LoadRunner®

Controller User's Guide

Windows

Version 7.02

MERCURY INTERACTIVE

LoadRunner® Controller User's Guide (Windows), Version 7.02

© Copyright 1994 – 2001 by Mercury Interactive Corporation

All rights reserved. All text and figures included in this publication are the exclusive property of Mercury Interactive Corporation, and may not be copied, reproduced, or used in any way without the express permission in writing of Mercury Interactive. Information in this document is subject to change without notice and does not represent a commitment on the part of Mercury Interactive.

Mercury Interactive may have patents or pending patent applications covering subject matter in this document. The furnishing of this document does not give you any license to these patents except as expressly provided in any written license agreement from Mercury Interactive.

WinRunner, XRunner, LoadRunner, TestDirector, TestSuite, WebTest, and Astra are registered trademarks of Mercury Interactive Corporation in the United States and/or other countries. Astra SiteManager, Astra SiteTest, Astra QuickTest, Astra LoadTest, Topaz, RapidTest, QuickTest, Visual Testing, Action Tracker, Link Doctor, Change Viewer, Dynamic Scan, Fast Scan, and Visual Web Display are trademarks of Mercury Interactive Corporation in the United States and/or other countries.

This document also contains registered trademarks, trademarks and service marks that are owned by their respective companies or organizations. Mercury Interactive Corporation disclaims any responsibility for specifying which marks are owned by which companies or organizations.

If you have any comments or suggestions regarding this document, please send them via e-mail to documentation@mercury.co.il.

Mercury Interactive Corporation
1325 Borregas Avenue
Sunnyvale, CA 94089 USA

LRCPCUG7.02/01

Table of Contents

PART II: DESIGNING A SCENARIO

PART IV: MONITORING A SCENARIO

Welcome to LoadRunner

Welcome to LoadRunner, Mercury Interactive's tool for testing the performance of applications. LoadRunner stresses your entire application to isolate and identify potential client, network, and server bottlenecks.

LoadRunner enables you to test your system under controlled and peak load conditions. To generate load, LoadRunner runs thousands of Virtual Users that are distributed over a network. Using a minimum of hardware resources, these Virtual Users provide consistent, repeatable, and measurable load to exercise your application just as real users would. LoadRunner's in-depth reports and graphs provide the information that you need to evaluate the performance of your application.

Online Resources

LoadRunner includes the following online tools:

Read Me First provides last-minute news and information about LoadRunner.

Books Online displays the complete documentation set in PDF format. Online books can be read and printed using Adobe Acrobat Reader, which is included in the installation package. Check Mercury Interactive's Customer Support web site for updates to LoadRunner online books.

LoadRunner Online Function Reference gives you online access to all of LoadRunner's functions that you can use when creating Vuser scripts, including examples of how to use the functions. Check Mercury Interactive's Customer Support site for updates to the *LoadRunner Online Function Reference*.

LoadRunner Context Sensitive Help provides immediate answers to questions that arise as you work with LoadRunner. It describes dialog boxes, and shows you how to perform LoadRunner tasks. To activate this help, click in a window and press F1. Check Mercury Interactive's Customer Support web site for updates to LoadRunner help files.

Technical Support Online uses your default Web browser to open Mercury Interactive's Customer Support Web site. This site enables you to browse the knowledge base and add your own articles, post to and search user discussion forums, submit support requests, download patches and updated documentation, and more. The URL for this Web site is *http://support.mercuryinteractive.com*.

Support Information presents the locations of Mercury Interactive's Customer Support web site and home page, the e-mail address for sending information requests, and a list of Mercury Interactive's offices around the world.

Mercury Interactive on the Web uses your default web browser to open Mercury Interactive's home page. The URL for this Web site is *http://www.mercuryinteractive.com*.

LoadRunner Documentation Set

LoadRunner is supplied with a set of documentation that describes how to:

- install LoadRunner
- create Vuser scripts
- use the LoadRunner Controller
- use the LoadRunner Analysis

Using the LoadRunner Documentation Set

The LoadRunner documentation set consists of one installation guide, a Controller user's guide, an Analysis user's guide, and three guides for creating Virtual User scripts.

Installation Guide

For instructions on installing LoadRunner, refer to the *LoadRunner Installation Guide*. The installation guide explains how to install:

- the LoadRunner Controller—on a Windows-based machine
- Virtual User components—for both Windows and UNIX platforms

Controller User's Guide

The LoadRunner documentation pack includes one Controller user's guide:

The *LoadRunner Controller User's Guide (Windows)* describes how to create and run LoadRunner scenarios using the LoadRunner Controller in a Windows environment. The Vusers can run on UNIX and Windows-based platforms. The Controller user's guide presents an overview of the LoadRunner testing process.

Analysis User's Guide

The LoadRunner documentation pack includes one Analysis user's guide:

The *LoadRunner Analysis User's Guide* describes how to use the LoadRunner Analysis graphs and reports after running a scenario in order to analyze system performance.

Guides for Creating Vuser Scripts

The LoadRunner documentation pack has two guides that describe how to create Vuser scripts:

- The *Creating Vuser Scripts* guide describes how to create all types of Vuser scripts. When necessary, supplement this document with the *LoadRunner Online Function Reference* and the following guide.
- The *WinRunner User's Guide* describes in detail how to use WinRunner to create GUI Vuser scripts. The resulting Vuser scripts run on Windows platforms. The *TSL Online Reference* should be used in conjunction with this document.

For information on	Look here...
Installing LoadRunner	*LoadRunner Installation Guide*
The LoadRunner testing process	*LoadRunner Controller User's Guide (Windows)*
Creating Vuser scripts	*Creating Vuser Scripts* guide
Creating and running scenarios	*LoadRunner Controller User's Guide (Windows)*
Analyzing test results	*LoadRunner Analysis User's Guide*

Typographical Conventions

This book uses the following typographical conventions:

1, 2, 3	Bold numbers indicate steps in a procedure.
➤	Bullets indicate options and features.
>	The greater than sign separates menu levels (for example, **File** > **Open**).
Stone Sans	The **Stone Sans** font indicates names of interface elements in a procedure (for example, "Click the **Run** button.").
Italics	*Italic* text indicates names (for example, names of variables or books).
Helvetica	The Helvetica font is used for examples and strings that are to be typed in literally.
<>	Angle brackets enclose a part of a URL address that needs to be typed in.
[ ]	Square brackets enclose optional parameters.
...	In a line of syntax, an ellipsis indicates that more items of the same format may be included.

Part I

Understanding LoadRunner

1

Introduction

To load test your application, LoadRunner emulates an environment where multiple users work concurrently. While the application is under load, LoadRunner accurately measures, monitors, and analyzes a system's performance and functionality.

Application Load Testing

Modern system architectures are complex. While they provide an unprecedented degree of power and flexibility, these systems are difficult to test. Whereas single-user testing focuses primarily on functionality and the user interface of a system component, application testing focuses on performance and reliability of an entire system.

For example, a typical application testing scenario might depict 1000 users that log in simultaneously to a system on Monday morning: What is the response time of the system? Does the system crash? To be able to answer these questions—and more—a complete application performance testing solution must:

- test a system that combines a variety of software applications and hardware platforms
- determine the suitability of a server for any given application
- test the server before the necessary client software has been developed
- emulate an environment where multiple clients interact with a single server application
- test an application under the load of tens, hundreds, or even thousands of potential users

Manual Testing Limitations

Traditional or manual testing methods offer only a partial solution to load testing. For example, you can test an entire system manually by constructing an environment where many users work simultaneously on the system. Each user works at a single machine and submits input to the system. However, this manual testing method has the following drawbacks:

- it is expensive, requiring large amounts of both personnel and machinery
- it is complicated, especially coordinating and synchronizing multiple testers
- it involves a high degree of organization, especially to record and analyze results meaningfully
- the repeatability of the manual tests is limited

The LoadRunner Solution

The LoadRunner automated solution addresses the drawbacks of manual performance testing:

- LoadRunner reduces the personnel requirements by replacing human users with virtual users or *Vusers*. These Vusers emulate the behavior of real users—operating real applications.
- Because numerous Vusers can run on a single computer, LoadRunner reduces the hardware requirements.
- The LoadRunner Controller allows you to easily and effectively control all the Vusers—from a single point of control.
- LoadRunner monitors the application performance online, enabling you to fine-tune your system during test execution.
- LoadRunner automatically records the performance of the application during a test. You can choose from a wide variety of graphs and reports to view the performance data.

- LoadRunner checks where performance delays occur: network or client delays, CPU performance, I/O delays, database locking, or other issues at the database server. LoadRunner monitors the network and server resources to help you improve performance.
- Because LoadRunner tests are fully automated, you can easily repeat them as often as you need.

Using LoadRunner

Scenarios

Using LoadRunner, you divide your application performance testing requirements into *scenarios*. A scenario defines the events that occur during each testing session. Thus, for example, a scenario defines and controls the number of users to emulate, the actions that they perform, and the machines on which they run their emulations.

Vusers

In the scenario, LoadRunner replaces human users with *virtual users* or *Vusers*. When you run a scenario, Vusers emulate the actions of human users working with your application. While a workstation accommodates only a single human user, many Vusers can run concurrently on a single workstation. In fact, a scenario can contain tens, hundreds, or even thousands of Vusers.

Vuser Scripts

The actions that a Vuser performs during the scenario are described in a Vuser script. When you run a scenario, each Vuser executes a *Vuser script*. The Vuser scripts include functions that measure and record the performance of your application's components.

Transactions

To measure the performance of the server, you define *transactions*. A transaction represents an action or a set of actions that you are interested in measuring. You define transactions within your Vuser script by enclosing the appropriate sections of the script with *start* and *end* transaction statements. For example, you can define a transaction that measures the time it takes for the server to process a request to view the balance of an account and for the information to be displayed at the ATM.

Rendezvous points

You insert *rendezvous points* into Vuser scripts to emulate heavy user load on the server. *Rendezvous points* instruct Vusers to wait during test execution for multiple Vusers to arrive at a certain point, in order that they may simultaneously perform a task. For example, to emulate peak load on the

bank server, you can insert a rendezvous point instructing 100 Vusers to deposit cash into their accounts at the same time.

Controller

You use the *LoadRunner Controller* to manage and maintain your scenarios. Using the Controller, you control all the Vusers in a scenario from a single workstation.

Load generator

When you execute a scenario, the LoadRunner Controller distributes each Vuser in the scenario to a *load generator*. The load generator is the machine that executes the Vuser script, enabling the Vuser to emulate the actions of a human user.

Performance analysis

Vuser scripts include functions that measure and record system performance during load-testing sessions. During a scenario run, you can monitor the network and server resources. Following a scenario run, you can view performance analysis data in reports and graphs.

