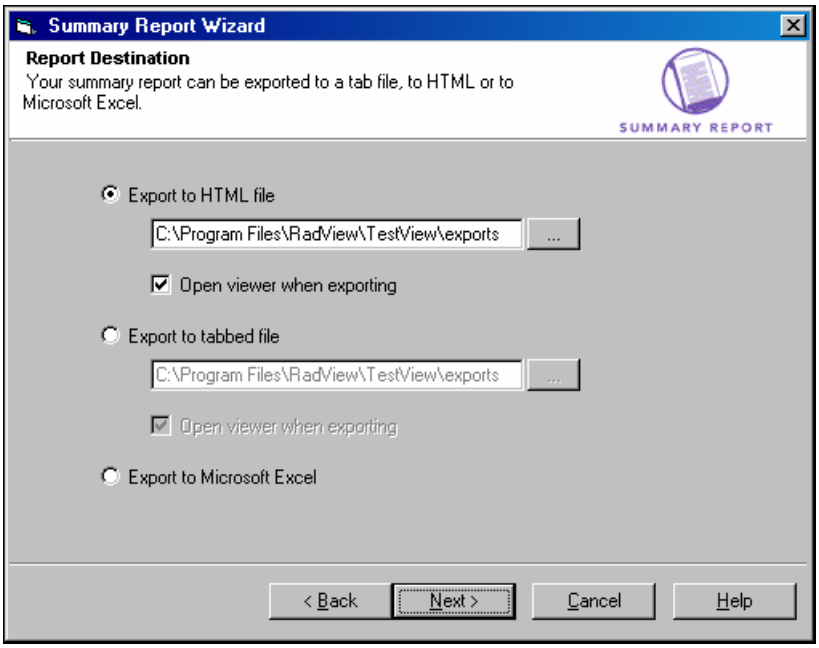


Begin your work in the Summary Report Wizard by customizing the report.

To customize the report:

1. Enter the name of the Load Session for which you are creating a report.
2. Provide a title for the report.
3. Include the name of the application you tested.
4. Include any other remarks you may have.
5. Click Next to move to the next step, defining how you want to view the Summary Report.

Report Destination



You may export the Summary Report to Microsoft Excel, HTML or a Tab file.

Exporting a Summary Report to a Tab File

To export a Summary Report to a Tab file:

1. Select Export to tabbed file.
2. Enter the name and path of the new file in the Export to tabbed file field.
3. Select Launch Tab File Editor with the report to view the file immediately.

   If you do not select this option, you may view the report later by browsing your system for the location of the file and opening it manually.

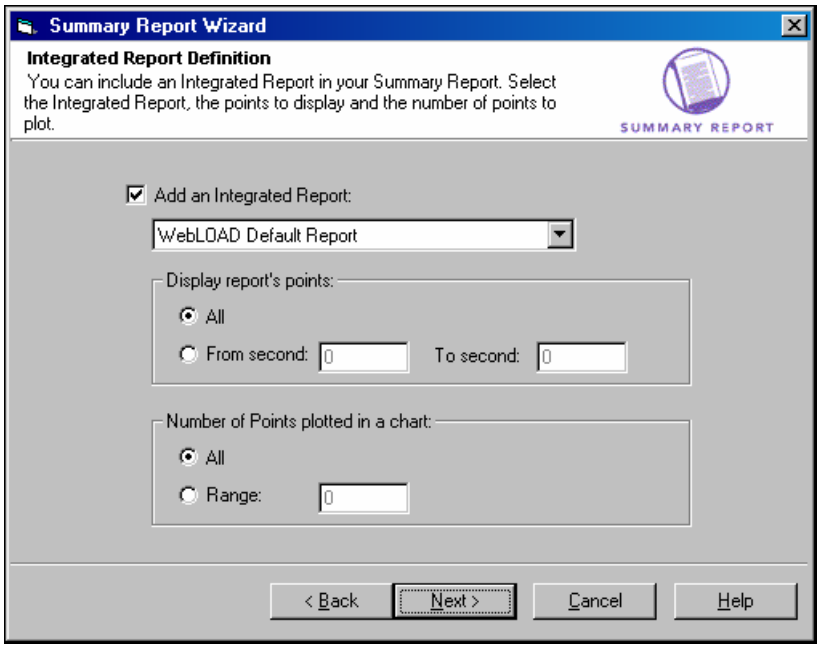
Exporting a Summary Report to Excel

To export a Summary Report to Excel:

1. Select Export to Microsoft Excel.

   Excel opens automatically, and the Summary Report is displayed.
2. Click Next.

Integrated Report Definition



To define an Integrated Report:

1. Select the Add an Integrated Report checkbox.
2. Select the Integrated Report you wish to include from the combo box.

   If you are exporting the Summary Report to Microsoft Excel, you may:

   - Define the time range from the Integrated Report to be displayed as a graph. Select All to display the entire report, or enter the time range you wish to be displayed as a graph in the Summary Report.
   - Define the number of points to be plotted on the graph generated in the Summary Report. Select All to include all of the points from the defined time range in one graph, or enter the number of points you wish to be plotted on each graph.
3. Click Finish.

   The Summary Report is created.

Sheet 1 of the Summary Report contains general information about the test Session, including:

- The participating hosts and Agendas.
- All information displayed in the report that was exported.
- Any remarks you may have included while configuring the report in the Summary Report Wizard.
- A graphic presentation of the exported report.

Sheet 2 of the Summary Report displays a tabular view of the exported report.

C H A P T E R 2 5

Performance Measurements Manager

WebLOAD provides the Performance Measurements Manager (PMM) for monitoring the performance of various server-side applications, databases, stream technology, system and Web server measurements in real-time while your test is running.

In This Chapter

Using the Performance Measurements Manager 365
About the Performance Measurements Manager 366
Opening the Performance Measurements Manager 366
Using PMM Configuration Files 373
Configuring the Data Sources, Hosts, and Measurements to Monitor 375
Application Server Monitors 385
Database Monitors 387
Web Server Monitors 391
Monitoring Generic SNMP Data Sources 392

Using the Performance Measurements Manager

Using the PMM, you configure WebLOAD to collect data for the hosts and services available from the server, to provide a complete picture of application performance while you are testing your Web application. Thus, in addition to the data accumulated from load testing, you receive data on the performance of your server.

Using the PMM, you can monitor:

- Application Server Resources
- Database Resources
- Stream Technology Resources
- System Resources
- Web Server Resources

About the Performance Measurements Manager

Using PMM you can monitor a wide range or performance objects such as application data, database, Web server, stream technology and system performance statistics in real-time during your test sessions. Each performance object, or data source, provides measurements that represent detailed data on specific aspects of the system or service.

Use the performance data collected to:

- Understand the effect of the load on your system's resources.
- Observe changes in resource usage so you can plan for future upgrades.
- Test application changes by monitoring the results.
- Diagnose problems and target components or processes for optimization.

PMM is where you set up the data sources, including the hosts and measurements, to monitor. You can maintain multiple configurations of servers, hosts and measurements to monitor.

Each configuration includes:

- Data sources
- Hosts
- Selected measurements from each data source on each host

To activate the monitors, you must configure the list of data sources, hosts and measurements to monitor before running you load test. While running your test, you can create Integrated Reports to display performance statistics for any monitors configured in PMM. Use the PMM Wizard to add data sources to the PMM main window and create configuration files.

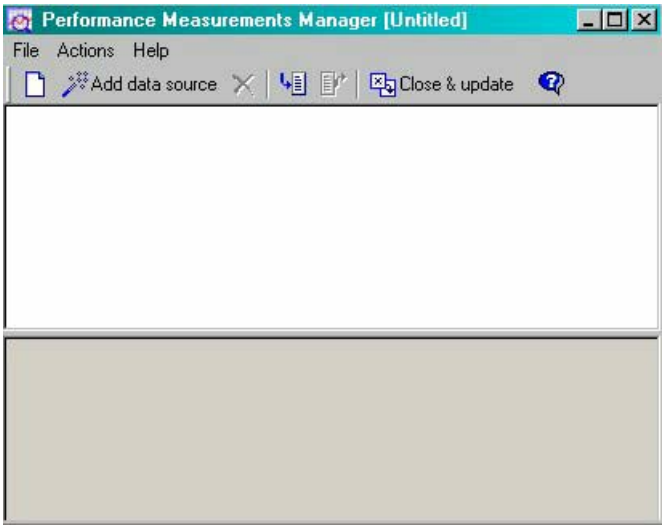
PMM can be run from within WebLOAD or as a standalone tool. Run PMM from within WebLOAD to create define the performance objects to monitor and assign the configuration to a WebLOAD template, or run the standalone tool to create a library of configuration files for use in future WebLOAD tests.

Opening the Performance Measurements Manager

PMM can be opened as a standalone tool for configuring the data to monitor before you even define your test or from within WebLOAD to assign PMM configurations to your test template.

- To open the PMM from within WebLOAD, select Session Control | Performance Measurements Manager.
- To open the PMM as a standalone tool select Start | All Programs | TestView | Tools | Performance Measurements Manager.

PMM opens with no configured data sources listed.



The Performance Measurements Manager Main Window

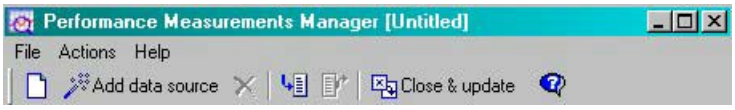
The PMM main window is where you set up the configuration file of data sources, hosts, and performance measurements to monitor. The main window displays the data sources configured to be monitored together with the hosts and the performance measurements being monitored on each host.

The main window consists of the following:

- Menu bar
- Toolbar
- Configuration window
- Description pane

Menu Bar

The PMM menu bar is located at the top of the window.



This section discusses each of the PMM menus, their options and their related toolbar buttons.

File Menu

The File menu includes the following functions for PMM configuration file management.

Note: Menu entries vary depending if PMM is run from within WebLOAD or as a standalone tool.

Menu Entry	Icon	Description
New Configuration / New		Clears the configuration in the PMM main window.
Open (available only in the PMM standalone tool)		Opens the Open dialog box, enabling you to select the PMM file to open.
Import PMM		Opens an existing PMM configuration. If a configuration is already defined in the Configuration pane, PMM prompts you to Append the selected file to the existing configuration, or replace the existing configuration.
Export PMM		Saves the current configuration to a `*.PMM` file. The following information is saved: ◆ Data sources ◆ Hosts ◆ Selected measurements from each data source on each host Note: Exporting a configuration from the PMM within WebLOAD does not assign the configuration to the current template. To update the template with the configuration displayed, click Close and update template.
Save / Save As (available only in the PMM standalone tool)		Saves the current configuration to a `*.PMM` file.
Close and update template (available only in the PMM initiated through the WebLOAD Console)		Closes PMM and updates the current WebLOAD template with the current PMM configuration. Note: The PMM configuration is not saved outside of the template.

Menu Entry	Icon	Description
Exit	(none)	Closes PMM.

Actions Menu

The Actions menu includes the following functions for creating a PMM configuration file and controlling the Configuration pane.

Menu Entry	Icon	Description
Add Data Source		Open the PMM Wizard for configuring new data sources, hosts and measurements to monitor.
Delete		Deletes the currently selected item and all sub-items from the configuration.
Clear All (available only in the PMM standalone tool)		Removes all elements from the current configuration.
Expand Tree		Expands the configuration tree.
Collapse Tree		Collapses the configuration tree.

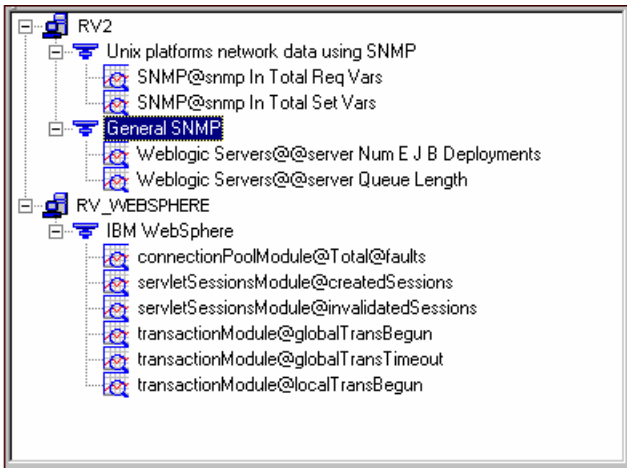
Help Menu

Use the PMM Help menu to access online help, as well as copyright and version information about PMM.

Configuration Pane

The Configuration pane displays the performance objects that are configured for monitoring, including hosts, data sources and measurements.

Configuration Pane configured with three data sources, two hosts and multiple measurements



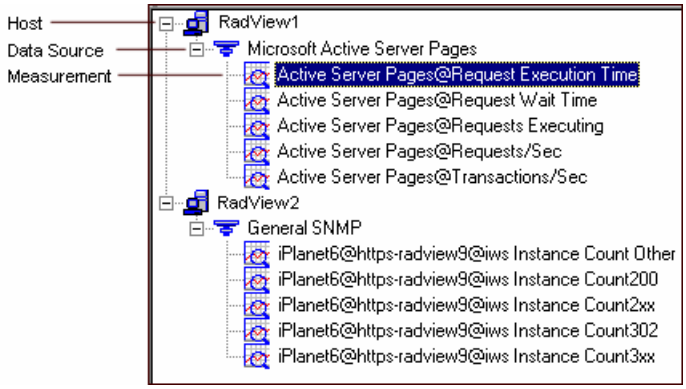
Description Pane

The Description pane displays information about the item selected in the Configuration window. As you click different items in the Configuration window, the information displayed in the description pane changes appropriately.

The PMM Configuration Tree Structure

The PMM Configuration tree gives you a complete graphical overview of the performance objects configured for monitoring including hosts, data sources and measurements. The icons adjacent to the tree items enable you to view your current configuration at a glance.

The following figure shows the PMM configuration tree, configured to collect data from two hosts, each with one data source and multiple measurements.



Configuration Tree Structure

The first level branches display the hosts for which the data sources and measurements are defined. The second level branches display the data sources configured on each host and the third level branches display the measurements configured for each data source on each host.

Configuration Tree Icons

PMM displays icons on each line of the Configuration Tree, making it easy to view the performance objects configured for collection during your test session.

The Configuration Tree icons are described in the following table:

Icon	Performance Object
	Host
	Data Source
	Measurement

The following options can be accessed through the right mouse button:

Tree Item	Right-Mouse Option	Description
Host	Add data source	Opens the PMM Wizard, enabling you to select additional data sources to monitor on the selected host.
	Delete	Removes the selected host from the configuration.

Tree Item	Right-Mouse Option	Description
Data Source	Add measurements	Opens the PMM Wizard to enabling you to select additional measurement to monitor on the selected data source.
	Delete	Removes the selected data source from the configuration.
Measurement	Delete	Removes the selected measurement from the configuration.

Navigating the PMM Configuration Tree

You can move through a PMM configuration tree by expanding and collapsing nodes as needed with the plus ⊞ and minus ⊟ sign keys, or by clicking and double-clicking with the mouse.

The following table lists the mouse actions you can use to move through a PMM configuration tree.

Action	Result
Click	Selects an item and displays a description for the selected item.
Double-click	Displays or hides items contained by the selected item.
Right-click	Displays the Actions shortcut menu for the selected item.

The following table lists the keystrokes you can use to move in the PMM main window.

Key	Result
Up arrow	Moves the selection up one item in the configuration tree.
Down arrow	Moves the selection down one item in the configuration tree.
Right arrow	Expands the selected item. If the selected item does not contain hidden items, behaves like down arrow.
Left arrow	Collapses the selected item. If the selected item doesn't contain exposed items, behaves like up arrow.
Page Up / Home	Moves the selection to the first item in the configuration tree.
Page Down	Moves the selection to the bottom item visible in the window.
End	Moves the selection to the last item in the configuration tree.
Plus sign “+” on the numeric keypad	Expands the selected node.
Minus sign “-” on the numeric keypad	Collapses the selected node.

Key	Result
Asterisk "*" on the numeric keypad	Expands the entire tree below the selected node.

Using PMM Configuration Files

PMM configuration files can be created and attached to WebLOAD templates for monitoring performance in a test session, or configured and saved for use in future WebLOAD sessions.

Creating a New Configuration File

When PMM is first opened the main configuration window is empty, enabling you to create a new configuration file. For directions on how to define the components to monitor, see Configuring the Data Sources, Hosts, and Measurements to Monitor (on page 375).

If the PMM currently displays a configuration file, creating a new configuration file requires clearing the current configuration information from the display window.

To clear an existing configuration from the configuration window in order to define a new configuration file:

1. Click the New toolbar button.

   A dialog box displays asking for confirmation.

2. Click Yes.

   The PMM configuration window is cleared.

Opening / Importing Saved PMM Configuration Files

PMM configurations can either be opened alone, or they can also be added to any existing configuration in the PMM configuration window, letting you reuse existing configuration files within a new test configuration.

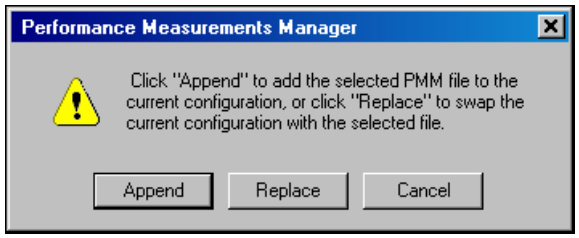
To open an existing configuration file:

1. Select File | Import PMM or click on the Import toolbar button.

   The Open dialog box opens for selecting an existing PMM configuration file.

2. Browse to the location of the PMM configuration files, select a file, and click Open.

   If a configuration is already defined and displayed within the PMM configuration window, a message box displays:



   The message box has the following options:

   - Append - Add the selected PMM configuration to the current configuration in the PMM configuration window.
   - Replace - Remove the current configuration in the PMM configuration window, and replace it with the selected configuration.

   The selected configuration is added to the PMM configuration window.

Assigning a PMM Configuration to a WebLOAD Template

PMM configurations can only be attached to WebLOAD templates when working with the PMM from within WebLOAD.

To assign a PMM configuration to a template:

1. Ensure that the template to which you want to assign the PMM configuration is open in WebLOAD.
2. Open / define the PMM configuration you want to attach to the template. For information on defining PMM configurations, see Configuring the Data Sources, Hosts, and Measurements to Monitor (on page 375).
3. Select File | Close and update template, or click the Close & update toolbar button.

   PMM closes and the configuration file is assigned to the current template.

Saving and Exporting PMM Configurations

After configuring the data sources, hosts and measurements to monitor you can save the configuration for use in WebLOAD tests. PMM lets you save multiple configurations to save you the time of reconfiguring the measurements to monitor each test. PMM configuration files are saved with the extension `*.PMM`.

PMM configuration files include the following information to monitor:

- Data sources
- Hosts
- Measurements

Exporting PMM Configuration Files

When working with PMM from within WebLOAD you create configurations for use with the current WebLOAD template. To save the configuration, you must export it to a PMM file.

To export PMM files:

1. From the PMM activated in WebLOAD, select File | Export PMM, or click the Export toolbar button.

   The Save As dialog box opens.
2. Browse to the location you want to save your PMM file, enter a name for the file in the File Name field, and click Save.

Saving PMM Configuration Files

When working with PMM as a standalone tool, you can save configurations for use in WebLOAD tests. The configuration is not attached to any WebLOAD template.

To save PMM files:

1. From the PMM standalone tool, select File | Save, or click the Save toolbar button.

   The Save As dialog box opens.
2. Browse to the location you want to save your PMM file, enter a name for the file in the File Name field, and click Save.

Configuring the Data Sources, Hosts, and Measurements to Monitor

PMM can be used for adding new data sources to the PMM configuration or updating an existing configuration. You select the type of data source to monitor and the measurements to monitor on each host.

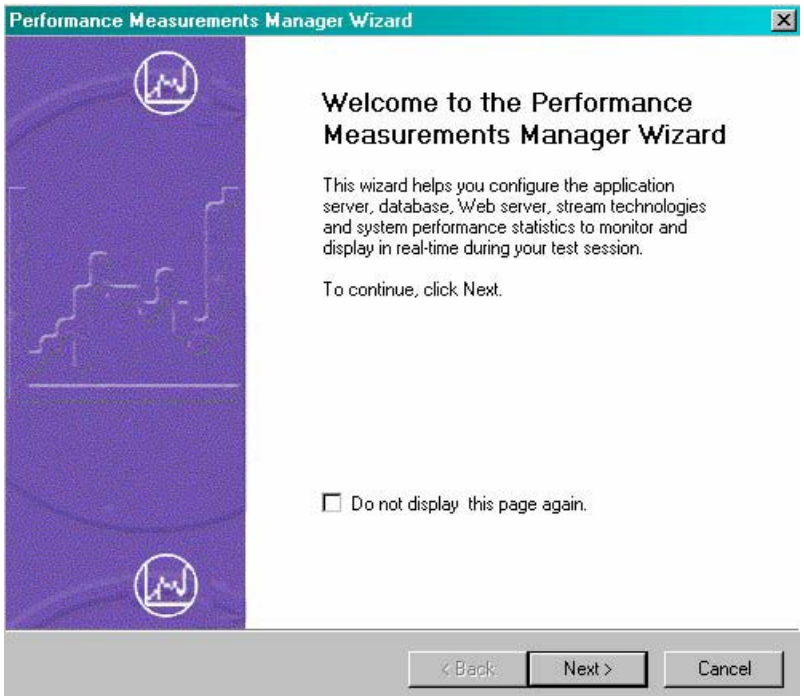
Adding Data Sources

Data sources are the performance objects you want to monitor.

To add new data sources:

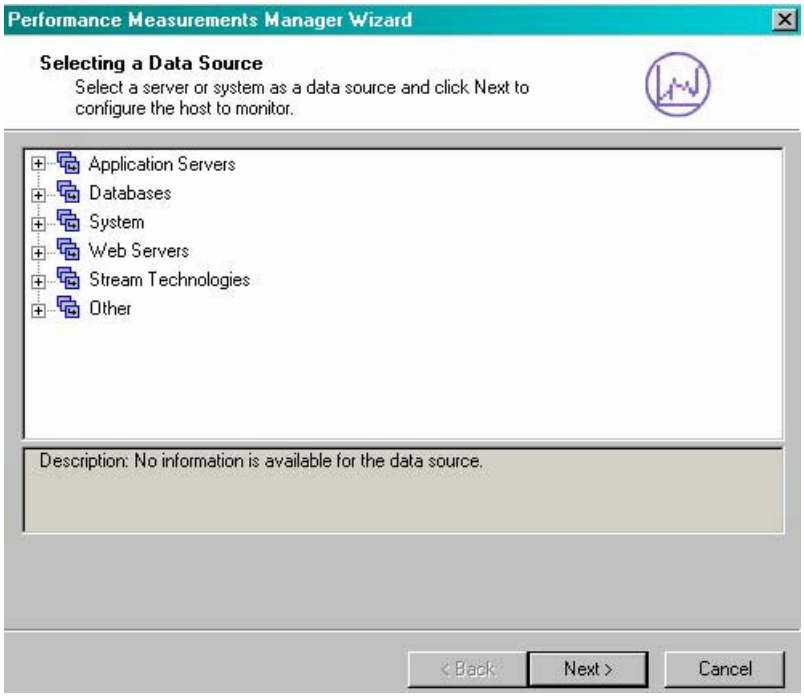
1. Click the Add data source toolbar button.

   The Performance Measurements Manager Wizard opens.



2. Click Next.

   The Data Source screen displays.



3. Expand the tree to view all available data sources. PMM supports the following types of data sources:

   Application Servers:

   - BEA WebLogic
   - IBM WebSphere 5
   - Microsoft Active Server Pages
   - Microsoft ASP .Net

   Databases:

   - Oracle
   - Microsoft SQL Server
   - Microsoft SQL Server using SNMP

   System:

   - Windows platform
   - UNIX platforms Network data using SNMP
   - UNIX platforms using RSTATD

- UNIX UC-Davis
- Windows .Net Framework

Web Servers:

- Microsoft Internet Information Server (IIS)
- Sun One (former iplanet 6)

Stream Technologies:

- Microsoft Media Server

Other:

- General Performance Monitor

  General Performance Monitor can be used just like the System > Windows Platform. The difference is that it gives access to all Performance Monitor counters.

- General SNMP

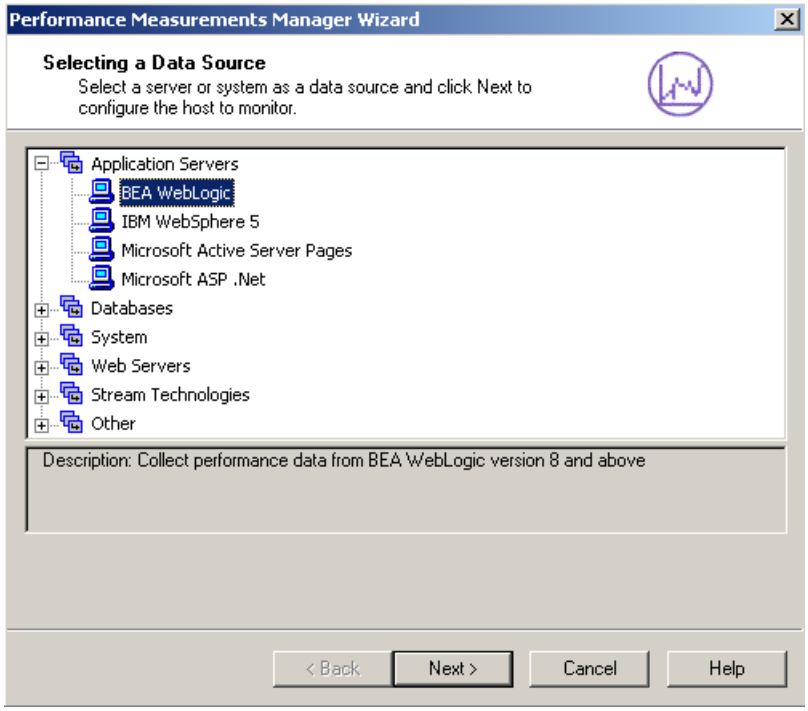
  General SNMP can be used similarly to other SNMP-based PMs. The difference is that is gives access to all MIBs served by the SNMP agent on that machine.

- General UNIX

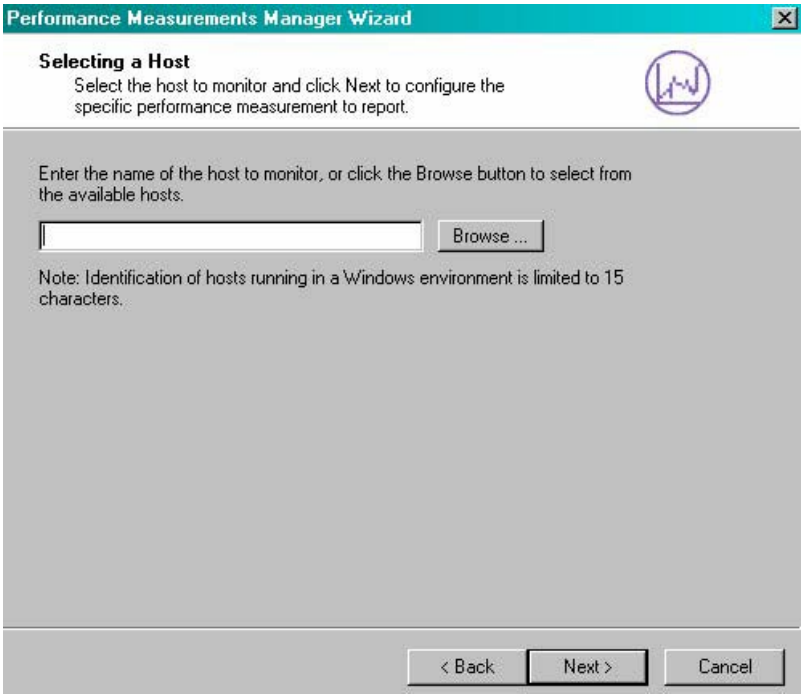
  General UNIX can be used similarly to the System > UNIX platform using RSTATD. The difference is that is gives access to all RSTATD counters.

4. Select a server or system as a data source, and click Next.

   In this example, BEA WebLogic is selected.