Working with LoadRunner

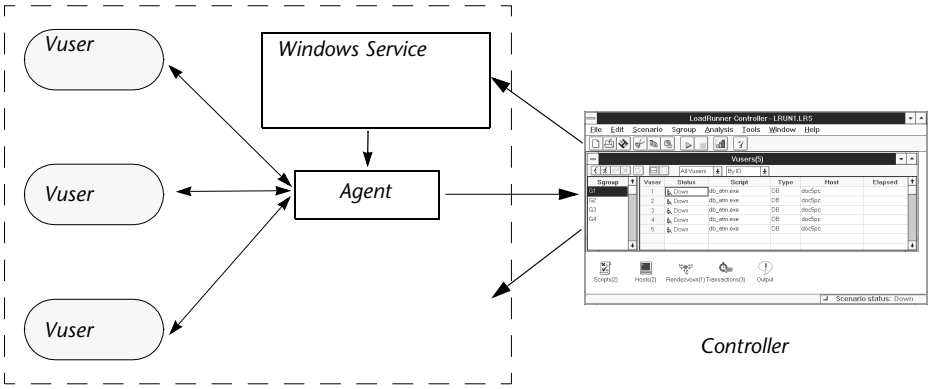
Suppose you want to test an online banking Web server that is accessed by many Internet users. The Web site provides a full range of banking services to the customers—such as the ability to transfer funds and check account balances. To test this server, you create a scenario. The scenario defines the actions that are performed on the server during the load test.

During the scenario that loads and monitors the bank server, you want to:

- emulate conditions of controlled load on the server
- emulate conditions of maximum load on the server
- measure server performance under load
- check where performance delays occur: network or client delays, CPU performance, I/O delays, database locking, or other issues at the server
- monitor the network and server resources under load

LoadRunner Vuser Technology

On each Windows load generator, you install the *Windows Service* and an *Agent*.



Windows Service

The Windows Service enables the Controller to start applications on the load generator machine.

Agent

The Agent enables the Controller and the load generator to communicate with each other. When you run a scenario, the Controller instructs the *Windows Service* to launch the LoadRunner *Agent*. The Agent receives instructions from the Controller to initialize, run, pause, and abort Vusers. At the same time, the Agent also relays data on the status of the Vusers back to the Controller.

LoadRunner Vuser Types

LoadRunner has various types of Vusers. Each type is designed to handle different aspects of today's system architectures. You can use the Vuser types in any combination in a scenario in order to create a comprehensive application test. The following Vuser types are available:

- **E-business**

  For FTP, LDAP, multi-protocol Web/Ws, Palm, POP3, Real, and Web (HTTP/HTML).

- **Wireless**

  For i-mode, VoiceXML, and WAP.

- **Enterprise Java Beans**

  For EJB Testing and RMI-Java.

- **Distributed Components**

  For COM/DCOM,CORBA-Java, and RMI-Java.

- **Middleware**

  For Jolt and Tuxedo protocols.

- **ERP**

  For SAP, Baan, Oracle NCA, Peoplesoft (Tuxedo or Web) and Siebel (DB2 CLI, Oracle, or MSSQL) protocols.

- **Client/Server**

  For DB2 CLI, Informix, Jolt, MSSQLServer, ODBC, Oracle (2-tier), Sybase Ctlib, Sybase Dblib, and Windows Sockets protocols.

- **Legacy**

  For Terminal Emulation (RTE).

- **Custom**

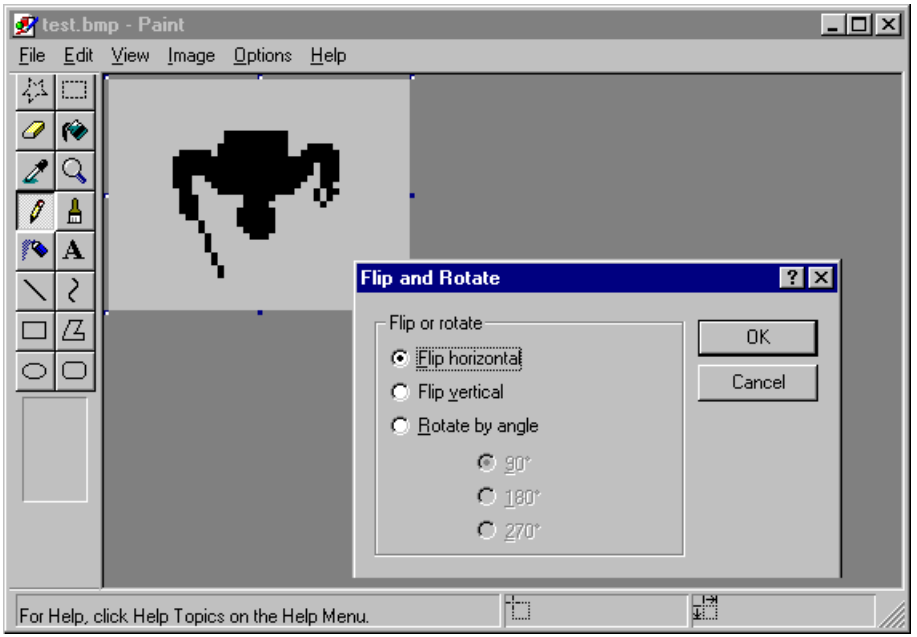
  For C Vuser, Java Script Vuser, Java Vuser, VB Script Vuser, VB Vuser.

GUI Vusers

GUI Vusers

GUI Vusers operate graphical user interface (GUI) applications. These applications can run in a Microsoft Windows environment. Each GUI Vuser that you develop emulates a real user by submitting input to, and receiving output from, GUI applications. For example, a GUI Vuser could operate Microsoft Paint as follows:

1. Select Open from the File menu.
2. Select a graphic file called test.bmp.
3. Click the Open button.
4. Select Flip/Rotate from the Image menu.
5. Click the Flip Horizontal radio button.
6. Click the OK button.
7. Select Save from the File menu.



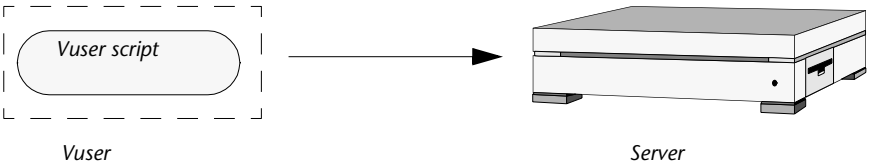
The operations that a GUI Vuser performs on an application are defined in a GUI Vuser script. You create GUI Vuser scripts using Mercury Interactive's

GUI testing tools: WinRunner (for Microsoft Windows applications), and Astra QuickTest (for Web applications).

You can run only a single GUI Vuser on a Windows-based load generator. Use Citrix to run multiple GUI Vusers. Refer to the Readme file for additional information about configuring your load generators using Cirtix. For additional information on Windows-based GUI Vusers, refer to the *Creating Vuser Scripts* guide.

Vuser Technology

Vusers (except for GUI and RTE Vusers) generate load on a server by submitting input directly to the server. Vusers do not operate client applications—they access the server using LoadRunner API functions. These API functions emulate the input from an actual application.



Because Vusers are not reliant on client software, you can use Vusers to test server performance even before the client software has been developed. Further, since Vusers do not have a user interface, the amount of system resources required is minimal. This allows you to run large numbers of Vusers on a single workstation.

The following example illustrates the use of Vusers: Suppose that you have a Web-based database server that maintains your customer information. The information is accessed by numerous customer service personnel who are located throughout the country. The server receives the queries, processes the requests, and returns responses, via the Web, to field personnel.

You want to test the response times of the entire system when numerous service personnel simultaneously access the server. Using LoadRunner, you could create several hundred Vusers, each Vuser accessing the server database. The Vusers enable you to emulate and measure the performance of your database and Web servers under the load of many users.

You develop a Vuser script to define the actions of a Vuser. A Vuser script includes functions that control the script execution, specify the input that the Vuser submits to the server, and measure the server performance.

You develop Vuser scripts either by recording with LoadRunner's Vuser Script Generator (VuGen) or by using LoadRunner's Vuser script templates.

For the database server example above, you could create a Vuser script that performs the following actions:

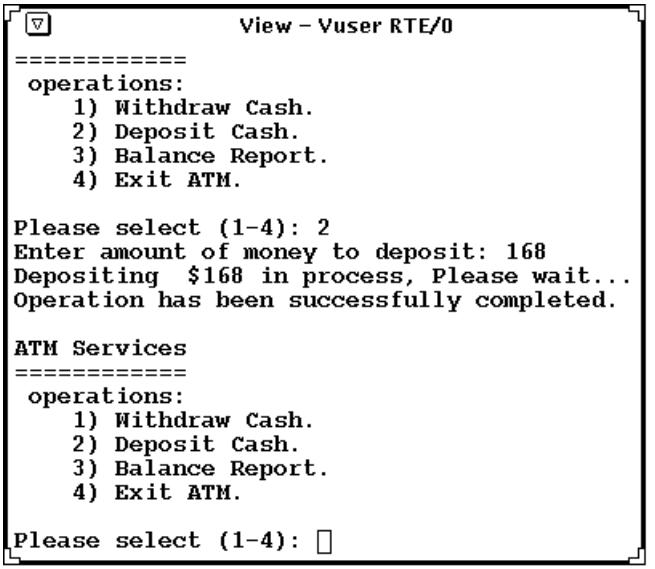
- logs in to the Web application
- connects to the database server
- submits an SQL query
- retrieves and processes the server response
- disconnects from the server and the Web

You can create Vuser scripts on a Windows-based platform, or program them on a UNIX platform. For a list of the supported UNIX platforms, see the LoadRunner Readme file. For more information about Vusers, refer to the *Creating Vuser Scripts* guide.

RTE Vusers

RTE Vusers

RTE Vusers operate character-based applications. Each RTE Vuser that you develop emulates a real user by submitting input to, and receiving output from, character-based applications.

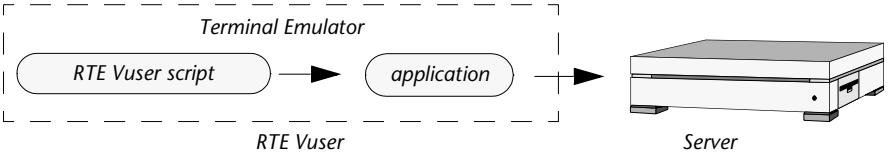


The following example illustrates the use of RTE Vusers: Suppose that you have a database server that maintains customer information. The information is accessed by numerous field service representatives who are located throughout the country. Every time a field service representative makes a repair, he accesses the server database by modem. Using a character-based application, the service representative records the customer complaint and accesses additional information about the customer.

You want to test the response times of the server when many service personnel simultaneously access the server. Using LoadRunner, you could create several hundred RTE Vusers, each Vuser accessing the server database using a character-based application. The RTE Vusers enable you to emulate and measure the performance of your server under the load of many users.

The operations that an RTE Vuser performs on an application are defined in an RTE Vuser script. You create RTE Vuser scripts by using the Vuser Script

Generator (VuGen). The generator enables you to record the actions that you perform on a character-based application.