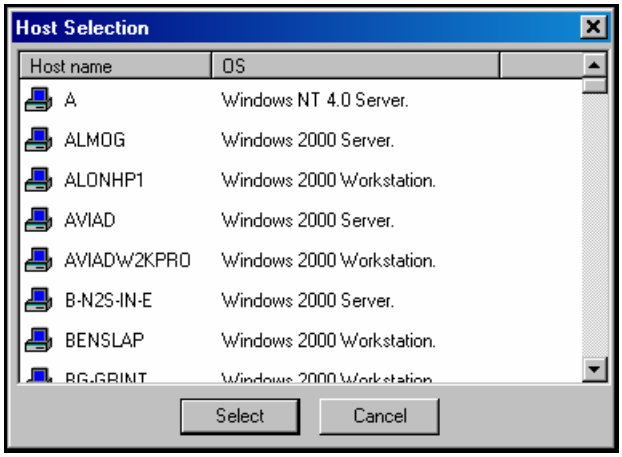


The Host Selection Screen displays:



5. Click the Browse button to select a host to monitor.

   The Host Selection dialog box opens.

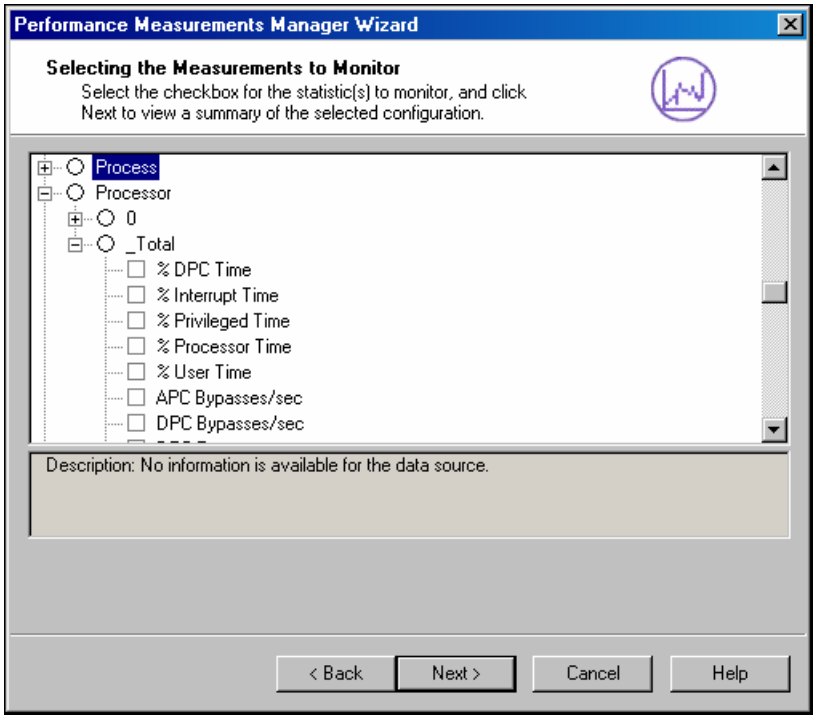


6. Choose the host to monitor from the list and click Select.

   The host is added to the Host Selection screen.

7. Press Next.

   PMM attempts to connect to the host. If a connection is made, the Wizard progresses to the Measurements to Monitor screen. If a connection cannot be established, the Wizard returns to the Host Selection screen, enabling you to select an alternate host.



8. Select one or more measurements to monitor by clicking on the checkbox adjacent to the item. PMM only collects the data for measurements that are checked. All other data is not collected.

   Explanations of each measurement are available by clicking on the measurement name. The explanation appears at the bottom of the window.

   The list of available measurements varies depending upon the type of data source. Each data source is supplied with a default set of measurements to monitor.

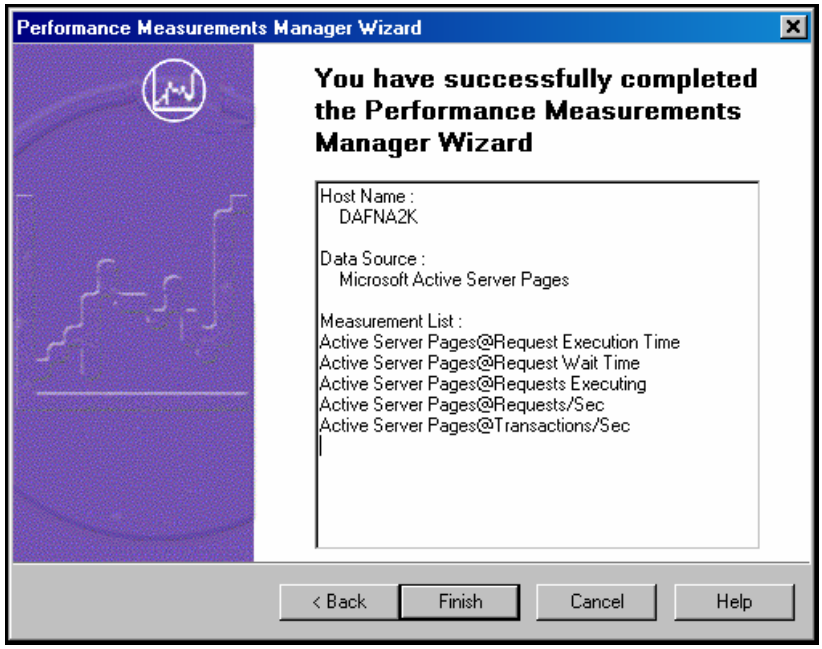
Notes:

An empty circle ○ next to a measurement indicates that the measurement contains sub-components and that no default subcomponents are defined (the upper level component cannot be selected). To select subcomponents, click on the + to expand the tree.

A circle with a checkmark ⊘ next to a measurement indicates that the measurement contains sub-components and that default subcomponents are defined. To view and configure the configured subcomponents, click on the + to expand the tree.

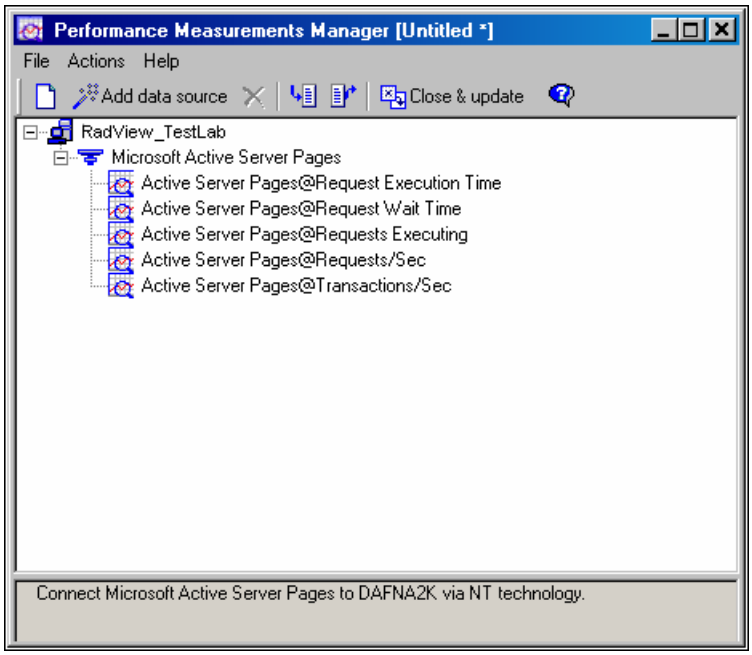
9. Click Next.

The Wizard displays a summary of the host, data source, and measurements configured for monitoring.



10. To accept the configuration, click Finish.

    The PMM Wizard closes and the selected configuration is added to the PMM main window.



11. To add additional data sources, hosts and measurements to monitor, click the Add data source toolbar button and repeat the process.

Adding Measurements to a Configured Data Source

Measurements can be added quickly and easily to a data source configured in the PMM window.

To add measurements to a data source already configured in the PMM:

1. Right-click on the data source to which you want to add measurements and click Add measurements from the pop-up menu.

   The Measurements to Monitor screen for the selected data source displays.

2. Select all of the measurements you want to monitor for the data source by selecting the checkbox adjacent to the measurement names.

   Explanations for each measurement are available by clicking on the measurement name. The explanation appears at the bottom of the window.

3. Click Next.

   A summary of the current configuration displays.

4. To accept the configuration, click Finish.

   The PMM Wizard closes and the configuration is updated in the PMM main window.

Adding Data Sources to a Configured Host

Data sources can be added quickly and easily to a host configured in the PMM window.

To add data sources to a host already configured in the PMM:

1. Right-click on the host in the PMM main window to which you want to add a data source, and click Add data source from the pop-up menu.

   The Select a Data Source Wizard screen displays.

2. Select the data-source to add to the configuration and click Next.

   The PMM Wizard progresses to the Measurements to Monitor screen.

3. Select all of the measurements you want to monitor for the data source.

   Explanations of each measurement are available by clicking on the measurement name. The explanation appears at the bottom of the window.

4. Click Next.

   A summary of the current configuration displays.

5. To accept the configuration, click Finish.

   The PMM Wizard closes and the configuration is updated in the PMM main window.

Deleting a Data Source

To delete a data source from PMM configuration:

1. Right-click on the data source you wish to delete in the PMM main window.
2. Select Delete from the pop-up menu.

   A message box appears asking for confirmation.

3. Click Yes.

   The data source is removed from the configuration and is no longer monitored.

Deleting a Monitored Host

To delete a host from the PMM configuration:

1. Right-click on the host you wish to delete in the PMM main window.
2. Select Delete from the pop-up menu.

   A message box appears asking for confirmation.
3. Click Yes.

   The host, and all data sources and measurements configured for the host are removed from the configuration and are no longer monitored.

Deleting a Measurement

To delete a measurement from the PMM configuration:

1. Right-click on the measurement you wish to delete in the PMM main window.
2. Select Delete from the pop-up menu.

   A message box appears asking for confirmation.
3. Click Yes.

   The measurement is removed from the configuration and no longer monitored.

Application Server Monitors

PMM enables you to monitor application servers during runtime to view performance data from the application server in real-time. Before running your test session, you must first configure the PMM monitors. WebLOAD supports monitoring for the following application servers:

- BEA WebLogic
- IBM WebSphere 5
- Microsoft Active Server Pages
- Microsoft ASP .Net

WebLogic Monitor Configuration Tasks

The PMM WebLogic monitor uses SNMP to retrieve server statistics.

To use this monitor you must:

- Ensure that the WebLogic SNMP agent is installed and activated on the WebLogic server host.
- Configure the WebLogic Server to be a proxy for the SNMP agent, described in Setting Up the SNMP Agent and Configuring the WebLogic Server as a Proxy for the SNMP Agent (on page 386).
- Configure the WebLogic Server host to use port 1161 as the SNMP message port, described in Configure the WebLogic Server Host to use Port 1161 (on page 386).

These configuration tasks are necessary because some UNIX/Linux machines have authentication issues when working with the default SNMP port 160.

Setting Up the SNMP Agent and Configuring the WebLogic Server as a Proxy for the SNMP Agent

To enable PMM to monitor performance data from the WebLogic application server you must first set up the WebLogic SNMP agent.

To set up the SNMP agent:

1. Follow the instructions outlined at:

   http://e-docs.bea.com/wls/docs70/////ConsoleHelp/snmp.html (http://e-docs.bea.com/wls/docs70/////consolehelp/snmp.html)

   Look under Setting up the WebLogic SNMP Agent. Be sure to enable the SNMP agent (by default the agent is off) and set the SNMP Port field to 161.

2. Configure a new SNMP proxy using the following values in the attribute fields:

Attribute	Value
Port	1161
Root	.1.3.6.1.2.1 (be sure to include the leading period)
Community	Public
Timeout	5000

   Instructions for creating the proxy are available at:

   http://e-docs.bea.com/wls/docs70/////ConsoleHelp/snmp.html (http://e-docs.bea.com/wls/docs70/////consolehelp/snmp.html)

   Look under Create an SNMP Proxy.

3. Restart the WebLogic Administration Server so that the changes take effect.

Configure the WebLogic Server Host to use Port 1161

The WebLogic Server host must be configured to use port 1161 as the SNMP message port.

To configure the WebLOGIC Server Host to use Port 1161:

- On the WebLogic Server host, update the SNMP port number in the system file:

  … `system32\driver\etc\services`

  Change the number from 161 to 1161.

IBM WebSphere Configuration Tasks

To monitor the IBM WebSphere application server, you must first:

1. Install the IBM J2EE Application Client provided with WebSphere on the WebLOAD Console machine.
2. Enable data collection as described in: http://www.ibm.com/software/webservers/appserv/doc/v35/ae/infocenter/was/atsepmcf.html (http://www.ibm.com/software/webservers/appserv/doc/v35/ae/infocenter/was/atsepmcf.html)

Database Monitors

PMM enables you to monitor database servers during runtime. This allows you to view performance data from the database in real-time. Before running your test session, you must first configure the PMM monitors.

WebLOAD supports monitoring for the following databases:

- Oracle
- Microsoft SQL Server
- Microsoft SQL Server using SNMP

Oracle Monitor Configuration Tasks on Windows

The PMM Oracle database monitor uses SNMP to retrieve statistics.

To use this monitor on Windows you must:

1. Install SNMP support for Windows.
2. Install SNMP support for Oracle using the Oracle Database installation.
3. Specify the port where the Oracle master agent is listening.

   The required port is specified in the TRANSPORT section of the MASTER.CFG file located in the \ADMIN directory.

   For example, add the following section to the MASTER.CFG file:

```
TRANSPORT ordinary SNMP
OVER UDP SOCKET
AT PORT 161
```

Use port 161 for the Oracle Peer SNMP master agent.

4. Specify the authentication in the COMMUNITY section of the MASTER.CFG file.

   For example:

```
COMMUNITY public
ALLOW ALL OPERATIONS
USE NO ENCRYPTION
```

5. Specify an unused port where the encapsulated agent, such as Microsoft SNMP service, should listen. Microsoft SNMP service typically uses port 1161. The port is specified in the SERVICES file located in the `NT_HOME|\SYSTEM32\DRIVERS\ETC` directory.

   For example, modify the `nt_home\system32\drivers\etc\services` file from:

```
…
snmp 161/udp
…
```

   To:

```
snmp 1161/udp
```

6. Specify which non-`PEER` master agents are to be encapsulated by editing the encapsulator configuration file, `ENCAPS.CFG`, located in the `ADMIN` directory. You must add the `AGENT` entry, including MIB-subtrees manageable by NMS, for the encapsulated master agent.

   For example, modify the `encaps.cfg` file as follows:

```
AGENT AT PORT 1161 WITH COMMUNITY public
SUBTREES 1.3.6.1.2.1.1,
1.3.6.1.2.1.2,
1.3.6.1.2.1.3,
1.3.6.1.2.1.4,
1.3.6.1.2.1.5,
1.3.6.1.2.1.6,
1.3.6.1.2.1.7,
1.3.6.1.2.1.8,
1.3.6.1.4.1.77
```

```
FORWARD ALL TRAPS;
```

Note: The port (1161) must match the unused port specified in step 5.

7. Make the "OracleSNMPPeerEncapsulator" service automatic and start it.
8. Reboot.

Oracle Monitor Configuration Tasks on UNIX

The PMM Oracle database monitor uses an SNMP Master Agent to communicate with the Oracle Intelligent Agent. The SNMP Master Agent and the Oracle Intelligent Agent must be configured correctly before the Oracle Intelligent Agent can communicate over SNMP to the Master Agent on a UNIX platform. For detailed information on configuring the Intelligent Agent, see the *Oracle Enterprise Manager Configuration Guide*.

To use this monitor on UNIX you must:

1. Configure the SNMP Master Agent.
2. Start the SNMP Agents.

Configuring the SNMP Master Agent

The necessary SNMP files are installed automatically when you install the Oracle Intelligent Agent. After installing the Oracle Intelligent Agent, you will have to edit the following files as described below:

- `$ORACLE_HOME/network/snmp/peer/CONFIG.master`
- `$ORACLE_HOME/network/snmp/peer/CONFIG.encap`
- `$ORACLE_HOME/network/snmp/peer/start_peer`
- `/etc/snmpd.config`

CONFIG.Master (Peer Master Agent file)

To edit the CONFIG.master file:

- Find the line beginning with `MANAGER` and change the IP address coded in this line to match the IP address of the machine where the SNMP traps will be sent.

CONFIG.encap (Peer Encapsulator file)

To edit the CONFIG.encap file:

- Find the line `AGENT AT PORT`. It normally reads `AGENT AT PORT 1161 WITH` COMMUNITY `public`.

  If you modify the port number from 1161, you must also modify the `start_peer` script.

start_peer Script

To edit the start_peer script:

1. Find the line `NEW_SNMPD_PORT=` and verify that it is using the same port number listed above in the CONFIG.encap file.
2. Find the line `NEW_TRAPD_PORT=` and verify the port number is different from `NEW_SNMPD_PORT=.`

snmpd.config File

To edit the snmpd.config file:

1. Add the following line to the file:

   `trap <`*`hostname`* `or` *`ipaddress`*`>`
2. Replace the information in angle brackets with the actual hostname or IP address of the local host where the file is located.

Starting the SNMP Agents

To start the SNMP agents:

1. Make sure no SNMP components are running. The three main components are the master_peer, encap_peer, and snmpd. If any of these processes are running, use the `ps` command to find them, and the `kill` command to terminate the processes.

   For example:

   ```
   ps -ef | grep snmp
   ```

   This command checks to see if the SNMP Master Agent is running.
2. Start the PEER Master Agent, PEER encapsulator, and native Digital UNIX SNMP Agent:

   ```
   cd $ORACLE_HOME/network/snmp/peer
   ```

```
su root
./start_peer -a
```

This command starts all three processes.

3. Now use the `ps` command to verify that all three processes were started:

```
ps -aux |grep peer
ps -aux |grep snmpd
ps -ef | grep snmp
```

Web Server Monitors

PMM enables you to monitor Web servers during runtime to view performance data from the server in real-time. Before running your test session, you must first configure the PMM monitors.

WebLOAD supports monitoring for the following Web servers:

- Microsoft Internet Information Server (IIS)
- Sun One (former iplanet 6)

Sun One (iPlanet) Configuration Tasks

The PMM Sun One monitor uses SNMP to retrieve server statistics.

To use this monitor you must:

1. Install SNMP support for Windows.
2. Ensure that the iPlanet SNMP agent is installed and activated on the iPlanet server host.

   Configure the Sun One Server to be a proxy for the SNMP agent.
3. Configure the Sun One Server host to use port 1161 as the SNMP message port.
4. Enable SNMP Statistics Collection on the SNMP Subagent Configuration page of the Server Manager.

Setting up the SNMP Agent and Configuring the Sun One Server as a Proxy for the SNMP Agent

To enable PMM to monitor performance data from the Sun One application server you must first set up the Sun One SNMP agent. The Administration Interface will not start the OS master agent on ports other than 161. The workaround is to use the Sun One SNMP master agent, modify the native master agent to use a different port, and enable the Sun One SNMP subagent to gather information about the server.

To set up the Sun One SNMP agent on Solaris:

1. Check whether an SNMP daemon (snmpd) is running on port 161.

   If an SNMP daemon is running:

   a. Make sure you know how to restart it and which MIB it supports.

   b. Kill its process.

2. Edit the `CONFIG file located in the plugins/snmp/magt server root directory` from:

   …

   ```
   snmp 161/udp
   ```

   …

   To:

   …

   ```
   snmp 1161/udp
   ```

   …

3. Manually activate the Sun One master agent and subagent.

   a. At the command prompt, enter the commands:

   ```
   cd <iws_install>/Servers/plugins/snmp
   cd maget
   ./magt CONFIG INIT&
   cd ../sagt
   ./sagt -c CONFIG &
   ```

   b. Restart the Sun One Server and the subagent from the Web server Administration.

Monitoring Generic SNMP Data Sources

WebLOAD currently supports a variety of SNMP monitors by default, including basic SNMP for Windows, Solaris, and UNIX, along with WebLogic, and Oracle servers. WebLOAD's SNMP monitoring support can be easily extended by updating WebLOAD's UserMibFilter.ini file.

Understanding MIB Files

To modify the WebLOAD UserMibFilter.ini file, an understanding of your vendor's SNMP and MIB structure is required. For a full overview of SNMP and MIB structure, there are many great resources on the Web, and you will likely find your SNMP vendor's documentation there.

SNMP relies on MIB (Management Information Base) files. These files are always readable, and usually have a file extension of either .mib or .txt, though others may have proprietary file extensions. These files provide a structure to the information that an SNMP agent can provide to a program that monitors SNMP. There are publicly available specifications for SNMP MIBs, and there are privately provided MIBs which are specific to a particular product.

All of the MIBs, both public and private, are organized into a tree structure that helps to identify them. Each location in the tree is specified by both a name and a numeric value. The MIB tree has ISO at its root, and then four main branches: directory, mgmt, experimental, and private. The private branch is where enterprise level MIBs are placed, and their location in the tree can be specified both by writing out the tree structure with points separating the names, and also numerically, using the numbers associated with each branch in the tree.

For example, all private SNMP MIBs will have the following structure:

- ISO has the value of 1
- Internet has the value of .3.6.1
- Private has the value of 4
- Enterprise has the value of 1.

So any enterprise level MIB will have an address that starts with:

```
.iso.org.dod.internet.private.enterprises
```

The numeric equivalent is:

```
.1.3.6.1.4.1
```

Each enterprise which creates and supports its own SNMP MIBs, such as UC Davis, WebLogic, or others will have an assigned place within the tree which provides it's name and numeric name. Each MIB also has associated attributes, which describe the SNMP data that is returned. There is often documentation of each of these MIBs and the base names and numeric numbers associated with each in the products being used. Another useful way to get information on installed MIBs is to use a MIB browser.

The WebLOAD UserMibFilter.ini File Structure

Following is an extract for Oracle databases from the `UserMibFilter.ini` file.

```
[Oracle Databases]
base=.iso.org.dod.internet.private.enterprises.oracle.oraDbMIB.oraDbObjects.oraDbSysTable.oraDbSysEntry
numbase=.1.3.6.1.4.1.111.4.1.1.1
instance=.oraDbSysTable.oraDbSysEntry
instdesc=Database
prefix=ora
file=1
```

```
attr_range=1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21
,22,23,24

attr_names=oraDbSysConsistentChanges,oraDbSysConsistentGets,oraD
bSysDbBlockChanges,oraDbSysDbBlockGets,oraDbSysFreeBufferInspect
ed,oraDbSysFreeBufferRequested,oraDbSysParseCount,oraDbSysPhysRe
ads,oraDbSysPhysWrites,oraDbSysRedoEntries,
oraDbSysRedoLogSpaceRequests, oraDbSysRedoSyncWrites,
oraDbSysSortsDisk,oraDbSysSortsMemory,oraDbSysSortsRows,
oraDbSysTableFetchRowid,oraDbSysTableFetchContinuedRow,
oraDbSysTableScanBlocks,oraDbSysTableScanRows,
oraDbSysTableScansLong,oraDbSysTableScansShort,oraDbSysUserCalls
,oraDbSysUserCommits,oraDbSysUserRollbacks,oraDbSysWriteRequests
```

The structure of the `UserMibFilter.ini` file is as follows:

Parameter	Description
[…]	A comment that describes the MIB information that follows. This comment is displayed by the Performance Measurements Manager (PMM) when a host running the SNMP agent is added to a test.
base	The name derived from the MIB tree. The first 6 elements describe the common MIB identifier. The seventh element describes the object, in this example Oracle, and the remaining elements describe the elements of the private object. For more information, see Understanding MIB Files (on page 392).
numbase	The numeric names, which correspond to the MIB tree location, listed in the base. For more information, see Understanding MIB Files (on page 392).
instance	Specifies a table or set of tables within the MIB file that contain the attributes to monitor.
instdesc	Provides a way for WebLOAD to display some of the MIB data in the PMM window. This parameter is often either set to the generic term `instance` or simply left blank.
prefix	The prefix is provided to give the option of removing things from the names provided by the attr_names parameter.
file	Represents the number corresponding to the name of the MIB file from which the SNMP data comes. This file is listed at the top of the UserMibFilter.ini file.
attr_range	The attributes or leaves of the MIB for which WebLOAD reports statistics.
attr_names	The attributes or leaves of the MIB for which WebLOAD reports statistics.

Adding SNMP Agents

Once you understand the MIB file containing the SNMP data desired, you can update the `UserMibFilter.ini`, enabling WebLOAD to monitor data.

To add SNMP agents:

1. Shut down WebLOAD and TestTalk.
2. Locate and open the WebLOAD `UserMibFilter.ini` file.

   The file resides in the \bin directory of your WebLOAD installation.
3. Enter an index number and the name of the MIB file for which you are adding a definition, at the top of the `UserMibFilter.ini` file under the [MibFiles] heading.
4. Determine the base, numbase, instance, and attributes to monitor by looking at the vendor MIB file, or using a MIB browser.

   a. Enter the base, numbase, and instance, determined from MIB file, to the WebLOAD `UserMibFilter.ini` file.

   b. Using a MIB browser, navigate through the MIB tree to the attributes you want WebLOAD to report on.

   c. Add the attributes to the attr_names in the WebLOAD `UserMibFilter.ini` file.

   d. Add the corresponding number (which comes directly from the order in which the attributes appear in the MIB file) to the attr_range parameter.
5. Save and close the updated `UserMibFilter.ini` file.
6. Restart WebLOAD and TestTalk.
7. Use Performance Measurements Manager to add the host providing the SNMP data to your PMM configuration.

Note: WebLOAD monitors SNMP statistics that are numeric in nature. In the MIB file of the attributes you want to monitor, locate the attribute by name and determine if it is defined as numeric or string. If the attribute is a string attribute (such as a DisplayString) WebLOAD cannot monitor it.

Example

The following example demonstrates adding information to the WebLOAD UserMibFilter.ini file to run the UC Davis implementation of SNMP:

```
[UC Davis SNMP]
base=.iso.org.dod.internet.private.enterprises.ucdavis.dskTable.
dskEntry
numbase= .1.3.6.1.4.1.2021.9.1
instance= .serverTable.serverEntry
instdesc= instance
prefix=
file= 11
```

```
attr_range=1,2,3,4,5,6,7,8,9,10

attr_names=dskIndex,dskPath,dskDevice,dskMinimum,dskMinPercent,d
skTotal,dskAvail,dskUsed,dskPercent,dskPercentNode
```

A P P E N D I X A

WebLOAD Error Messages

This section describes the errors and log messages that can be generated by WebLOAD. These log messages include messages generated by the JavaScript compiler, the Console, and any human errors or user messages detected by WebLOAD at run-time. Log messages can also be added to the Agenda by the user. The log messages are displayed in the Log Window.

In This Appendix

Message Severity 397
WebLOAD Error Messages 398

Message Severity

Messages fall into four categories:

- User Note
- Warning
- Error
- Severe

USER NOTE

User Notes are messages added to the Agenda by the user. User Notes can be used in order to facilitate debugging. They do not stop WebLOAD from running.

WARNING

Warning errors do not stop WebLOAD from running. However, they do indicate suspicious conditions. The source of the problem may be generated by the ABT, the connection to the server, networking etc.

ERROR

Errors do not stop WebLOAD from running. However, they do stop the round and restart the Agenda for the Virtual Client that encountered the error.

SEVERE

Severe errors cause the Load Generator, on which the error was generated to stop immediately. A dialog box will be displayed.

WebLOAD Error Messages

Connection errors are usually due to an error in the connection definition. For example, an incorrect name or address specification, or an incorrect proxy designation can cause an error. Most such specification errors are the result of errors in data entry. Verify your configuration settings and addresses with your system administrator. Careful attention to detail will prevent most errors from ever occurring.

Errors are listed in alphabetical order.