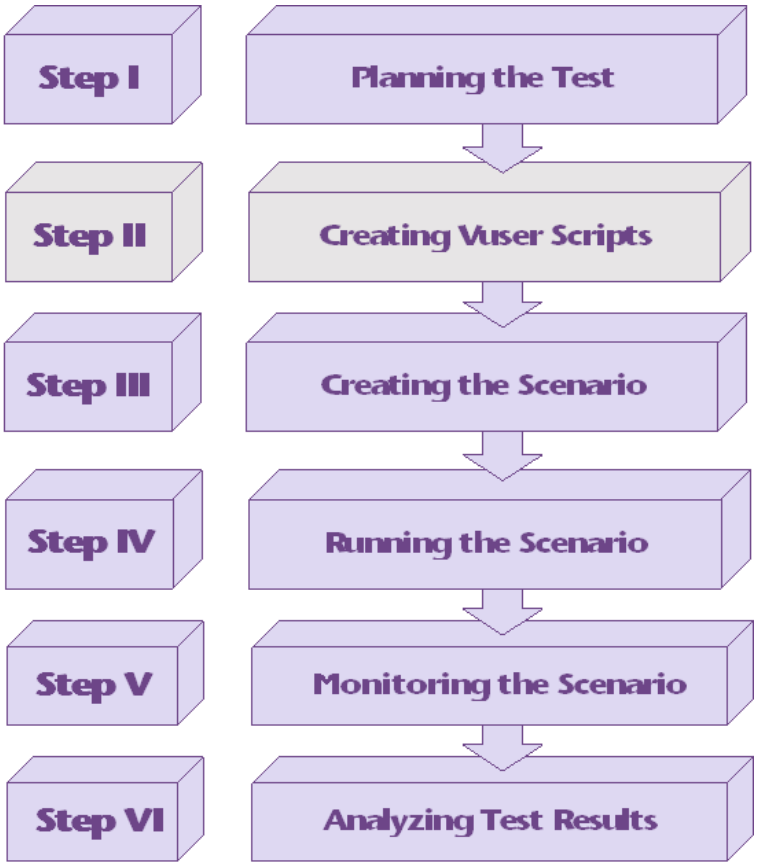


For further information on RTE Vusers, refer to the *Creating Vuser Scripts* guide.

2

The LoadRunner Testing Process

You can easily create and run load-test scenarios by following the LoadRunner testing process below. The following illustration outlines the testing process:



This chapter gives you an overview of LoadRunner's six–step process for testing your Web-based application under load.

Step I: Planning the Test

Successful load testing requires that you develop a thorough test plan. A clearly defined test plan will ensure that the LoadRunner scenarios that you develop will accomplish your load testing objectives. For more information, see Chapter 3, "Load Test Planning."

Step II: Creating the Vuser Scripts

Vusers emulate human users interacting with your Web-based application. A Vuser script contains the actions that each virtual user performs during scenario execution.

In each Vuser script you determine the tasks that will be:

- performed by each Vuser
- performed simultaneously by multiple Vusers
- measured as transactions

For more information on creating Vuser scripts, refer to the *Creating Vuser Scripts* guide.

Step III: Creating the Scenario

A scenario describes the events that occur during a testing session. A scenario includes a list of machines on which Vusers run, a list of scripts that the Vusers run, and a specified number of Vusers or Vuser groups that run during the scenario. You create scenarios using the LoadRunner Controller. For an introduction to the Controller, see Chapter 4, "The LoadRunner Controller at a Glance."

Creating a Manual Scenario

You create a scenario by defining Vuser groups to which you assign a quantity of individual Vusers, Vuser scripts, and load generators to run the scripts. For instructions on creating a manual scenario, see Chapter 5, "Creating a Manual Scenario."

Creating a Goal-Oriented Scenario

For Web tests, you can create a goal-oriented scenario, in which you define the goals you want your test to achieve. LoadRunner automatically builds a scenario for you, based on these goals. For instructions on creating a goal-oriented scenario, see Chapter 7, "Creating a Goal-Oriented Scenario."

Step IV: Running the Scenario

You emulate user load on the server by instructing multiple Vusers to perform tasks simultaneously. You can set the level of load by increasing and decreasing the number of Vusers that perform tasks at the same time. For more information, see Chapter 8, "Using Rendezvous Points."

Before you run a scenario, you set the scenario configuration and scheduling. This determines how all the load generators and Vusers behave when you run the scenario. For more information, see Chapter 9, "Configuring a Scenario" and Chapter 6, "Scheduling a Manual Scenario."

You can run the entire scenario, groups of Vusers (Vuser groups), or individual Vusers. While a scenario runs, LoadRunner measures and records the transactions that you defined in each Vuser script. You can also monitor your system's performance online. For more information, see Part III, "Executing a Scenario."

Step V: Monitoring a Scenario

You can monitor scenario execution using the LoadRunner online run-time, transaction, system resource, Web resource, Web server resource, Web application server resource, database server resource, network delay, streaming media resource, and firewall server monitors. For more information, see Part IV, "Monitoring a Scenario."

Step VI: Analyzing Test Results

During scenario execution, LoadRunner records the performance of the application under different loads. You use LoadRunner's graphs and reports to analyze the application's performance. For more information about LoadRunner's reports and graphs, see the *LoadRunner Analysis User's Guide*.

3

Load Test Planning

Developing a comprehensive test plan is a key to successful load testing. A clearly defined test plan ensures that the LoadRunner scenarios you develop will accomplish your load testing objectives.

This chapter introduces the load test planning process:

- Analyzing the Application
- Defining Testing Objectives
- Planning LoadRunner Implementation
- Examining Load Testing Objectives

About Load Test Planning

As in any type of system testing, a well-defined test plan is the first essential step to successful testing. Planning your load testing helps you to:

- Build test scenarios that accurately emulate your working environment.

  Load testing means testing your application under typical working conditions, and checking for system performance, reliability, capacity, etc.

- Understand which resources are required for testing.

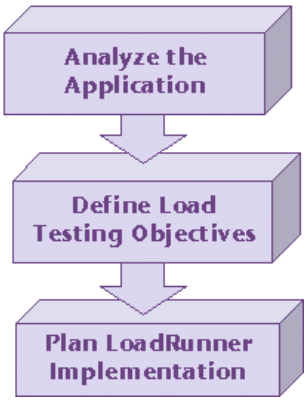
  Application testing requires hardware, software, and human resources. Before you begin testing, you should know which resources are available and decide how to use them effectively.

- Define success criteria in measurable terms.

  Focused testing goals and test criteria ensure successful testing. For example, it's not enough to define vague objectives like "Check server response time

under heavy load." A more focused success criteria would be "Check that 50 customers can check their account balance simultaneously, and that the server response time will not exceed one minute."

Load test planning is a three-step process:



Analyzing the Application

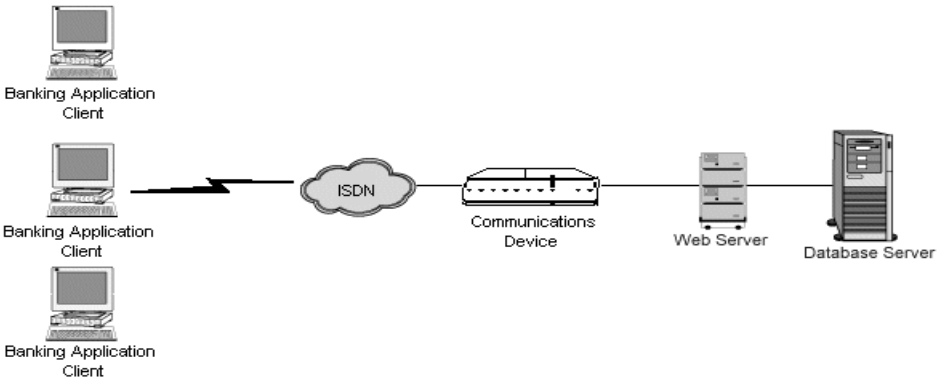
The first step to load test planning is analyzing your application. You should become thoroughly familiar with the hardware and software components, the system configuration, and the typical usage model. This analysis ensures that the testing environment you create using LoadRunner will accurately reflect the environment and configuration of the application under test.

Identifying System Components

Draw a schematic diagram to illustrate the structure of the application. If possible, extract a schematic diagram from existing documentation. If the application under test is part of a larger network system, you should identify the component of the system to be tested. Make sure the diagram includes all system components, such as client machines, network, middleware, and servers.

The following diagram illustrates an online banking system which is accessed by many Web users. The Web users each connect to the same

database to transfer funds and check balances. The customers connect to the database server through the Web, using multiple browsers.



Describing the System Configuration

Enhance the schematic diagram with more specific details. Describe the configuration of each system component. You should be able to answer the following questions:

- How many users are anticipated to connect to the system?
- What is the application client's machine configuration? (hardware, memory, operating system, software, development tool, etc.)
- What types of Database and Web servers are used? (hardware, database type, operating system, file server etc.)
- How does the server communicate with the application client?
- What is the middleware configuration and application server between the front-end client and back-end server?
- What other network components may affect response time? (modems etc.)
- What is the throughput of the communications devices? How many concurrent users can each device handle?

For example, the schematic diagram above specified that there are multiple application clients accessing the system.

Front-End Client Configuration	
Anticipated number of application clients	50 concurrent application clients
Hardware / Memory	586 / 32MB
Operating system & version	Windows NT 4.0
Client browser	Internet Explorer 4.0

Analyzing the Usage Model

Define how the system is typically used, and decide which functions are important to test. Consider who uses the system, the number of each type of user, and each user's common tasks. In addition, consider any background load that might affect the system response time.

For example, suppose 200 employees log on to the accounting system every morning, and the same office network has a constant background load of 50 users performing various word processing and printing tasks. You could create a LoadRunner scenario with 200 virtual users signing in to the accounting database, and check the server response time.

To check how background load affects the response time, you could run your scenario on a network where you also simulate the load of employees performing word processing and printing activities.

Task Distribution

In addition to defining the common user tasks, examine the distribution of these tasks. For example, suppose the bank uses a central database to serve clients across many states and time zones. The 250 application clients are located in two different time zones, all connecting to the same Web server. There are 150 in Chicago and 100 in Detroit. Each begins their business day at 9:00 AM, but since they are in different time zones, there should never be more than 150 users signing in at any given time.

You can analyze task distribution to determine when there is peak database activity, and which activities typically occur during *peak load* time.

Defining Testing Objectives

Before you begin testing, you should define exactly what you want to accomplish.

Following are common application testing objectives that LoadRunner helps you test, as described in Robert W. Buchanan, Jr's *The Art of Testing Network Systems* (John Wiley & Sons, Inc., 1996).

Objective	Answers the Question
Measuring end-user response time	How long does it take to complete a business process?
Defining optimal hardware configuration	Which hardware configuration provides the best performance?
Checking reliability	How hard or long can the system work without errors or failures?
Checking hardware or software upgrades	How does the upgrade affect performance or reliability?
Evaluating new products	Which server hardware or software should you choose?
Measuring system capacity	How much load can the system handle without significant performance degradation?
Identifying bottlenecks	Which element is slowing down response time?

A more detailed description of each objective appears at the end of this chapter.

Stating Objectives in Measurable Terms

Once you decide on your general load testing objectives, you should provide more focused goals by stating your objectives in measurable terms. To provide a baseline for evaluation, determine exactly what constitutes acceptable and unacceptable test results.

For example:

General Objective - Product Evaluation: choose hardware for the Web server.

Focused Objective - Product Evaluation: run the same group of 300 virtual users on two different servers, HP and NEC. When all 300 users simultaneously browse the pages of your Web application, determine which hardware gives a better response time.

Deciding When to Test

Load testing is necessary throughout the product life cycle. The following table illustrates what types of tests are relevant for each phase of the product life cycle:

Planning and Design	Development	Deployment	Production	Evolution
Evaluate new products	Measure response time	Check reliability	Measure response time	Check HW or SW upgrades
Measure response time	Check optimal hardware configuration	Measure response time	Identify bottlenecks	Measure system capacity
	Check HW or SW upgrades	Measure system capacity		
	Check reliability			

Planning LoadRunner Implementation

The next step is to decide how to use LoadRunner to achieve your testing goals.

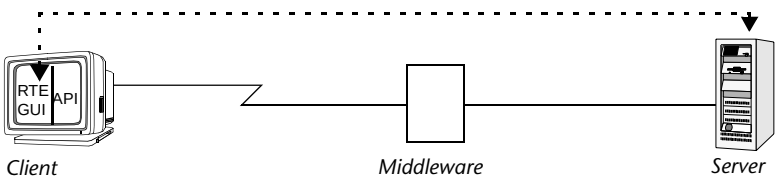
Defining the Scope of Performance Measurements

You can use LoadRunner to measure response time at different points in the application. Determine where to run the Vusers and which Vusers to run according to the test objectives:

➤ Measuring end-to-end response time:

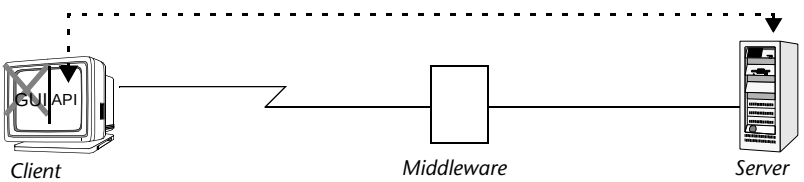
You can measure the response time that a typical user experiences by running a GUI Vuser or RTE Vuser at the front end. GUI Vusers emulate real users by submitting input to and receiving output from the client application; RTE Vusers emulate real users submitting input to and receiving output from a character-based application.

You can run GUI or RTE Vusers at the front end to measure the response time across the entire network, including a terminal emulator or GUI front end, network, and server.



➤ Measuring network and server response times:

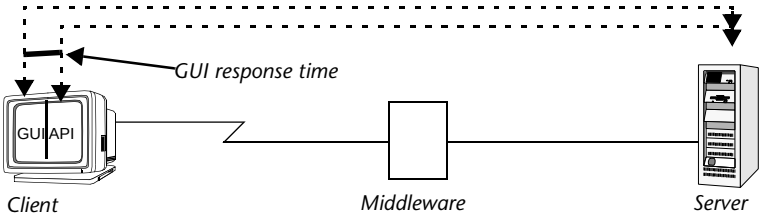
You can measure network and server response time, excluding response time of the GUI front end, by running Vusers (not GUI or RTE) on the client machine. Vusers emulate client calls to the server without the user interface. When you run many Vusers from the client machine, you can measure how the load affects network and server response time.



➤ Measuring GUI response time:

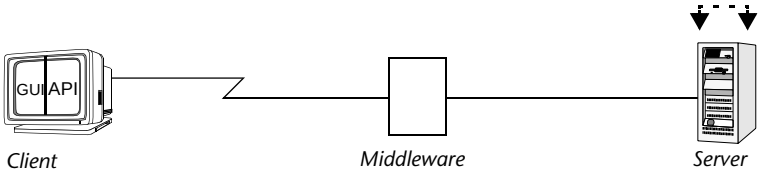
You can determine how the client application interface affects response time by subtracting the previous two measurements:

GUI response time = end-to-end - network and server



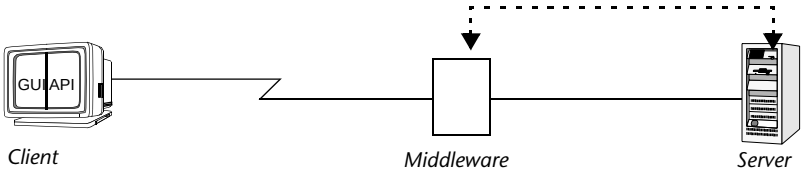
➤ Measuring server response time:

You can measure the time it takes for the server to respond to a request without going across the network. When you run Vusers on a machine directly connected to the server, you can measure server performance.



➤ Measuring middleware-to-server response time:

You can measure response time from the server to middleware if you have access to the middleware and its API. You can create Vusers with the middleware API and measure the middleware-server performance.



Defining Vuser Activities

Create Vuser scripts based on your analysis of Vuser types, their typical tasks and your test objectives. Since Vusers emulate the actions of a typical end-user, the Vuser scripts should include the typical end-user tasks. For example, to emulate an online banking client, you should create a Vuser script that performs typical banking tasks. You would browse the pages that you normally visit to transfer funds or check balances.

You decide which tasks to measure based on your test objectives and define *transactions* for these tasks. Transactions measure the time that it takes for the server to respond to tasks submitted by Vusers (end-to-end time). For example, to check the response time of a bank Web server supplying an account balance, define a transaction for this task in the Vuser script.

In addition, you can emulate peak activity by using *rendezvous points* in your script. Rendezvous points instruct multiple Vusers to perform tasks at exactly the same time. For example, you can define a rendezvous to emulate 70 users simultaneously updating account information.

Selecting Vusers

Before you decide on the hardware configuration to use for testing, determine the number and type of Vusers required. To decide how many Vusers and which types to run, look at the typical usage model, combined with the testing objectives. Some general guidelines are:

- Use one or a few GUI users to emulate each type of typical user connection.
- Use RTE Vusers to emulate terminal users.
- Run multiple non-GUI or non-RTE Vusers to generate the rest of the load for each user type.

For example, suppose that you have five kinds of users, each performing a different business process:

Usage Model	GUI	RTE	Other
100 customer service users in New York (LAN connection)	2	_	98
30 customers in Europe (dial-in ISDN connection)	2	_	28
5 background batch processes	_	_	5
150 customers (terminal connection)	_	150	_
6 managers (2 users with 486 PCs, 4 with 586 PCs)	1 (486 PC) 1 (586 PC)	_	4

Choosing Testing Hardware/Software

The hardware and software should be powerful and fast enough to emulate the required number of virtual users.

To decide on the number of machines and correct configuration, consider the following:

- It is advisable to run the LoadRunner Controller on a separate machine.
- Each GUI Vuser requires a separate Windows-based machine; several GUI Vusers can run on a single UNIX machine.
- Configuration of the test machine for GUI Vusers should be as similar as possible to the actual user's machine.

Refer to the following tables to estimate the required hardware for each LoadRunner testing component. These requirements are for optimal performance.

Windows Configuration Requirements

Requirement	Controller with On-line Monitors	Virtual Vuser Generator	Virtual Users	Analysis Module
Computer/ Processor	Pentium 300 MHZ or higher	Pentium 300 MHZ or higher	Pentium 500 MHZ CPU or higher	Pentium 300 MHZ or higher
Operating System	Windows NT service pack 3 or higher Windows 2000	Windows NT service pack 3 or higher Windows 2000	Windows NT service pack 3 or higher Windows 2000	Windows 2000
Memory	64 MB or more	128 MB or more	At least 1 MB RAM for non-multithreaded Vuser or at least 400 KB multi-threaded Vuser	128 MB or more
Swap Space	Two times the total physical memory	Two times the total physical memory	Two times the total physical memory	Two times the total physical memory
Free Hard Disk Space	200 MB	200 MB	Minimum 500 MB	200 MB
Browser	IE 4.x or higher Netscape Navigator 4.x	IE 4.x or higher Netscape Navigator 4.x	N/A	IE 4.x or higher Netscape Navigator 4.x

---

Note: The results file requires a few MB of disk space for a long scenario run with many transactions. The load generator machines also require a few MB of disk space for temporary files if there is no NFS. See Chapter 9, "Configuring a Scenario" for more information about run-time file storage.

UNIX Configuration Requirements

Requirement	GUI Vuser (per user)	Vuser (per user)	Web Vuser (per user)
Memory	4-5 MB plus client application requirements	At least 1.5 MB (depends on application)	~0.5 MB
Swap Space	Four times the total physical memory	Four times the total physical memory	Two times the total physical memory
Disk Space	n/a	n/a	n/a
No. of Processes	4	1	1
No. of pty's	n/a	n/a	n/a
1 CPU supports *x* users	30-50 or more	200-300 or more	300-400 or more

The results file requires a few MB of disk space for a long scenario run with many transactions. The load generator machines also require a few MB of disk space for temporary files if there is no NFS. Refer to Chapter 9, "Configuring a Scenario" for more information about run-time file storage.

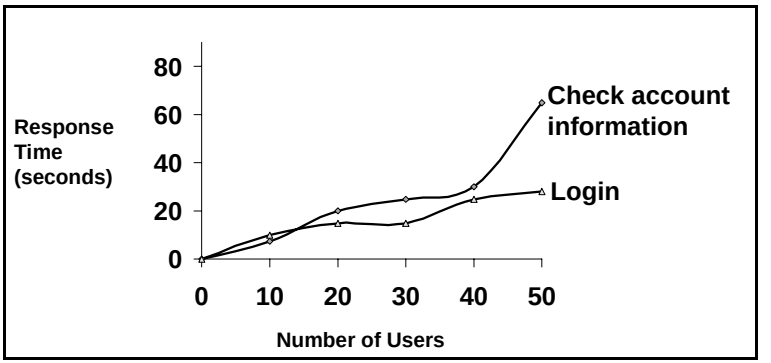
Examining Load Testing Objectives

Your test plan should be based on a clearly defined testing objective. This section presents an overview of common testing objectives:

- Measuring end-user response time
- Defining optimal hardware configuration
- Checking reliability
- Checking hardware or software upgrades
- Evaluating new products
- Identifying bottlenecks
- Measuring system capacity

Measuring End-user Response Time

Check how long it takes for the user to perform a business process and receive a response from the server. For example, suppose that you want to verify that while your system operates under normal load conditions, the end users receive responses to all requests within 20 seconds. The following graph presents a sample load vs. response time measurement for a banking application:



Defining Optimal Hardware Configuration

Check how various system configurations (memory, CPU speed, cache, adaptors, modems) affect performance. Once you understand the system architecture and have tested the application response time, you can measure the application response for different system configurations to determine which settings provide the desired performance levels.