Error	Explanation	Severity
Analysis error	Data analysis error.	Error
Cannot resolve DNS entry	DNS address not found. Verify the URL in the JavaScript file.	Severe
Conditional GET error	Document has not been modified, as expected by a conditional GET statement	Severe
Conflict error	Request could not be completed due to resource conflict. Check what other programs are currently running on your system.	Severe
Connection failed	Connection failed for any reason other than the reasons specified in this table.	Severe
Connection refused	Connection refused by the server. Unable to access the server.	Warning

Error	Explanation	Severity
Connection timeout	Connection has timed out. Reason for failure not specified. If this connection is essential for WebLOAD to continue, this error will be considered Severe and WebLOAD will stop. If the rest of the test Session is able to continue, this error will be a Warning error only.	Error /Severe
Data retrieval error	Data retrieval error. Cannot retrieve data from the server.	Severe
Delete Cookie error-Parameters	Parameters missing from DeleteCookie function.	Error
Delete Cookie error-not found	Error deleting cookie. Cookie not found.	Error
Error while reading	Error reading data from the server.	Error
Forbidden, reason not specified	The specified URL address or server request is forbidden.	Severe
Gateway error	Invalid gateway response to WebLOAD request.	Severe
Gateway timeout	Gateway did not respond within specified timeout period. Reason for failure not specified.	Severe
Illegal command	Illegal JavaScript command.	Severe
Illegal request	Illegal server request.	Severe
Internal server error	The server went down.	Severe
Length required	More length is required for the current request than is presently available.	Severe
Method not allowed	Specified access or request method is not allowed.	Severe
No buffer space	No buffer space available. Unable to access the server.	Error
Not Acceptable	Specified configuration, object description, parameters, authentication fields, or other submitted information is not acceptable.	Severe
Not implemented	Selected feature is not currently implemented	Severe
Other	Failure due to any other reason not specified on this list.	Severe

Error	Explanation	Severity
Precondition failed	The current request failed due to failure in a necessary precondition to this request.	Severe
Proxy authentication required	Access to the requested proxy must be authenticated.	Severe
Proxy required	You must use a proxy for the current test Session.	Severe
Redirection error	New location not found.	Severe
Request entity too large	Requested server object is too large to transfer successfully.	Severe
Request timeout	Request was not completed within specified timeout period. Reason for failure not specified. If this request is essential for WebLOAD to continue, this error will be considered Severe and WebLOAD will stop. If the rest of the test Session is able to continue, this error will be a Warning error only.	Error/ Severe
Request URI too long	URI of requested object is too long to handle.	Severe
Resource not available	The requested resource is not currently available from the server.	Severe
Send failed	Request was not sent successfully to the server.	Error
Service temporarily unavailable	The specified service is temporarily not available.	Severe
SetCookie error-Parameters	Parameters missing from SetCookie function.	Error
SetCookie error	SetCookie failure for unspecified reason.	Error
Socket failed.	You are unable to reach this URL.	Severe
	Stop the test Session, and restart.	
Socket not bound	Socket not bound to an address.	Error
	Unable to access to the server.	
Unauthorized request	Request requires authorization and user authentication. For example, by specifying a username and password.	Severe

Error	Explanation	Severity
Unsupported media type	The specified media format is not supported by WebLOAD at this time (for example, rewriteable CD's).	Severe
Unsupported protocol	The specified protocol is not supported by WebLOAD at this time (for example, fttp://…).	Severe
Unsupported version of HTTP	The specified version of HTTP is not supported by WebLOAD at this time.	Severe
URL not found	URL address as specified is not found. Verify the URL in the JavaScript file.	Severe
URL temporarily not correct	Requested resource has been temporarily moved to a different URL.	Severe

A P P E N D I X B

Reference Materials

In This Appendix

WebLOAD Files 403
WebLOAD Hot Key Commands 404

WebLOAD Files

The following is an explanation of the files associated with a WebLOAD Load Session.

WebLOAD Extension	WebLOAD File Type
WebLOAD.INI	Configuration variables
MONITOR.INI	INI settings for the Console
HOSTS.LST	List of local hosts sites read from your system's LOCAL.HOSTS file
.JS FILES	Agenda files for all Clients, Virtual and Probing
.TPL FILES	Load Template files
.LS FILES	Load Session files
.DAT FILES	Load Session data files
.ISD FILES	Index statistics database files

WebLOAD Extension	WebLOAD File Type
.SDB FILES	Statistics database files
.MIX FILES	Agenda Mix files (Cruise Control)
.MDB FILES	Data Drilling files
WebLOAD.TAR	UNIX installation files
.EXR Files	Expected Repository
.FTS Files	Session Files
.FTA Files	Actual Repository Files
.FTE Files	Expected Repository Files
.LOG Files	Saved Log Window Files
.PFT Files	Project Files
.WLP Files	WebLOAD IDE Project Files
.WLS Files	WebLOAD IDE Session Files
.WLA Files	WebLOAD IDE Actual Repository Files
.WLE Files	WebLOAD IDE Expected Repository Files

WebLOAD Hot Key Commands

WebLOAD offers the following hot key commands:

Key Combination	WebLOAD Function
Alt + F	Open the File menu
Alt + F + N	Open the New dialog box
Alt + F + O	Open a Load Session (*.LS file)
Alt + F + C	Open the Cruise Control Wizard
Alt + F + W	Open the WebLOAD Wizard
Alt + F + M	Open a Load Template (*.TPL file)
Alt + F + A	Save As a Load Session (*.LS file)
Alt + F + X	Exit WebLOAD
Alt + S	Open the Session Control menu
Alt + S + V	Open the Server Monitor View Definition dialog box
Alt + S + D	Delete the Load Template
Alt + R + I + N	Open the New Report dialog box

Key Combination	WebLOAD Function
Alt + R + I + D	Open the Delete Report dialog box
Alt + R + I + O	Open the Open Report dialog box
Alt + R + C	Open the Statistics Report
Alt + R + D	Open the Data Drilling Transaction Grid
Alt + V + T	Open the Toolbars dialog box
Alt + V + G	Toggle the Gauges on/off
Alt + V + E	Toggle the Session Tree window on/off
Alt + V + L	Toggle the Log window on/off
Alt + V + W	Clear the Log window
Alt + T + C	Open the Customize dialog box
Alt + W + C	Cascade the windows open in the WebLOAD Results window
Alt + W + T	Horizontally tile the windows open in the WebLOAD Results window
Alt + W + V	Vertically tile the windows open in the WebLOAD Results window
Alt + H + C	Open the WebLOAD Help Contents tab
Alt + H + J	Open the JavaScript Programming Help file
Alt + H + L	Open the WebLOAD License Information dialog box
Alt + H + W	Open the About Console dialog box
Ctrl + P	Open the Print dialog box
Ctrl + O	Open a Load Session (*.LS file)
Ctrl + N	Open a New Load Session
F1	Open the WebLOAD Help file
F5	Save the current Load Template

I N D E X

Index

▪

.dat • 129
.fta • 403
.fte • 403
.fts • 403
.log • 271, 403
.ls • 129
    open • 132
    save • 129
.mix • 139
    save • 171
.pft • 403
.tpl • 125, 276
    open • 127

A

About Agenda Options • 222, 224
About the Goal Definition Dialog Box • 175
About the Performance Measurements Manager • 366
About WebRM • 122
Acknowledgments from a Ping • 67
Actions menu
    PMM • 369
Actions Menu • 369
Activating Throttle Control • 286
ActiveX
    counters • 320
    timers • 317
Actual Repository Files
    .fta • 403
Adding a User-Agent • 233
Adding a User-Defined Timer • 303
Adding an Agenda to a Mix • 170
Adding Data Sources • 376
Adding Data Sources to a Configured Host • 384
Adding Host Computers • 142
Adding Host Computers to the Host Computers List • 173
Adding Measurements to a Configured Data Source • 383
Adding Performance Goals • 177
Adding SNMP Agents • 394
additional information
    application being tested • 128, 130
    created by • 128, 130
    custom • 128, 130
    load session • 130
    load template • 127
    load template fields • 128
    results description • 130
    session comments • 130
    set • 128, 130
    subject • 128, 130
    template comments • 128, 130
    test description • 128, 130
    title • 128, 130
After Completing the WebLOAD Wizard • 159
Agenda • 136
    add to mix • 170
    assign equal weight in mix • 170
    copy • 72
    create • 36
    create new • 36
    current session options • 222
    default options • 222
    delete from mix • 140, 170
    icon • 82
    mix • 165, 191
    options • 221, 222
    record • 36
    run a mix of Agendas • 167
    run in cruise control wizard • 167
    run mix • 192

run test with single • 191
select • 190, 191
select in cruise control wizard • 165
select in WebLOAD wizard • 136, 137
select mix • 192
share • 72
shared disk • 72
single • 165, 191
syntax check • 244
verify • 260
weight in mix • 140
Agenda options • 221, 222
ASM diagnostic messages • 245
authentication tab • 240
automatic data collection • 248
automatic state management diagnostic messages • 245
browser emulation tab • 231
browser type • 232
client type tab • 225
connection speed • 242
connection tab • 242
current session • 222
default • 222
diagnostic tab • 243
DNS cache • 236
functional testing tab • 247
HTTP version • 237
keep-alive • 236
load generator thread properties • 235
log report diagnostics • 246
modify • 224
multithread virtual clients • 235
page time verifications • 249
pass / fail definition tab • 252
persistent connection • 236
probing client thread properties • 235
RadView support diagnostic • 247
redirection limits • 235
sleep time control tab • 227
SSL cache • 236
syntax check • 244
user agent • 232
verification • 251
web service tab • 258
Agenda verification
compare forms • 260
compare images • 260
compare links • 260
compare number of frames • 260
compare tables • 260
compare title • 260
report default values • 260
run all tests • 260
text search • 260
Analyzing Test Results • 36, 45
append
configuration files • 373
PMM configuration files • 373
application being tested
additional information • 128, 130
application server
resource monitor • 365
application server monitors • 385
Application Server Monitors • 385
application servers
BEA WebLogic • 376
IBM WebSphere • 376
Microsoft Active Server Pages • 376
Microsoft ASP .Net • 376
MySAP.com • 376
ASM diagnostic messages
enable • 245
Assigning a PMM Configuration to a WebLOAD Template • 374
Assigning Equal Weight to all Agendas in the Mix • 140, 170
attempted connections
definition • 320
authentication
set • 240
web services • 258

authentication for protocol scripts
Agenda options • 240
session options • 240
authentication for protocol scripts tab • 240
authentication settings
configure • 240
automatic data collection
Agenda option • 248
counters for • 320
enable • 248
HTTP level • 248
object level • 248
pages • 248
session option • 248
timers for • 317
Automatic State Management diagnostic messages
enable • 245
automatic transactions
data drilling • 248
autosave file name
setting • 205
average • 323

B

BEA WebLogic
configuration tasks • 385
configure • 385
configure hosts to port 1161 • 386
data source • 376
set as proxy for SNMP agent • 386
Browser Client • 26, 27
browser emulation
add user agent • 233
Agenda options • 231
delete user agent • 234
enable keep-alive • 236
enable persistent connection • 236
options • 231
session options • 231
set browser type • 232
set HTTP version • 237
set load generator thread properties • 235
set probing client thread properties • 235
set redirection limits • 235
set user agent • 232
simulate DNS cache • 236
simulate SSL cache • 236
browser emulation settings
configure • 231
browser type
set • 232

C

cache
clear • 36
case inspections
syntax check • 244
Changing the Length of the Reporting Interval • 314
Changing the Reports Parameters • 302
Changing the Scale of a Report for a Selected Measurement • 308
client
web service • 258
Client Type Feature Summary • 26, 28
Client Type Settings • 224, 225
client type tab
Agenda options • 225
full client • 225
HTTP protocol mode • 225
session options • 225
command line
example • 278
launch WebLOAD • 277
parameters • 277
syntax • 278
Commands Dialog Box • 104
communication
between Hosts • 62
Communication Between Host Computers • 62
communications
errors • 278
establishing • 278

ping a host • 67
required connections • 67
test connections • 67
troubleshooting • 68
Completing the WebLOAD Wizard • 146, 149, 158
Completing Your Manual Load Template Configuration • 198
components • 79
connected workstation • 23
context menus • 80
menu bar • 80
results window • 89
session tree • 81
standalone workstation • 23
status bar • 81
title bar • 79
toolbars • 80
CONFIG.encap (Peer Encapsulator file) • 390
CONFIG.Master (Peer Master Agent file) • 389
configuration • 69
ABT • 69
Console • 69
example • 71
load machine • 69
probing client machine • 69
Configuration • 69
configuration files
append • 373
attach to WebLOAD templates • 373, 374
create new • 373
export • 374, 375
import • 373
new • 373
open • 373
replace • 373
save • 373, 374, 375
use • 373
configuration pane
PMM • 370
Configuration Pane • 370
configuration tree
PMM • 370
Configuration Tree Icons • 371
configuration tree structure
PMM • 370
Configuration Tree Structure • 371
Configure the WebLogic Server Host to use Port 1161 • 386
Configuring a Goal - An Example • 179
Configuring a Load Template Manually • 112, 114, 189
Configuring a Load Template using the WebLOAD Wizard • 36, 40
Configuring an Integrated Report • 300, 305
Configuring an Integrated Report to Include Data from Previously Saved Load Sessions • 305
Configuring Browser Parameter Settings • 224, 231
Configuring Emulation Settings • 224, 256
Configuring Reset Mechanism Settings • 224, 254
Configuring the Data Sources, Hosts, and Measurements to Monitor • 373, 374, 375
Configuring the SNMP Master Agent • 389
connect time
definition • 317
connected workstation • 121
WebLOAD component • 23
Connecting to the Floating License Server • 58
Connecting to WebRM • 70, 112, 118, 121
connection
attempted • 320
connection speed
Agenda options • 242
configure • 242
definition • 315
session options • 242
set • 242
set for load generator • 242
set for probing client • 242
Connection Speed Settings • 224, 242
connection speed tab • 242
console

components • 79
configuration dialog boxes • 190
context menues • 80
dashboard • 92
data drilling • 93
definition • 23
edit mode • 79, 96
features • 77
file menu • 98
help menu • 105
load session diagram • 23
log window • 94
menu bar • 80
menus • 98
modes • 79
open • 78, 117
reports menu • 100
results window • 89
run mode • 79, 97
session control menu • 99
session mode • 281
session tree • 81
shortcut screen • 95
status bar • 81
title bar • 79
toolbar • 105
toolbars • 80
tools menu • 104
view menu • 101
windows menu • 105
Console Elements • 79
context menus • 80
Context Menus • 80
Control Buttons • 177
cookies
clear • 36
Correcting the All Hosts are Unreachable or Stopped Error • 279
Correcting the Load Session Setup Not Completed Error • 279
Correcting the Load Session Setup Timed Out Error • 280
count • 323
Counter Definitions • 320
counter statistics
current value calculation • 315
defined • 313
list of • 320
counters
for Java and ActiveX • 320
for user-defined transactions • 320
user-defined • 320
Create
Agenda • 36
new Agenda • 36
created by
additional information • 128, 130
Creating a Load Schedule Manually • 196
Creating a Load Schedule with the Load Profiler • 197
Creating a Mix • 139, 169
Creating a New Configuration File • 373
Creating a New Load Template • 126
Creating a New Mix • 140, 171
Creating a Summary Report • 359
Creating an Agenda • 36
Creating an Equal Load Schedule Manually • 146
Creating an Equal Load Schedule with the Load Profiler • 147
Creating Integrated Reports • 46
Creating Load Templates • 111
Creating Load Templates with an Agenda Mix • 192
Creating Load Templates with Cruise Control • 112, 114, 118, 126, 161
Creating Load Templates with One or More Single Agendas • 191
Creating Load Templates with the WebLOAD Wizard • 41, 112, 114, 118, 126, 133
Creating the Test Agendas • 116
cruise control
acceleration point measurements set • 206
goals reached delay set • 206
cruise control wizard • 95, 161

acceleration rate • 182
acceleration rate dialog box • 182
add Agenda to mix • 170
add goal • 177
Agenda selection • 166
default report • 186
define a mix • 169
define a performance goal • 175
define acceleration rate • 182
delete Agenda from mix • 170
delete goal • 180
example of configuring a goal • 179
finish dialog box • 184
goal definition • 175
goal definition dialog box • 175
goal definition example • 179
goal definition measurement • 176
goal definition source • 176
goal table • 176
goals achieved dialog box • 183
goals definition control buttons • 177
goals definition operator • 176
goals definition report • 176
goals definition value • 176
how it works • 162
mix • 165
mix Agenda workflow • 167
new mix • 171
open • 163
operator • 177
performance measurements manager • 166
PMM • 166
run a mix of Agendas • 167
save a mix • 171
select an Agenda • 165, 167
select host computers • 172
set action after reaching goal • 183
set Agenda options • 166
set and use goal defaults • 180
set defaults • 180, 181
set performance monitor • 167
single Agenda • 165
view Agenda • 166
view test results • 186
welcome dialog box • 163
current project options • 260
functional testing • 260
current session
Agenda options • 222
current slice • 323
current value • 323
calculation • 315
defined • 313
Current Value Computation • 315
custom
additional information • 128, 130
Customizing the Export File • 350, 351, 353, 355, 356, 357
Customizing the Report Fields • 360