For example, you could set up three different server configurations and run the same tests on each configuration to measure performance variations:

- Configuration 1: 200MHz, 64MB RAM
- Configuration 2: 200MHz, 128MB RAM
- Configuration 3: 266MHz, 128MB RAM

Checking Reliability

Determine the level of system stability under heavy or continuous work loads. You can use LoadRunner to create stress on the system: force the system to handle extended activity in a compressed time period to simulate the kind of activity a system would normally experience over a period of weeks or months.

Checking Hardware or Software Upgrades

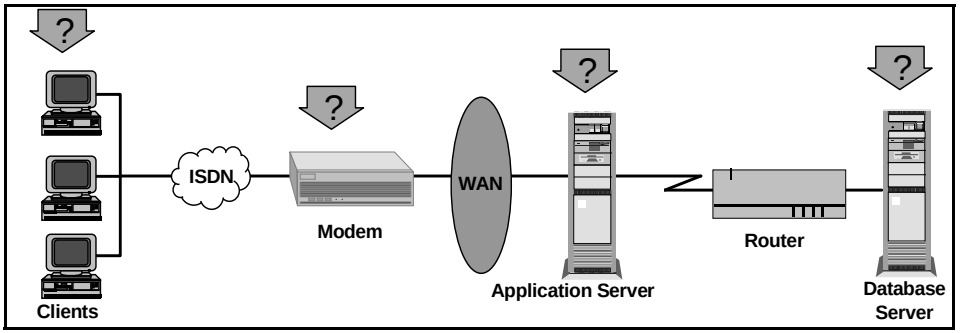
Perform regression testing to compare a new release of hardware or software to an older release. You can check how an upgrade affects response time (benchmark) and reliability. Application regression testing does not check new features of an upgrade; rather it checks that the new release is as efficient and reliable as the older release.

Evaluating New Products

You can run tests to evaluate individual products and subsystems during the planning and design stage of a product's life cycle. For example, you can choose the hardware for the server machine or the database package based on evaluation tests.

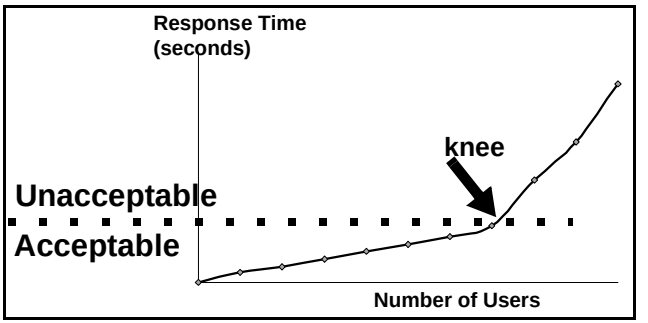
Identifying Bottlenecks

You can run tests which identify bottlenecks on the system to determine which element is causing performance degradation, for example, file locking, resource contention and network overload. Use LoadRunner in conjunction with the new network and machine monitoring tools to create load and measure performance at different points in the system. For more information, see Part IV, "Monitoring a Scenario.".



Measuring System Capacity

Measure how much excess capacity the system can handle without performance degradation. To check capacity, you can compare performance versus load on the existing system, and determine where significant response-time degradation begins to occur. This is often called the "knee" of the response time curve.



Once you determine the current capacity, you can decide if resources need to be increased to support additional users.

4

The LoadRunner Controller at a Glance

This chapter introduces the Controller window and explains how to perform basic scenario operations.

This chapter describes:

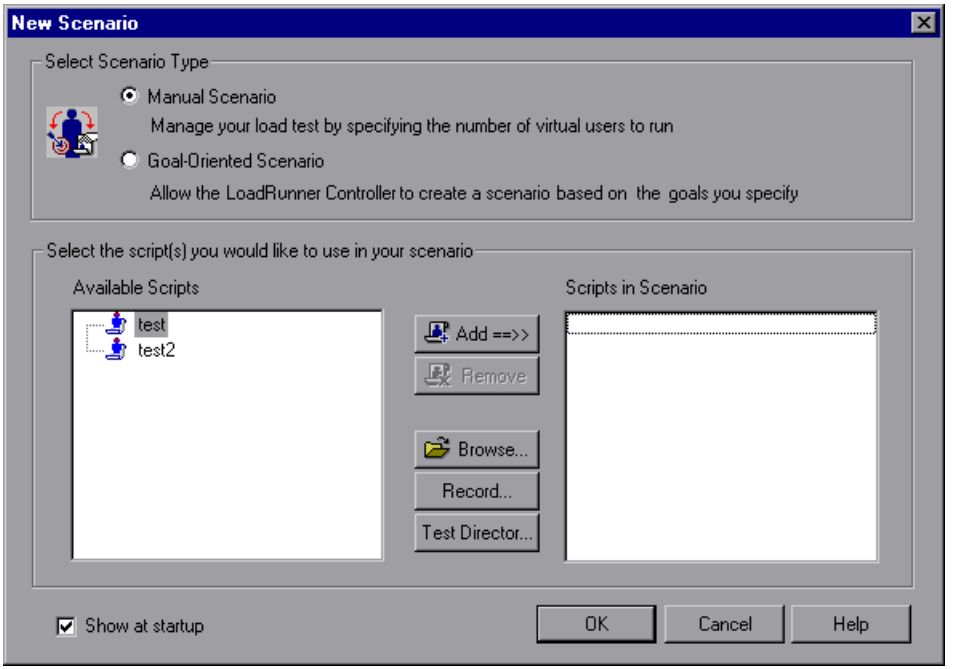
- Opening the Controller
- Introducing the LoadRunner Controller
- Managing Scenario Files
- Running a Scenario

Opening the Controller

Set up the LoadRunner environment according to the instructions in the *LoadRunner Installation Guide*.

To open the Controller:

Select **Start** > **Programs** > **LoadRunner** > **Controller**. The Controller opens with the New Scenario dialog box inside.



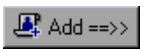
You can select one of two methods to create a scenario: **Manual Scenario** or **Goal-Oriented Scenario.** In a manual scenario, you create a scenario yourself by defining the number of Vusers and Vuser groups you want to run, and building a schedule for LoadRunner to run these groups. In a goal-oriented scenario, you define the goals you want your test to achieve, and LoadRunner automatically builds a scenario for you, based on these goals.

For instructions on creating a manual scenario, see Chapter 5, "Creating a Manual Scenario." For instructions on creating a goal-oriented scenario, see Chapter 7, "Creating a Goal-Oriented Scenario."

To select the script or scripts that you want to use in your scenario:

1 Select a script from the Available Scripts window, or click **Browse** to locate the script you want to use. To select a script saved in the TestDirector database, click the **TestDirector** button. To record a new script using VuGen, click **Record**.

Note: To select a VB Vuser script, browse to locate the .usr file.



2 Click the **Add** button to copy the script you selected to the Scripts in Scenario window.

3 Click the **Remove** button to remove a script from the Scripts in Scenario window.

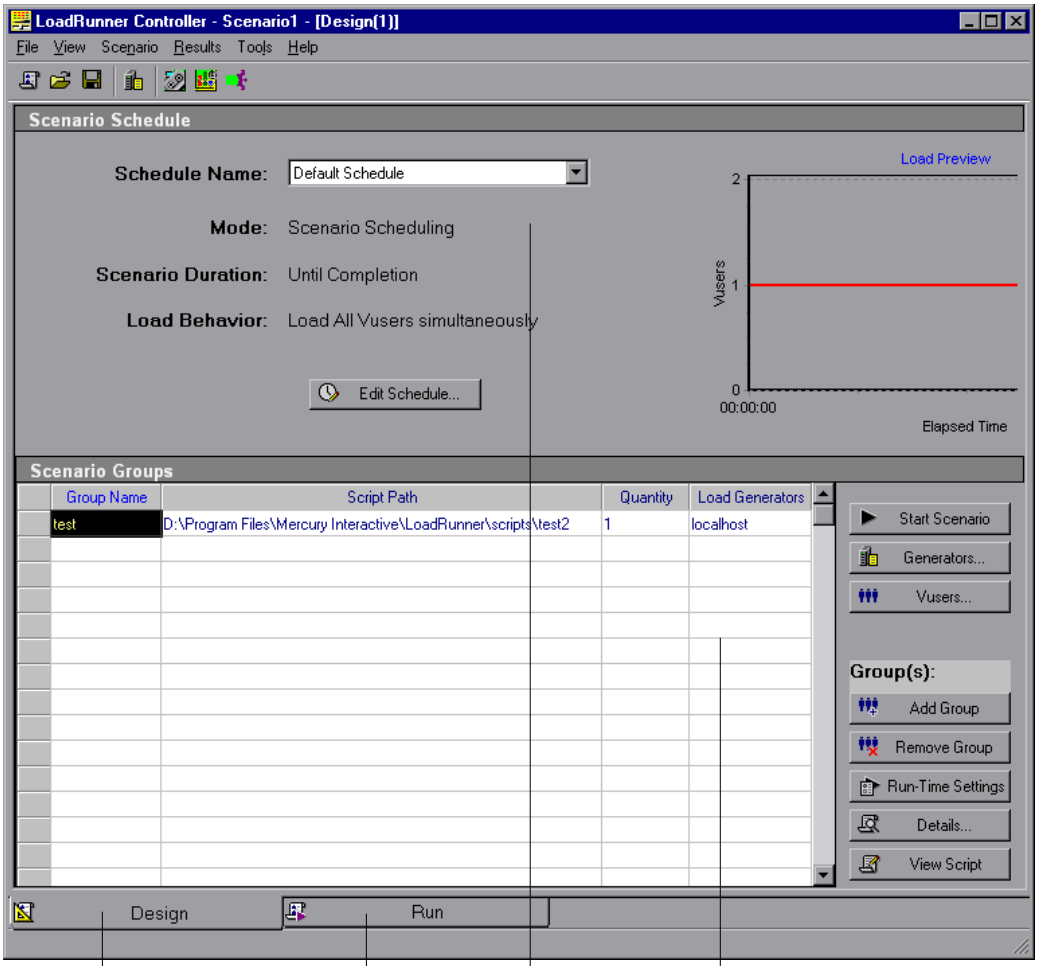
4 To bypass this dialog box the next time you create a new scenario, clear the **Show at startup** check box. You will be able to add scripts later on, while building your scenario.

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

Introducing the LoadRunner Controller

The LoadRunner Controller window contains the following elements:

Title bar	Displays the name of the scenario on which you are currently working.
Menu bar	Displays the menus from which you select commands.
Toolbar	Provides shortcuts for selecting commands. Clicking on a button executes a command.



Design tab

Run tab

Scenario Groups window (Manual Scenario)

Scenario Schedule window (Manual Scenario)

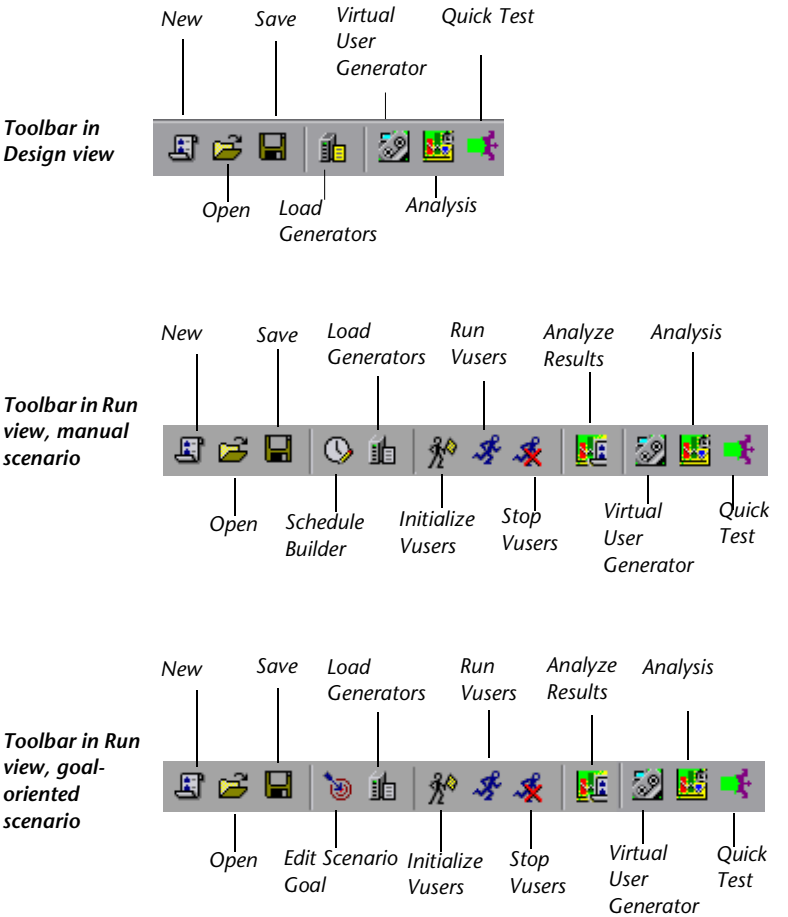
The Controller window has two tabs which correspond to two views:

Design view	In a manual scenario, this view displays a list of all the Vuser groups in a scenario, and the Vuser scripts, load generator machines, and number of Vusers assigned to each group. In a goal-oriented scenario, this view displays a list of all the Vuser scripts in a scenario, and the load generator machines and number of Vusers assigned to each script. This view also displays basic information about the scenario schedule (manual scenario) or goal (goal-oriented scenario).
Run view	Displays information on the running Vusers and Vuser groups, as well as online monitor graphs.

In addition, if you select **View > Show Output**, the Controller opens the **Output** window which displays error and notification messages generated by Vusers during scenario execution.

Choosing Commands from the Toolbar

You can execute many LoadRunner commands by clicking a button on the toolbar in the LoadRunner Controller. There are some variations in the buttons the toolbar displays, depending on whether you are in Design view or Run view, and depending on whether you are creating a manual scenario or a goal-oriented scenario.



Managing Scenario Files

A scenario describes the events that occur during each load testing session. You create a scenario using the Design view of LoadRunner Controller.

After you create the scenario, LoadRunner saves the information in a scenario file (*.lrs*). You use the commands in the File menu to create, open,

save, and close scenario files. Some of these commands are available from the toolbar.

Creating a New Scenario

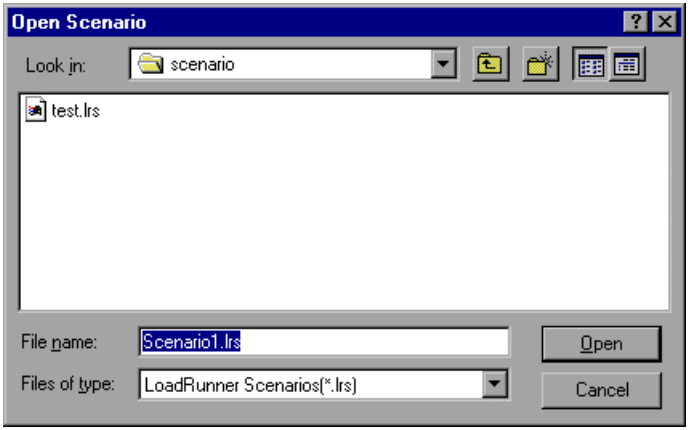
The New command creates a completely new scenario. Note that the New command clears all the information displayed in the Controller windows. To create a new scenario, choose **File** > **New**, or click the **New** button on the Controller toolbar.

Opening an Existing Scenario

The Open command opens any existing scenario.

To open an existing scenario:

1 Choose **File** > **Open**, or click the **Open** button. The Open Scenario dialog box opens.



2 Click a file in the File Name list or type a file name in the File Name box.

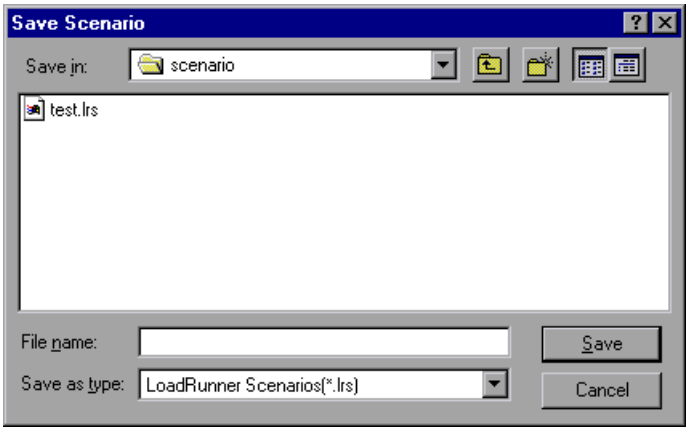
3 Click **Open**. The File Open dialog box closes and the scenario appears in the LoadRunner Controller.

Saving a Scenario

The Save command saves the current scenario.

To save a scenario:

1 Choose **File** > **Save**, or click the **Save** button. The Save Scenario dialog box opens the first time you save a scenario.



2 Type a scenario name in the File Name text box. Note that by default scenario files have the extension *.lrs*.

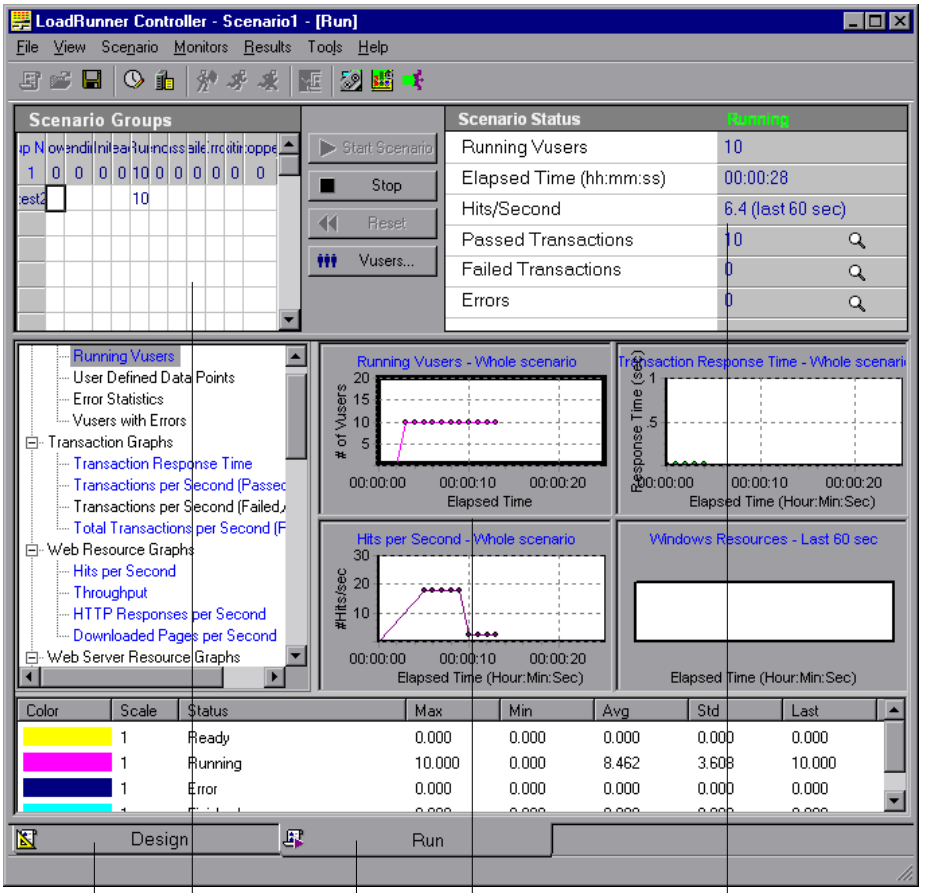
3 Click **Save**. The scenario is saved in the location you specified.

Closing a Scenario

Closing a scenario closes all the Controller windows. To close the scenario, choose **File** > **Close**. If you made changes to the scenario, a Save Changes message appears. Choose **Yes** to save the changes you made. All open windows and icons in the Controller close.

Running a Scenario

Once you have designed your scenario, you are ready to run it. You can control the Vusers and Vuser groups and monitor their performance online using the Run view of the LoadRunner Controller.



Design tab

Run tab

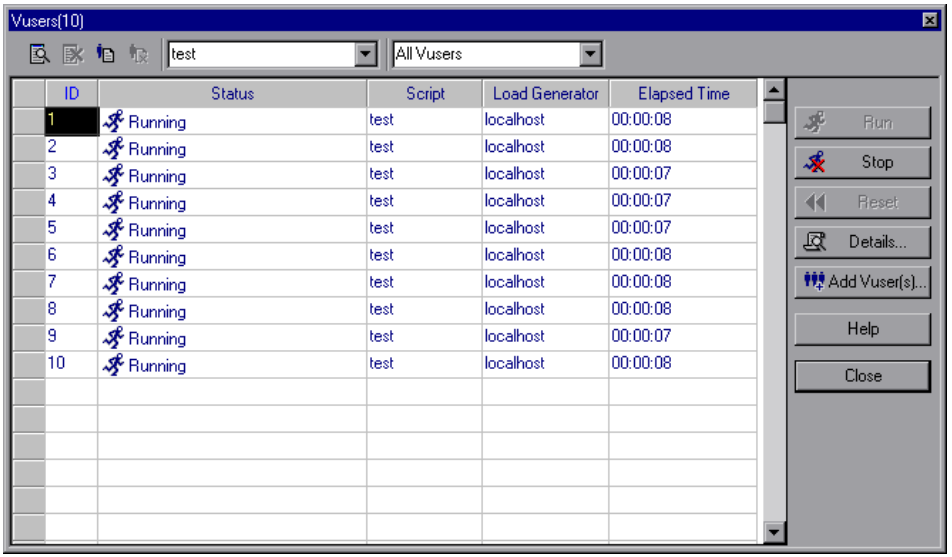
Scenario Groups window

Online Monitor Graphs

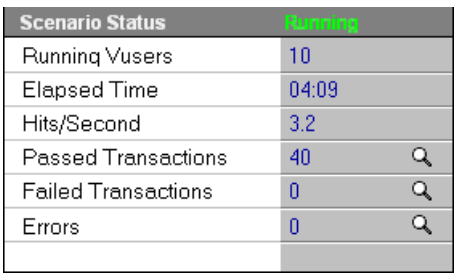
Scenario Status window

During scenario execution, you use the Scenario Groups window in the Run view to monitor the actions of all the Vusers and Vuser groups in the scenario. The Status field of each Vuser group displays the current state of each Vuser in the group.