D

dashboard • 92, 293, 294
calculated hits per day • 294
calculated pages per day • 294
calculated throughput per day • 294
failed hits • 294
hits per second • 294
pages per second • 294
rounds executed • 294
rounds failed • 294
running Virtual Clients • 294
throughput per second • 294
time • 294
Dashboard • 92
data collection
automatic • 320
definition • 317
data drilling • 93, 271, 335
automatic transactions • 248
collect instances • 212
instance grid • 338
open report • 336
transaction instance tree • 339

transaction reason failure grid • 337
transactions grid • 337
Data Drilling • 269, 271
Data Drilling and Table Compare Reports • 267, 272, 274, 335
Data Drilling Report • 93
Data Drilling Reports • 335
data sources
add • 376
add measurements to • 383
add to hosts • 384
application servers • 376
configure • 375
databases • 376
delete • 384
other • 376
PMM • 371
PMM Wizard • 376
stream technologies • 376
system • 376
Web servers • 376
database monitors • 387
Database Monitors • 387
databases
Microsoft SQL Server 2000 • 376
Microsoft SQL Server using SNMP • 376
monitoring • 365
Oracle • 376
resource monitor • 365
Deactivating Throttle Control • 286
default options
Agenda • 222
default project options • 260
functional testing • 260
default time in Integrated Report
set • 206
default viewer
open when export • 210
defect tracking
set • 215
Defining a Mix • 168, 169
Defining Host Computers • 194
Defining Load Machines • 194
Defining Performance Goals • 172, 173, 175
Defining Probing Client Machines • 195
Defining Test Session Options • 201
Defining the Acceleration Rate • 179, 182
Defining WebLOAD's Action Once the Goals are Reached • 183
Deleting a Data Source • 384
Deleting a Goal • 180
Deleting a Host Computer • 144, 174, 195
Deleting a Measurement • 385
Deleting a Monitored Host • 385
Deleting a Report • 300
Deleting a User-Agent • 234
Deleting an Agenda from a Mix • 140, 170
description
log window • 269
description pane
PMM • 370
Description Pane • 370
Designating a Host Computer as a Load Machine • 142, 173, 194
Designating a Host Computer as a Probing Client Machine • 142, 143, 173, 174, 195
Designating all Host Computers as Load Machines • 143, 174, 195
diagnostic
Agenda options • 243
ASM diagnostic messages • 245
automatic state management diagnostic messages • 245
case inspections • 244
enable • 243
log report diagnostics • 246
RadView support diagnostic • 247
scope inspections • 244
session options • 243
syntax check • 244
type inspections • 244
value inspections • 244
diagnostic options • 243

Diagnostic Settings • 224, 243
diagnostic tab • 243
Distributing the Load Equally • 145
DNS cache
    simulate • 236

E

edit mode • 36, 96
    Console • 79
    options • 85
    right-click menu options • 85
    session tree • 84, 96
Edit Mode and Run Modes of Operation • 80, 96
Editing a Report • 298
Editing the Load Template • 159
Enabling Dynamic Object Recognition Diagnostic Messages • 245
Enabling Persistent Connection (Keep-Alive) • 236
Enabling RadView Support Diagnostic • 247
Enabling Save CSJS and CSJS Diagnostic Messages • 247
Enabling Syntax Checking • 244
Error • 291
error handling • 269
Error messages • 397
    aphabetical listing of errors • 398
    logging • 287
    message statusSee Message status • 269, 397
    print • 290
    save • 290
    sort • 289
Errors in Communication • 278
Establishing Communication • 278
event viewer
    expected vs actual results • 270
Event Viewer • 94
Examining Your Test Resources • 117
Example • 395
Example of a Test Configuration • 71
Examples • 278
Expected Repository Files
    .fta • 403
expected vs. actual results • 270
Explanation of Creation and Analysis of Performance Reports • 273
export
    configuration files • 374, 375
    PMM configuration files • 374, 375
export options • 208, 209, 210, 211
export resolution
    set • 209
Export to HTML Settings • 210
Exporting a Summary Report to a Tab File • 362
Exporting a Summary Report to Excel • 362
Exporting All Statistical Data • 353
Exporting All Statistics to a Tab File • 356
Exporting All Statistics to Excel • 354
Exporting All Statistics to HTML • 357
Exporting Performance Reports • 274, 345
Exporting Performance Reports to a Tab File • 350
Exporting Performance Reports to Excel • 348
Exporting Performance Reports to HTML • 352
Exporting PMM Configuration Files • 375
Exporting Selective WebLOAD Report Data • 347
Extended Toolbar Buttons • 108
Extended Toolbar Segments • 107

F

failed connections
    definition • 320
failed hits
    definition • 320
failed hits per second
    definition • 315
failed pages per second
    definition • 315
failed rounds
    definition • 320
failed rounds per second
    definition • 315
failure

setting severity level for • 266
features • 28
File menu • 98
additional information • 98
create mix • 98
cruise control • 98
edit template • 98
exit • 98
new • 98
open load session • 98
open load template • 98
PMM • 368
print • 98
recent files • 98
save load session • 98
save load template • 98
WebFT • 98
WebLOAD REPORTER • 98
WebLOAD wizard • 98
File Menu • 98, 368
Finishing the Cruise Control Wizard • 184
firewall
work with • 73
floating license
details • 60
Floating License Server • 55, 57
form
Agenda level verification • 260
Forming a Test Plan • 115
forms
compare • 264
frame
Agenda level verification • 260
frames
compare • 264
full client
set • 225
function calls • 317, 320
functional testing • 247
automatic data collection • 248
compare forms • 264
compare frames • 264
compare images • 264
compare links • 264
compare tables • 264
compare title • 264
forms • 260
frames • 260
images • 260
links • 260
list of tests • 264
page time verifications • 249
report default values • 260
reporting events • 267
select all tests • 263
select tests • 262, 263
tables • 260
test selection • 251
text search • 264
titles • 260
track transaction failures • 267
user-defined transactions • 264
verification • 251
verification function return values • 266
verify text • 260
view errors • 269
view results • 269
Functional Testing Options • 224, 247
functional testing tab • 247
Agenda options • 247
open • 260
session options • 247
functional verification
test selection • 251
Functional Verification Test List • 252, 264
Functional Verification Testing • 201, 252, 259

G

general options
autosave file name setting • 205
statistic view set • 208
Getting Started • 19, 21, 121, 226

global options • 203
acceleration point measurements set • 206
advanced • 213
autosave file name setting • 205
data drilling • 212
default integrated report time range set • 206
defect tracking • 215
export • 208, 209, 210, 211
file locations • 211
general tab • 205
goal reached delay setting • 206
open • 203
set • 203
statistic view set • 208
statistics reporting resolution set • 206
globaloOptions
load generator statistics set • 206
goal defaults
set • 180
use • 180
goal definition
about • 175
add goal • 177
control buttons • 177
delete goal • 177
measurement • 176
operator • 176, 177
report • 176
set as default • 177
set defaults • 177
source to track • 176
value • 176
goal table
about • 176

H

Help menu • 105
PMM • 369
Help Menu • 105, 369
hit time
definition • 317
hits
calculated per day • 294
definition • 320
failed • 294
hits per second • 294
definition • 315
host
communication • 62
define host computers • 194
selection window • 193
host computer
add • 173
cruise control • 172
define • 173, 194
delete • 174, 195
designate all load machines as • 174
designate load machine as • 173
designate probing client as • 174
select • 193
select in cruise control wizard • 172
host computers
add • 142
define • 142
select • 141
host selection dialog box
open • 193
hosts
add • 376
add data sources • 384
configure • 375
delete • 385
PMM • 371
How are Tests Defined? • 25
How does Cruise Control Work? • 162
How Does WebLOAD Work? • 22
How Does WebRM Work • 123
How to Set Global Options • 203
HTML Export Settings • 211
HTML file
export settings • 210
HTTP Client • 26

HTTP protocol mode
set • 225
HTTP version
set • 237

I

IBM WebSphere
configuration tasks • 387
configure • 387
data source • 376
IBM WebSphere Configuration Tasks • 387
icons
PMM • 371
status • 82
image
Agenda level verification • 260
images
compare • 264
import
configuration files • 373
PMM configuration files • 373
Importing a Complete Report Configuration • 297, 303
Importing a Report Configuration for a Selected Component • 297, 304
Importing Report Parameters from a Saved Load Template • 297, 303
Including a User-Defined Transaction in your Agenda • 265
Incrementing Intervals • 152
Incrementing Intervals (time calculate) • 153
incrementing intervals (time calculate) model
base load size • 153
concluding load size • 153
load profiler • 153
parameters • 153
time between each interval T1 • 153
time of each interval T2 • 153
time to increase each interval • 153
incrementing intervals model
base load size • 152
load profiler • 152
load to increase each interval • 152
parameters • 152
time between each interval T1 • 152
time of each interval T2 • 152
total time in minutes • 152
Inserting a Text Verification Test • 250
Inserting Test Selections • 251
instal
TestTalk on Windows • 64
Installing SNMP for ORACLE • 64
Installing SNMP for WebLogic • 63
Installing SNMP Service on a Windows 2000 Platform • 63
Installing WebLOAD • 51, 62
instance grid
duration • 338
load generator • 338
reason • 338
round number • 338
severity • 338
thread number • 338
time • 338
total load size • 338
Integrated Report • 90, 293
add a saved Load Session • 305
add a user-defined timer • 303
change report parameters • 302
change the scale value • 308
chart time window width • 206
chart view • 306
configure • 300
deleting a report • 300
graphic format • 306
import report parameters • 303
include data from saved Load Sessions • 305
new report • 295
open an existing report • 298
regression testing • 305
remove a saved Load Session • 306
remove data from saved Load Sessions • 306
select measurements • 302

spread view • 310
tabular format • 310
view true value vs. relative value • 308
Integrated Report Chart View • 306
Integrated Report Definition • 362
Integrated Report in Spread(sheet) View • 310
Integrated Report Toolbar in Chart View • 308, 309
Introducing WebLOAD • 19
Introduction • 15

J

Java
counters • 320
timers • 317

K

keep-alive
enable • 236
keyboard
navigating PMM with • 372
navigating session tree with • 83
shortcut keys for navigating PMM • 372
shortcut keys for navigating session tree • 83
keyboard shortcuts
for navigating PMM • 372
for navigating session tree • 83

L

Launching WebLOAD through the Command Line Interface • 277
License Registration and Update • 52
linear model
concluding load size • 150
load profiler • 150
parameters • 150
starting load size • 150
total time in minutes • 150
Linear Scheduling Model • 150
link
Agenda level verification • 260
links
compare • 264
load generator
connection speed • 242
set thread properties • 235
sharing data • 213
load generator level statistics
set • 206
load generator software
load session diagram • 23
load generator spawning
setting • 213
load machine
about • 172
define • 194
definition • 23
delete • 144
designate • 142
designate all as host computers • 174
designate as host computer • 173
icon • 82
load session diagram • 23
replace • 143
schedule equally in WebLOAD wizard • 145
load machines
select • 193
load profiler
incrementing intervals (time calculate) model • 153
incrementing intervals model • 152
linear model • 150
ramp up model • 156
random model • 151
schedule equal load manually • 147, 149
schedule load manually • 197
scheduling models • 150
step increment (time calculate) model • 155
step increment model • 154
user defined model • 157
Load Session • 95
.ls • 129
available resources • 117

configure • 113
diagram • 23
establishing communication • 278
goals of the test • 115
icon • 82
launch through command line • 277
launch through TestView Scheduler • 276
manage • 129
open • 132
operating commands • 282
pause command • 283
plan • 113, 115
resume command • 283
run • 113
runn from system memory • 276
running • 275
save • 129
save additional information • 130
start • 117, 275
start command • 283
stop all command • 284
stop command • 284
view performance reports • 113
workflow • 113
Load Session Workflow • 113
load size
change dynamically during runtime • 284
definition • 320
load template • 95, 125
.tpl • 125
Agenda selection • 190
define manually • 190
edit • 159
manual defintion workflow • 190
new • 126
open • 127
run • 113
save • 127
save additional information • 127
load template configuration
add Agenda • 191
add all load machines • 195
add load machine • 194
add mix • 192
add probing client • 195
add probing clients • 195
delete host computer • 195
host computer definition • 193, 194
schedule load machines • 195
schedule load machines with load profiler • 197
load template defintion
set Agenda options • 191, 192
view Agenda • 191, 192
Log Files
.log • 403
log message
options • 208
view • 289
log report
diagnostics • 246
log report diagnostics
enable • 246
Log Report Diagnostics • 246
log window • 94
attachments • 269
customize • 287
description • 269
error • 291
log error messages • 287
message attachments • 269
message source • 269
message status • 269
options • 287
print contents • 271
print error messages • 290
save contents • 271
save error messages • 290
severe error • 291
sort error messages • 289
test summary • 269
view • 288

view message status • 290
viewiresults • 269
warning • 291
Log Window • 94
logwindow
usernote • 291

M

main window
PMM • 367
Managing Load Sessions • 129
Managing Load Templates • 125, 159
Managing Load Templates and Load Sessions • 95, 112, 125
manual test configuration
workflow • 190
manual verification tests
text • 262
Manually Scheduling the Load to Generate on Each Load Machine • 145, 148
max • 323
max time
page time verifications • 249
measurements
add • 376
add to data sources • 383
configure • 375
delete • 385
PMM • 371
menu bar • 80
Menu Bar • 80, 367
menus
file menu • 98
help menu • 105
reports menu • 100
session control menu • 99
tools menu • 104
view menu • 101
windows menu • 105
Message Severity • 397
message source • 269
Message status • 269, 290, 397
Error • 269, 291, 397
Info Message • 269
Minor Error • 269
Severe Error • 291, 397
Success • 269
User Note • 291, 397
Warning • 291, 397
MIB files
understanding • 392
Microsoft ASP .Net
data source • 376
Microsoft Active Server Pages
data source • 376
Microsoft Internet Information Server (IIS)
data source • 376
Microsoft Media Server
data source • 376
Microsoft SQL Server 2000
data source • 376
Microsoft SQL Server using SNMP
data source • 376
min • 323
mix • 136
add Agenda • 169, 170
assign Agenda weight • 140
assign equal weight • 170
create • 139, 140, 169
cruise control wizard • 165
define • 139, 169
delete an Agenda • 140
delete an Agenda • 170
icon • 82
make equal • 169
new • 171
run a mix of Agendas • 167
save • 140, 171
select • 167, 191
select in cruise control wizard • 167
select in WebLOAD wizard • 136
set Agenda options • 169
set Agenda weight • 169

set weight • 170
WebLOAD wizard • 138
weight • 169
modes of operation
edit mode • 96
run mode • 96, 97
Session mode • 281
Modifying Agenda Options • 224
Modifying the Host Selection • 160
Modifying the Load Schedule • 160
monitor
SNMP data sources • 392
monitoring
application server resources • 365
database resources • 365
databases • 365
performance • 365
PMM • 365
real-time • 365
server-side applications • 365
stream technology • 365
stream technology resources • 365
system resources • 365
web server resources • 365
Monitoring Generic SNMP Data Sources • 392
multithreading • 235
MySAP.com
data source • 376

N

named transactions • 248
Navigating the PMM Configuration Tree • 372
Navigating the Session Tree • 83
navigation block • 266
Netscape LDAP
data source • 376
notification mail
manager • 215
Notification Manager • 215

O

Online Help • 17
open
configuration files • 373
PMM configuration files • 373
Open Viewer When Exporting HTML Files • 211
Open Viewer When Exporting Tab Files • 210
Opening / Importing Saved PMM Configuration Files • 373
Opening a Data Drilling Report • 336
Opening a Saved Load Template • 127
Opening an Existing Report • 298
Opening Reports • 295
Opening Saved Load Sessions Files • 132
Opening TestTalk • 64
Opening the Console • 78
Opening the Cruise Control Wizard • 163
Opening the Functional Testing Tab • 260, 262, 263
Opening the Host Selection Dialog Box • 193
Opening the Log Window • 288
Opening the Log Window Menu • 289
Opening the Performance Measurements Manager • 366
Opening the Session Tree • 81
Opening the Table Compare Viewer • 343
Opening the WebLOAD Wizard • 40, 135
Opening Throttle Control • 285
operating commands • 282
pause • 283
resume • 283
start • 283
stop • 284
stop all • 284
Operator • 177
options
global • 203
set for Agenda • 221
set for current session • 221
Oracle
configure on Windows • 387, 389
configure SNMP master agent • 389
data source • 376
start SNMP agents • 390

Windows configuration tasks • 387, 389
Oracle Monitor Configuration Tasks on UNIX • 389
Oracle Monitor Configuration Tasks on Windows • 387
other
general performance monitor • 376
general SNMP • 376
general UNIX • 376

P

page time verifications
Agenda option • 249
enable • 249
max time • 249
session option • 249
severity level • 249
verification • 249
pages
calculated per day • 294
per second • 294
pages per second
definition • 315
Parameters • 277
pass / fail definition
Agenda options • 252
session options • 252, 258
set absolute • 252
set relative • 252
Pass / Fail Definition Settings • 224, 252
pause
test session • 283
Pausing a Test Session • 283
per time-unit statistics
current value calculation • 315
defined • 313
list of • 315
Per Time-Unit Statistics Definitions • 315
Performance Measurements Manager • 137, 138, 139, 167, 168, 274, 300, 365
performance measurements managersee PMM • 365
performance monitor
data source • 376
persistent connection
enable • 236
ping
acknowledgments • 67
host • 67
Pinging a Host • 67, 117
Planning a Test Program • 116
Planning a Test Session • 112, 113
PMM
about • 366
Actions menu • 369
activate monitors • 366
add data sources • 376
add data sources to hosts • 384
add hosts • 376
add measurements • 376
add measurements to data sources • 383
adding data source • 369
application server monitors • 385
application servers data sources • 376
clear all • 369
close and update template • 368
collapse tree • 369
configuration files • 373
configuration pane • 370
configuration tree • 370
configuration tree icons • 371
configuration tree structure • 370
configurations • 366
create new configuration files • 373
data sources • 366, 371, 375
database monitors • 387
databases data sources • 376
delete • 369
delete data sources • 384
delete hosts • 385
delete measurments • 385
description pane • 370
exit • 368

expand tree • 369
export configuration files • 374, 375
exporting configuration files • 368
extend the MIB filter • 392
File menu • 368
Help menu • 369
hosts • 366, 371, 375
import configuration files • 373
importing configurtion files • 368
Interated Report • 366
keyboard shortcut keys for navigating • 372
main window • 367
measurements • 366, 371, 375
menu bar • 367
monitor databases • 365
monitor server side applications • 365
monitor SNMP data sources • 392
monitor stream technology • 365
monitor Web server measurements • 365
monitoring performance • 365
new • 368
new configuration • 368
open • 366, 368
open configuration files • 373
open PMM Wizard • 376
other data sources • 376
real-time monitor • 365
save configuration files • 374, 375
savinf configuration files • 368
standalone tool • 366
stream technologies data sources • 376
system data sources • 376
template • 366
understanding MIB files • 392
understanding the MIB filter • 392
UserMIBfilter.ini structure • 393
Web server monitors • 391
Web servers data sources • 376
Wizard • 366

PMM Actions menu
add data source • 369
clear all • 369
collapse tree • 369
delete • 369
expand tree • 369

PMM configuration files
append • 373
attach to WebLOAD templates • 373, 374
create new • 373
export • 374, 375
import • 373
new • 373
open • 373
replace • 373
save • 373, 374, 375
use • 373

PMM configuration tree
icons • 371

PMM File menu
close and update template • 368
exit • 368
export • 368
import • 368
new • 368
new configuration • 368
open • 368
save • 368

PMM Wizard
data source screen • 376
finish screen • 376
host selection screen • 376
measurements selection screen • 376
open • 376
welcome screen • 376

Printing Error Messages Recorded in the Log Window • 290

Printing the Contents of the Log Window • 271

probing client
about • 172
connection speed • 242
define • 143, 195
designate • 143

designate as host computer • 174
set thread properties • 235
probing client machine
icon • 82
load session diagram • 23
probing client software
load session diagram • 23
probing clients • 121
select • 193
process time
definition • 317
Project Files
.pft • 403

Q

Quick Start Steps • 35

R

RadView support
diagnostics • 247
RadView support diagnostic
enable • 247
ramp up
base load size • 156
load profiler • 156
max load size • 156
parameters • 156
ramp down time • 156
ramp up time • 156
time to run max load size • 156
Ramp Up • 156
Random • 151
random model
load profiler • 151
max load size • 151
min load size • 151
parameters • 151
start load size • 151
total time in minutes • 151
real-time monitor
PMM • 365
receive time
definition • 317
Record
Agenda • 36
start • 36
stop • 36
redirection
set limits • 235
Reference Materials • 403
regression testing • 305
Regression Testing • 305
Removing a Previously Saved Load Sessions from the Integrated Report Configuration • 306
Renaming a Report • 299
replace
configuration files • 373
PMM configuration files • 373
Replacing a Load Machine or Probing Client Machine • 143
report
events • 267
report default values
Agenda level verification • 260
Report Destination • 361
Report Export Options • 345
reported valueSee current value. • 315
ReportEvent
description • 267
event-name • 267
syntax • 267
Reporting Events • 267
reporting intervalSee statistics reporting resolution. • 314
reports
dashboard • 92, 293, 294
data drilling • 93, 335
export • 345
export all statistical data • 353
export all statistics to a HTML file • 357
export all statistics to a tab file • 356
export all statistics to Excel • 354
export options • 345

export selective data • 347
export to a HTML • 352
export to a tab file • 350
export to Excel • 348
Integrated • 293
set • 206
set options • 206
statistics • 313
summary report • 359
Reports menu • 100
dashboard • 100
data drilling • 100
delete integrated report • 100
export • 100
export all • 100
export all to a tab file • 100
export all to Excel • 100
export all to HTML • 100
export to Excel • 100
export to HTML • 100
export to tab file • 100
integrated reports • 100
new integrated report • 100
open integrated report • 100
statistics report • 100
summary report • 100
Reports Menu • 100
Requesting Resources from WebRM • 123
Required Software for each Load Session Component • 70
resource manager • 121
connected workstation • 23
standalone workstation • 23
Resource ManagerSee WebRM • 121
response data size
definition • 320
response time
definition • 317
responses
definition • 320
results
view in log window • 269
results description
additional information • 130
results window • 89
dashboard • 92
data drilling • 93
integrated report • 90
resume
test session • 283
Resuming a Test Session • 283
return status
definition • 320
return values
Error • 266
MinorError • 266
SevereError • 266
Success • 266
Right-Click Options in Edit Mode • 85
Right-Click Options in Run Mode • 88
round time
definition • 317
rounds
definition • 320
executed • 294
failed • 294
rounds per second
definition • 315
run all tests • 260
run mode • 96
Console • 79
options • 88
right-click menu options • 88
session tree • 86, 97
status icons • 82
view the session tree • 86
Running a Load Session • 100, 115, 199, 274, 275
Running a Load Session from the System Memory • 276
Running a Mix of Agendas • 166, 167
Running Single Agendas • 165, 166
Running Tests and Analyzing Test Results • 273

Running the Test • 36, 44

S

save
configuration files • 374, 375
PMM configuration files • 374, 375
save flag • 265
Saving a Load Session • 129
Saving a Load Template • 127
Saving a Mix • 140, 171
Saving Additional Information for the Load Session • 130
Saving Additional Information with the Load Template • 127
Saving and Exporting PMM Configurations • 374
Saving Error Messages Recorded in the Log Window • 290
Saving PMM Configuration Files • 375
Saving the Contents of the Log Window • 271
Saving the Current Load Template • 276
schedule
manual • 148
WebLOAD wizard • 144
scheduling
manually • 196
Scheduling a Session Manually with the Load Profiler • 149
scheduling model
incrementing intervals • 152
incrementing intervals(time calculate) • 153
linear • 150
load profiler • 150
ramp up • 156
random • 151
step increments • 154
step increments (time calculate) • 155
user defined • 157
Scheduling Models • 147, 150, 198
Scheduling the Load Session • 142, 143, 144
Scheduling the Session Manually • 148
Scheduling the Test • 42
Scheduling the Test Session • 195
Scheduling WebLOAD Tests with TestView Scheduler • 276
scope inspections
syntax check • 244
search paths
set default • 211
Selecting a Host • 42
Selecting a Mix • 136, 138
Selecting an Agenda • 136, 137, 191
Selecting an Agenda and a Performance Statistic to Monitor • 41
Selecting an Agenda or Mix • 136, 165
Selecting Browser Client Type Mode • 227
Selecting Host Computers • 138, 140, 141, 167, 170, 171, 172
Selecting Hosts • 193
Selecting HTTP Client Type Mode • 227
Selecting Measurements for the Integrated Report • 297, 302
Selecting Specific Tests to Run • 263
Selecting the Tests to Perform • 262
send time
definition • 317
server monitor
statistics reports • 317
server-side applications
monitoring • 365
session comments
additional information • 130
session control menu • 99
delete • 99
modify agenda selection • 99
modify host selection • 99
modify schedule • 99
open session tree • 99
Pause • 99
performance statistics manager • 99
resume • 99
start session • 99
stop • 99
stop all • 99
throttle control • 99

Session Control Menu • 99
Session Files
.fts • 403
session mode
Console • 281
viewing the Session Tree • 282
session options
authentication for protocol scripts • 240
automatic data collection • 248
browser dialog tab • 258
browser emulation tab • 231
client type tab • 225
connection speed • 242
diagnostic tab • 243
functional testing tab • 247
sleep time control tab • 227
web service tab • 258
Session Options • 201
Session Tree • 81
edit mode • 84, 96
editing configuration • 96
icons • 82
navigate with keyboard shortcuts • 83
open • 81
right mouse button functions • 96, 97
run mode • 86, 97
session mode • 282
status icons • 82
viewing configuration • 97
Session Tree Icons • 82
cruise control • 206
SetFailureReason
syntax • 267
Setting Additional Information Properties • 128, 130
Setting Agenda Options • 86, 104, 105, 137, 166, 169, 192, 201, 221
Setting and Using Defaults • 177, 180
Setting Authentication • 224, 240
Setting Automatic Data Collection • 248
Setting Cruise Control Options • 206
Setting Defaults • 180
Setting Global Options • 104, 118, 201, 203
Setting Load Generator and Probing Client Thread Properties • 235
Setting Page Time Verifications • 249
Setting Redirection Limits • 235
Setting Session Options • 36, 43
Setting Sleep Time Playback Settings • 224, 227
Setting the Autosave File Name Default Location • 205
Setting the Browser Cache • 224, 229
Setting the Browser Type and User-Agent • 232
Setting the Export to Tab File Settings • 210
Setting the HTTP Parameters • 224, 237
Setting the Log Message Options • 208
Setting the Log Window Options • 287
Setting the Reports Options • 206
Setting the SSL Options • 224, 225
Setting the Statistic View Settings • 208
Setting the Statistics Resolution to Export • 209
Setting the Throttle Control Max Value • 207
Setting up the SNMP Agent and Configuring the Sun One Server as a Proxy for the SNMP Agent • 391
Setting Up the SNMP Agent and Configuring the WebLogic Server as a Proxy for the SNMP Agent • 386
Setting WebLOAD Timeout Values • 207
Severe • 291
severity level • 266
Agenda verification • 260
page time verifications • 249
Sharing an Agenda • 72
Simulating DNS Cache • 236
Simulating SSL Cache • 236
sleep time
as recorded • 227
ignore • 227
set • 227
set deviation • 227
set random • 227
sleep time control