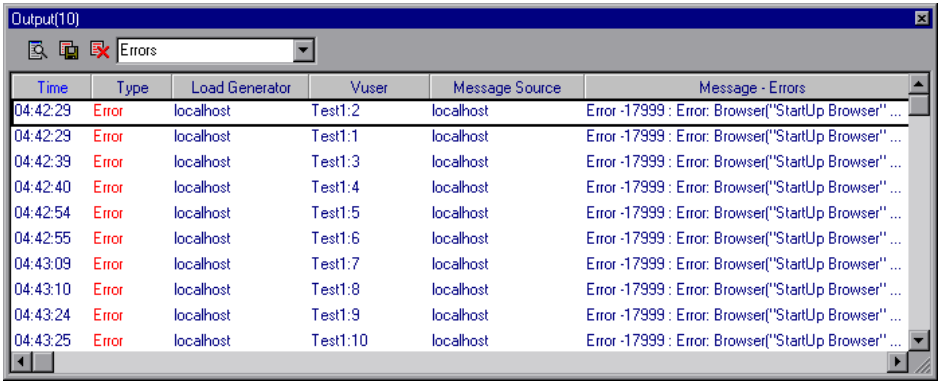
You can also manipulate individual Vusers within the Vuser groups you have defined by selecting a group and clicking the **Vusers** button. The Vusers dialog box appears, with a list of the ID, Status, Script, Load Generator, and Elapsed Time (since the beginning of the scenario) for each of the Vusers in the group.



In addition, you can view a synopsis of the running scenario in the box at the top right-hand corner of the Run view.



While the scenario runs, the Vusers and load generators send error, notification, and warning messages to the Controller. You can view these messages in the Output window (**View** > **Show Output**).



Note that there are different color codes for the Output window messages. Error messages are displayed in red, while notification and warning messages are displayed in blue.

You use the online monitors and online monitor graphs to monitor Vuser status, transactions, system resources, database server resources, Web resources, Web server resources, Web application server resources, streaming media resources, firewall server resources, and network delay, while running a scenario. For more information on online monitors, see Chapter 14, “Online Monitoring.”

Part II

Designing a Scenario

5

Creating a Manual Scenario

You build a manual scenario by creating groups and specifying the scripts, the load generators, and the number of Vusers included in each group. This chapter describes how to create a manual scenario. See Chapter 7, "Creating a Goal-Oriented Scenario" for information on creating a goal-oriented scenario.

This chapter discusses:

- Creating Vuser Groups
- Configuring Vusers in a Vuser Group
- Configuring Load Generators
- Configuring Load Generator Settings
- Configuring Scripts
- Using Relative Paths for Scripts

About Creating a Scenario

To test your system with LoadRunner, you must create a scenario—a file with information about the test session. The scenario is the means by which you emulate a real-life user. The scenario contains information about how to emulate real users: the groups of virtual users (Vusers), the test scripts the Vusers will run, and the load generator machines upon which to run the scripts.

If you chose to create a manual scenario, each script you selected in the New Scenario dialog box is assigned to a Vuser group. To each Vuser group you then assign a number of virtual users. You can instruct all Vusers in a group

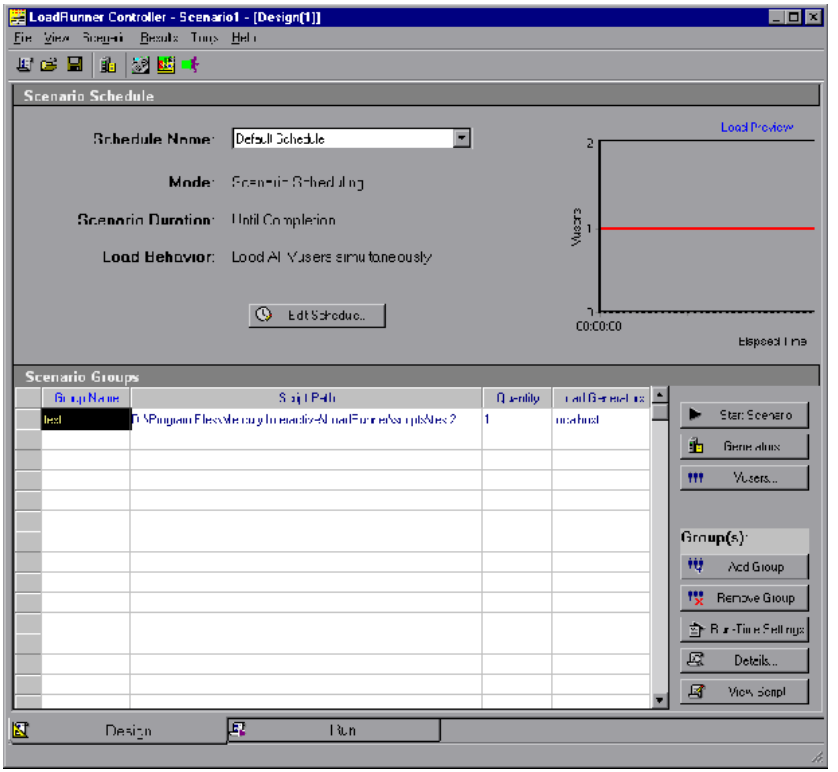
to run the same script on the same load generator machine, or you can assign different scripts and load generators to the various Vusers in a group.

Once you create your Vuser groups, you select or build a schedule for your scenario. See Chapter 6, "Scheduling a Manual Scenario" for more information on creating a scenario schedule.

Creating Vuser Groups

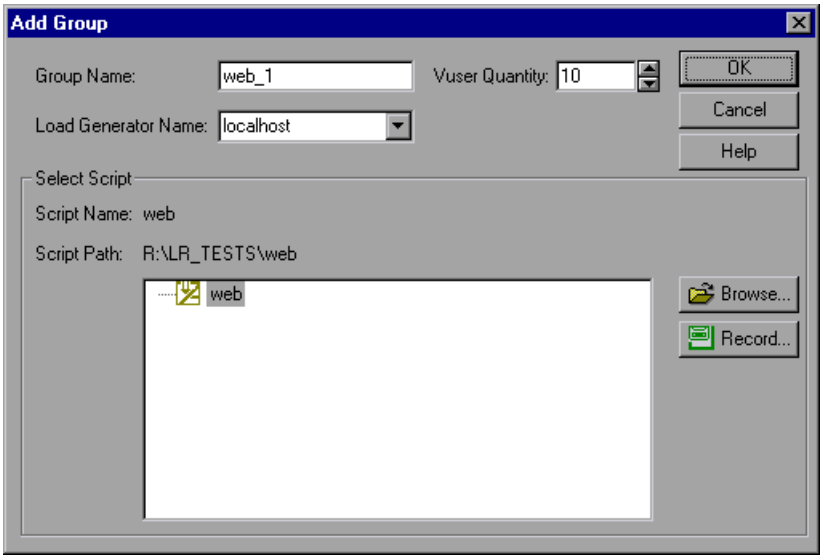
A scenario consists of groups of Vusers which emulate human users interacting with your application. When you run a scenario, the Vusers generate load on the server, and LoadRunner monitors the server and transaction performance.

You create Vuser groups from the Scenario Groups window of the Controller.



To create Vuser Groups:

1 Click the **Add Group** button on the right of the Scenario Groups window. The Add Group dialog box opens:

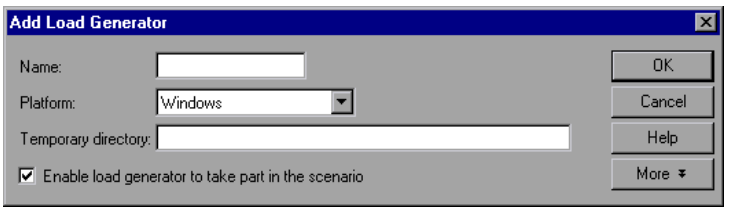


2 In the Group Name box, enter a name for the Vuser group.

3 From the Vuser Quantity box, select the number of Vusers that you want to create in the group.

4 Select a load generator from the Load Generator Name list. The Load Generator list contains all load generators that you previously added to the scenario.

To use a load generator that does not appear, select **Add** from the Load Generator Name list. The Add Load Generator dialog box opens:



Type the name of the load generator in the Name box.

By default, LoadRunner stores temporary files on the load generator during scenario execution, in a temporary directory specified by the load generator's TEMP or TMP environment variables. To override this default for a specific load generator, type a location in the Temporary Directory box.

To allow the load generator to take part in the scenario, check **Enable load generator to take part in the scenario**.

Click **More** to expand the dialog box and show the Add Load Generator tabs. For information on configuring settings for each load generator, see "Configuring Load Generator Settings" on page 60.

Click **OK** to close the Add Load Generator dialog box.

5 Select a script from the script list. The script list contains all scripts that you previously added to the scenario.

To use a script that does not appear, click the **Browse** button. Browse to select the path and file name of the new script. To use a VB Vuser script, select the .usr file.

Note: When you specify the location of a script, you can specify a location that is relative to the current scenario directory. For details, see "Using Relative Paths for Scripts" on page 70.

---

6 Click **OK** to close the Add Group dialog box. The new group's properties appear in the Scenario Groups window.

Deleting a Vuser Group

To delete a Vuser group, click the **Remove Group** button on the right of the Scenario Groups window, or right-click the Vuser group and select **Remove Group**.

Modifying a Vuser Group

You can modify the script, vuser quantity, and load generator for a Vuser group directly from the Scenario Groups window of the Controller, or by using the Group Information dialog box.

To modify a Vuser Group directly from the Scenario Groups window:

1 Select the **Group Name**, **Script Path**, **Quantity**, or **Load Generator** you want to modify.

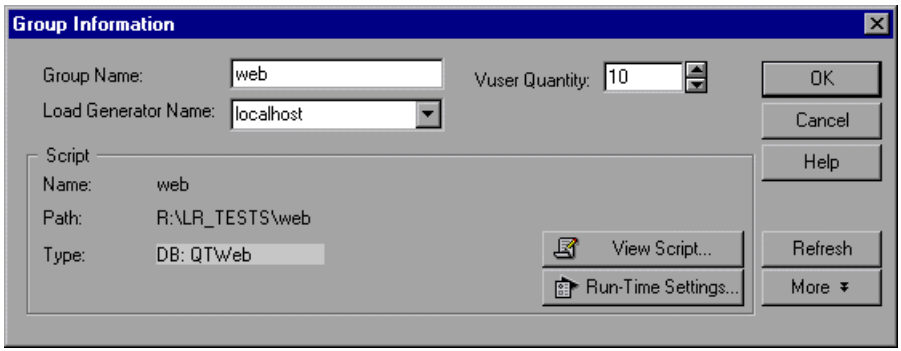
2 Enter or select another name or number for the property.