Agenda options • 227
session options • 227
SNMP • 62
agents • 394
configure master agent on UNIX • 389
data source • 376
instal for ORACLE • 64
instal for WebLogic • 63
instal on Windows 2000 • 63
start agents on UNIX • 390
SNMP agent
add • 394
BEA WebLogic • 386
set up • 386
Sun One • 391
SNMP data sources
monitor • 392
snmpd.config File • 390
Sorting Error Messages • 289
SSL bit limit
set • 225
SSL cache
simulate • 236
standalone workstation
WebLOAD component • 23
standard deviation. • 323
Standard License Registration • 53
start
test session • 283
start_peer Script • 390
Starting a Load Session • 275
Starting a New Template Manually • 190
Starting a Test Session • 283
Starting the SNMP Agents • 390
Starting WebLOAD • 117
statistic view
set • 208
Statistical Measurements and Their Values • 313
statistics
current value • 313
defined • 313
differences between probing and virtual clients • 322
report resolution • 314
statistics exporting resolution
set • 209
Statistics for Virtual Clients and Probing Clients • 322
statistics report • 313
history by measurement and Agenda • 331
history by measurement and total • 332
main window • 313
summary by Agenda • 325
summary by measurement • 329
summary by total • 327
summary reports • 323
use • 324
Statistics Report • 91
Statistics Report-History by Measurement and Agenda • 326, 330, 331
Statistics Report-History by Measurement and Total • 328, 332
statistics reporting resolution
set • 206
Statistics Report-Summary by Agenda • 325
Statistics Report-Summary by Measurement • 329
Statistics Report-Summary by Total • 327
statistics window • 91
status bar • 81
Status Bar • 81
status icons
currently running • 82
error • 82
initialization complete • 82
no current activity • 82
stopped • 82
temporarily paused • 82
Status Icons • 81, 82
step increment (time calculate) model
load profiler • 155
step increment model
load profiler • 154
step increments

load to increase each interval • 154
parameters • 154
starting load size • 154
time of each interval • 154
total time in minutes • 154
Step Increments • 154
step increments (time calculate)
concluding load size • 155
load to increase each interval • 155
parameters • 155
starting load size • 155
time of each interval • 155
Step Increments (time calculate) • 155
stop
test session • 284
stop all
test session • 284
Stopping a Load Session Component • 284
Stopping a Test Session • 284
storage locations
set default • 211
stream technologies
Microsoft Media Server • 376
monitoring • 365
resource monitor • 365
subject
additional information • 128, 130
successful connections
definition • 320
successful hits
definition • 320
successful hits per second
definition • 315
successful pages per second
definition • 315
successful rounds
definition • 320
successful rounds per second
definition • 315
sum • 323
summary report • 323
create • 359
values • 323
summary report wizard
report destination • 361
report fields • 360
Summary Reports • 323
Sun One
configuration tasks • 391
configure • 391
data source • 376
set as proxy for SNMP agent • 391
Sun One (iPlanet) Configuration Tasks • 391
Syntax • 278
syntax check
case inspections • 244
enable • 244
scope inspections • 244
type inspections • 244
value inspections • 244
system
resource monitor • 365
UNIX platforms network data using SNMP • 376
UNIX platforms using RSTATD • 376
UNIX UC-Davis • 376
Windows .Net framework • 376
Windows NT/2000/XP • 376

T

tab file
export settings • 210
Tab File Export Settings • 210
table
Agenda level verification • 260
Table Compare Reports • 341
table compare viewer • 341
open • 343
table compare events • 340, 341
tables
compare • 264
Technical Support • 17
Technical Support Web Site • 18

template comments
additional information • 128, 130
templates
attach PMM configuration files • 373, 374
templateSeelLoad template • 95
test
configuration/tSee Configuration • 69
schedule • 195
schedule manually • 196
schedule with load profiler • 197
text • 262
Test description
additional information • 128, 130
test parameters
edit • 96
test selection
functional verification • 251
test session
diagram • 23
test summary • 269
Testing Your Connection • 67, 68, 69
TestTalk
definition • 23
file menu • 64
help menu • 64
instal on Windows • 64
network configuration parameters • 66
open • 64, 117
view menu • 64
TestTalk for Windows • 64
TestView Scheduler
launch WebLOAD • 276
text
search • 262, 264
test • 262
verification of • 262
text not to find • 262
text search
Agenda level verification • 260
text to find • 262
The Advanced Tab • 213
The Console in Edit Mode • 96
The Console in Run Mode • 97
The Console Menu Options • 98, 134
The Console Screen in Session Mode • 281
The Console Toolbar • 80, 105
The Cruise Control Wizard Workflow • 163
The Data Drilling Tab • 212
The Defect Tracking Tab • 215
The Export Tab • 208
The File Locations Tab • 211
The Functional Testing Tab • 260
The General Tab • 205, 280
The Goal Table • 176
The Manual Test Configuration Workflow • 190
The Notification Manager • 215
The Performance Measurements Manager Main Window • 367
The PMM Configuration Tree Structure • 370
The Required Connections • 67
The Results Window • 89
The Session Tree • 81
The Session Tree in Edit Mode • 84, 89
The Session Tree in Run Mode • 86
The Session Tree Structure • 82
The WebLOAD Console • 77
The WebLOAD Console Shortcut Screen • 95
The WebLOAD UserMibFilter.ini File Structure • 393
The WebLOAD Wizard Workflow • 134
thread properties
set • 235
throttle control • 284
activate • 286
de-activating • 286
max value set • 207
maximum number of virtual clients • 207
open • 285
Throttle Control - Changing the Load Size Dynamically During Runtime • 284
throughput
calculated per day • 294
definition • 315

per second • 294
time • 294
time to first byte
definition • 317
timeout
set options • 207
start session timeout set • 207
terminate session timeout set • 207
Timer Definitions • 317
timer statistics
current value calculation • 315
defined • 313
list of • 317
timers
for Java and ActiveX • 317
for user-defined transactions • 317
user-defined
definition • 317
title
additional information • 128, 130
Agenda level verification • 260
compare • 264
title bar • 79
Title Bar • 79
toolbar
buttonstoolbar buttons • 106
Console • 105
customize • 103
file management options • 105
run commands • 105
session control options • 105
WebLOAD wizards • 105
Toolbar • 80
toolbar buttons
cascade windows • 108
close window • 108
cruise control wizard • 106
customize • 104
dashboard • 108
delete • 106
delete report • 108
edit template • 106
export all to Excel • 108
export all to HTML • 108
export all to tab file • 108
export to Excel • 108
export to tab file • 108
export toHTML • 108
functional verification • 106
global options • 106
host selection • 106
log window • 106
new • 106
new report • 108
open data drilling • 108
open load template • 106
open report • 108
open statistics report • 108
optional buttons • 107, 108
pause • 108
print • 106
resume • 108
save load session • 106
session options • 106
session tree • 106
start • 106
stop all • 108
summary report • 108
throttle control • 108
tile horizontally • 108
tile vertically • 108
WebFT • 106
WebLOAD REPORTER • 106
WebLOAD wizard • 106
Toolbar Buttons • 106
toolbars • 80
Toolbars Dialog Box • 103
tools
Agenda options • 104
current project options • 260
current session options • 104
default project options • 260

tools menu • 104
    default options • 104
    global options • 104
Tools Menu • 104
Tracking Reasons for Transaction Failure • 267
transaction failure
    track • 267
transaction name • 265
transaction reason failure grid • 337
transactions
    named • 248
    user-defined • 265, 317, 320
transactions grid
    failed transactions • 337
    marked count • 337
    successful count • 337
    total count • 337
    transaction name • 337
troubleshooting
    communications • 68
Troubleshooting Communications • 68
type inspections
    syntax check • 244
Typographical Conventions • 16

U

Understanding MIB Files • 392, 394
Uninstalling WebLOAD • 61
UNIX
    data source • 376
UNIX platforms network data using SNMP
    data source • 376
UNIX platforms using RSTATD
    data source • 376
UNIX UC-Davis
    data source • 376
Updating the Number of Requested Resources • 124
Upgrading WebLOAD • 62
user agent
    add • 233
    delete • 234
    set • 232
user defined
    data file name • 157
    delimeter • 157
    load profiler • 157
    max load size • 157
    min load size • 157
    parameters • 157
    total time in minutes • 157
User Defined • 157
User Note • 291
user-defined
    counters • 320
    timers • 317
    transactions • 264, 265, 317, 320
User-Defined Transaction Example • 265
user-defined transactions
    syntax • 265
User-Defined Transactions and Verification • 264
User-Defined Transactions and Verification Syntax • 265, 267
UserMIBFilter.ini
    attr_names • 393
    attr_range • 393
    base • 393
    extend • 392, 394
    file • 393
    instance • 393
    instdesc • 393
    numbase • 393
    prefix • 393
    structure • 393
    understanding • 392
Using Cruise Control • 161
Using Defaults • 181
Using PMM Configuration Files • 373
Using the Performance Measurements Manager • 365
Using the Statistics Report • 324
Using the WebLOAD Wizard to create Load Templates • 133

V

value inspections
    syntax check • 244
verification
    Agenda option • 251
    compare forms • 251
    compare images • 251
    compare links • 251
    compare no of frames • 251
    compare tables • 251
    compare title • 251
    enable • 251
    page time verifications • 249
    run all tests • 251
verification function • 265
    return values • 266
Verification Function Return Values • 250, 252, 262, 264, 266
verification test
    compare forms • 264
    compare images • 264
    compare links • 264
    compare no of frames • 264
    compare tables • 264
    compare title • 264
    list of tests • 264
    perform • 262
    text search • 264
    view errors • 269
    view results • 269
verification tests
    for entire Agenda • 260
    form • 260
    frame • 260
    image • 260
    link • 260
    report default values • 260
    table • 260
    text • 260, 262
    title • 260
Verifying Text • 262, 263
view menu • 101
    clear log window • 101
    commands • 101
    gauges. • 101
    log window • 101
    session tree window • 101
    toolbar • 101
    workbook • 101
View Menu • 101
Viewing Data Drilling Reports • 47
Viewing Error Information in the Event Viewer • 270
Viewing Error Information in the Log Window • 269
Viewing License Details • 60
Viewing Results with WebLOAD REPORTER • 48
Viewing Test Results on the WebLOAD Dashboard • 294
Viewing the Child Transaction Instance Tree • 340
Viewing the Cruise Control Default Report • 186
Viewing the Entire Log Message • 289
Viewing the Instance Grid • 338
Viewing the Log Window • 208, 274, 287
Viewing the Message Status • 290
Viewing the Network Configuration • 66
Viewing the Parent Transaction Instance Tree • 339
Viewing the Results in Real Time • 46
Viewing the Session Tree in Session Mode • 281, 282
Viewing the Transaction Grid • 337
Viewing the Transaction Reason Failure Grid • 337
Viewing the True Value vs. Relative Value of a Measurement • 308
Viewing the WebLOAD Wizard Default Report • 159
Viewing Verification Test Results • 269
Virtual Clients • 121
    running • 294

W

WAP
    set browser type • 232
Warning • 291
web server
    resource monitor • 365
Web server monitors • 391
Web Server Monitors • 391
Web servers
    Microsoft Internet Information Server (IIS) • 376
    Netscape LDAP • 376
    Sun One • 376
web service tab
    Agenda options • 258
    session options • 258
web services
    authentication • 258
    client • 258
Web Services Settings • 224, 258
WebFT • 113
    create new Agenda • 36
    definition • 23
    edit mode • 36
    start • 36
WebLOAD
    architecture • 23
    dashboard • 293, 294
    description • 21
    features • 28
    files used by WebLOAD • 403
    launch through command line interface • 277
    launch through TestView Scheduler • 276
    open • 117
    start • 117
    upgrade • 62
WebLOAD Architecture • 23
WebLOAD Console Features • 19, 77
WebLOAD Documentation • 15
WebLOAD Error Messages • 266, 397, 398
WebLOAD Features • 28
WebLOAD Files • 403
WebLOAD Hot Key Commands • 404
WebLOAD Installation and Configuration • 19, 51
WebLOAD Offers Two Client Type Modes • 26
WebLOAD Operating Commands • 282
WebLOAD Options for Creating Load Templates: • 111
WebLOAD Performance Reports • 274, 293, 313
WebLOAD Quick Start • 35
WebLOAD Resource ManagerSee WebRM • 121
WebLOAD Statistics Reports • 274, 313
WebLOAD templates
    attach PMM configuration files • 373, 374
WebLOAD wizard • 95
    Agenda selection • 137
    Agenda source dialog box • 135
    Agenda/Mix selection • 136
    assign Agenda weight • 140
    default report • 159
    define a mix • 139
    delete Agenda from mix • 140
    delete load machine • 144
    designate load machine • 142
    designate probing clients • 143
    finish dialog box • 158
    host selection dialog box • 141
    manually schedule • 148
    mix create • 138, 139, 140
    mix select • 138
    modify host selection • 160
    modify load machine • 143
    modify load schedule • 160
    open • 135
    replace load machine • 143
    save a mix • 140
    schedule equal load • 145
    schedule equal load manually • 146
    schedule equal load with load profiler • 147, 149
    schedule session • 144
    select host computers • 141

view test results • 159
WebLOAD.exe command • 277
WebLogic Monitor Configuration Tasks • 385
WebRM • 121
connect to • 58
floating license • 57
Welcome to WebLOAD • 21
What is the Product of Your WebLOAD Testing? • 26
Where to Get More Information • 17
Window Menu • 105
Windows
data source • 376
Windows .Net framework
data source • 376
windows menu • 105
Working with a Firewall • 73
Working with WebRM • 123