3 To modify a Vuser group script's run-time settings, click the **Run-Time Settings** button on the right of the Scenario Groups window. For more information about run-time settings, see "Configuring Scripts" on page 67.

4 To edit a Vuser group's script, click the **View Script** button on the right of the Scenario Groups window. LoadRunner's script generation tool, VuGen, opens. For more information on editing scripts, see the *Creating Vuser Scripts* guide.

To modify a Vuser Group using the Group Information dialog box:

1 Click the **Details** button on the right of the Scenario Groups window, or right-click the property you want to modify and select **Details**. The Group Information dialog box opens.



2 In the Group Name box, enter the Vuser group name.

3 From the Vuser Quantity box, select the number of Vusers that you want to run in the group.

4 Select a load generator from the Load Generator Name list. To use a load generator that does not appear, select **Add** from the Load Generator Name list and add a new load generator using the Add Load Generator dialog box.

5 To modify the run-time settings you specified while recording a script using VuGen, click **Run-Time Settings**. For more information about run-time settings, see "Configuring Scripts" on page 67.

6 To edit a Vuser group's script, click **View Script**. LoadRunner's script generation tool, VuGen, opens. For more information on editing scripts, see "Configuring Scripts" on page 67.

7 Click **OK** to close the Group Information dialog box.

Sorting Vuser Groups in the Scenario Groups Window

Once you have created your Vuser groups, you can sort them by group name, script name, load generator name, or quantity of Vusers assigned to the group.

To sort Vuser groups:

➤ Select the column by which you want to sort the groups. Click the column header.

➤ Alternatively, you can right-click anywhere within the column you want to sort, and select **Sort Groups**. Choose to sort by name, path, quantity, or generator.

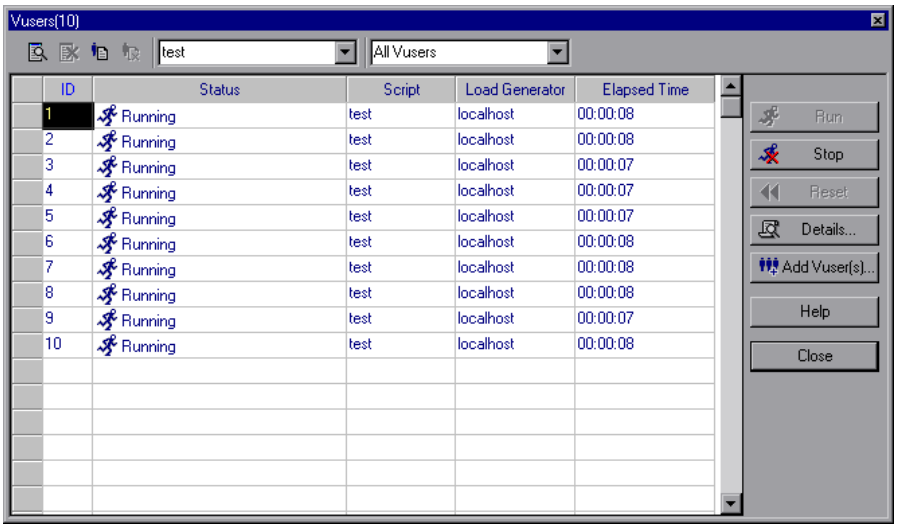
➤ To instruct the Controller to automatically sort a new Vuser group entry, right-click the entry and select **Auto Sort**.

Configuring Vusers in a Vuser Group

You can define properties for individual Vusers within the Vuser groups you have defined using the Vusers dialog box. To each Vuser you can assign a different script and/or load generator machine.

To define properties for individual Vusers:

1 Select the Vuser group whose Vusers you want to modify, and click the **Vusers** button on the right of the Scenario Groups window. The Vusers dialog box opens.



2 To change the script for an individual Vuser, select a different script in the Script column. Alternatively, you can click the **Details** button, and select a different script from the script list in the Vuser Information dialog box.

3 To change the load generator on which a Vuser runs, select a different load generator in the Load Generator column. Alternatively, you can click the **Details** button, and select a different load generator from the Load Generator Name list in the Vuser Information dialog box.

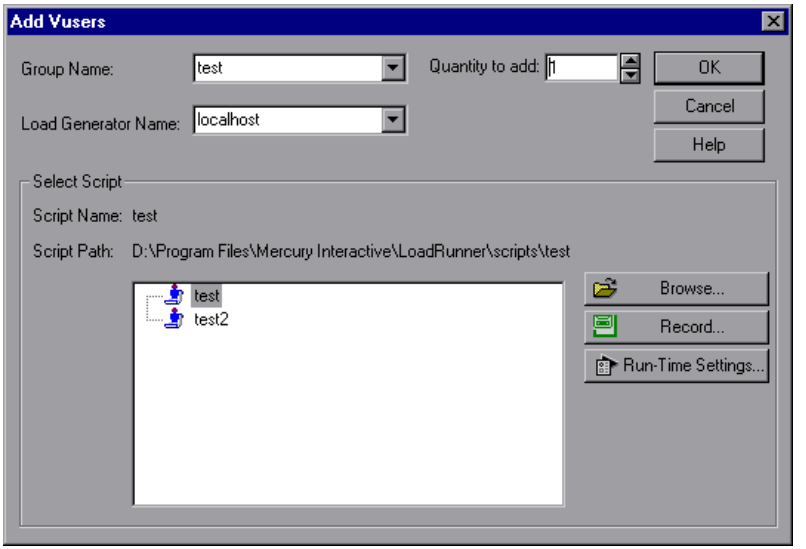
To use a load generator that does not appear, select **Add** from the Load Generator Name list and add a new load generator using the Add Load Generator dialog box.

Adding Vusers to a Vuser Group

You add Vusers to a Vuser group and define their properties using the Add Vusers dialog box.

To add Vusers to a Vuser group:

1 In the Vusers dialog box, click the **Add Vuser(s)** button. The Add Vusers dialog box opens.



2 From the Group Name box, select the name of the Vuser group.

3 From the Quantity to add box, select the number of Vusers that you want to add to the group.

4 Select a load generator from the Load Generator Name list. To use a load generator that does not appear, select **Add** from the Load Generator Name list and add a new load generator using the Add Load Generator dialog box.

5 Select a script from the script list. The script list contains all scripts that you previously added to the scenario.

To use a script that does not appear, click the **Browse** button. Browse to select the path and file name of the new script. To use a VB Vuser script, select the .usr file.

Note: When you specify the location of a script, you can specify a location that is relative to the current scenario directory. For details, see "Using Relative Paths for Scripts" on page 70.

To modify the run-time settings you specified while recording a script using VuGen, click **Run-Time Settings**. Note that modifying the run-time settings for the new Vuser will modify the run-time settings for all the Vusers in the group. For more information about run-time settings, see "Configuring Scripts" on page 67.

6 Click **OK** to close the Add Vusers dialog box. The new Vuser's properties appear in the Vusers dialog box.

Configuring Load Generators

You can set a load generator's attributes while adding it to the load generator list, or modify the attributes of an existing load generator at any time, using the Load Generators dialog box.

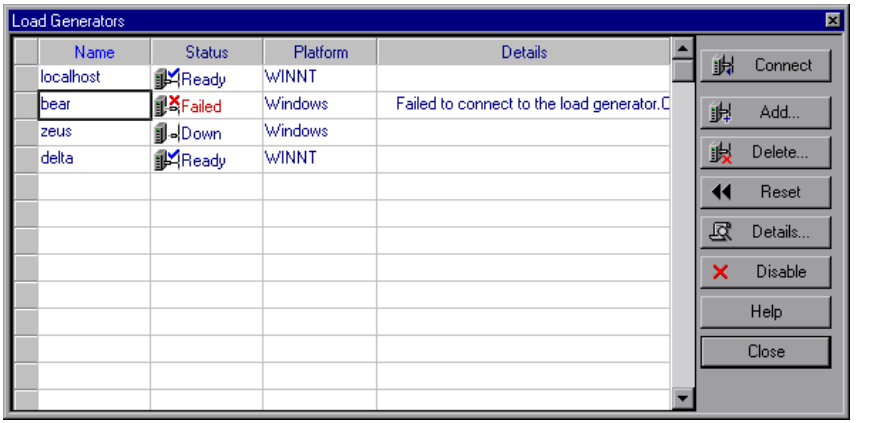
To configure global settings for all load generators participating in the scenario, use LoadRunner's Options dialog box. For more information, see Chapter 9, "Configuring a Scenario." To set properties specific for each load generator, use the Load Generators dialog box as described below.

You can also indicate which load generators will run Vusers in the scenario. For example, if a load generator is unavailable for a particular scenario run, you can exclude it temporarily instead of removing it entirely from your list of load generators.

You select which load generators will take part in the scenario by using the Enable and Disable commands. Disabling a load generator temporarily removes it from the list. Enabling a load generator reinstates it. Disabling load generators is particularly useful if you want to isolate a specific machine to test its performance.

To configure a load generator:

1 Click the **Generators** button, or select **Scenario** > **Load Generators**. The Load Generators dialog box opens. The **Name** of the load generator, its **Status**, **Platform**, and **Details** are displayed.

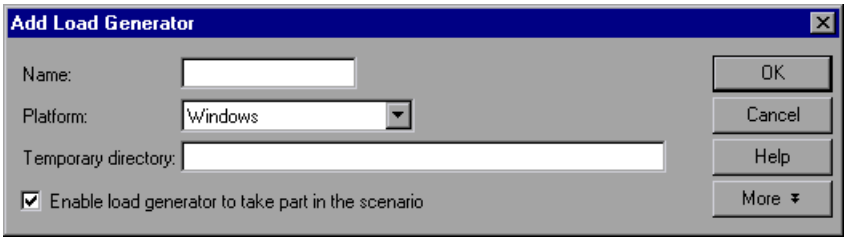


2 Click **Connect** to change the Status of the load generator from Down to Ready. Click **Disconnect** to change the Status of the load generator from Ready to Down.

3 To disable a load generator, select the load generator and click **Disable**. The load generator name changes from blue to gray, and the load generator is disabled. To enable a load generator, select the load generator and click **Enable**. The load generator name changes from gray to blue, and the load generator is enabled.

4 To view details of a load generator, select the load generator and click **Details**. The Load Generator Information dialog box opens with information about the load generator you selected.

5 To add a load generator, or modify information for an existing load generator, click **Add**. The Add Load Generator dialog box opens.



Type the name of the load generator in the **Name** box.

By default, LoadRunner stores temporary files on the load generator during scenario execution, in a temporary directory specified by the load generator's TEMP or TMP environment variables. To override this default for a specific load generator, type a location in the Temporary Directory box.

To allow the load generator to take part in the scenario, check **Enable load generator to take part in the scenario**.

Click **More** to expand the dialog box and show the Add Load Generator tabs. For information on configuring these settings, see "Configuring Load Generator Settings" on page 60.

6 To remove a load generator, click **Delete**.

7 Click **Close** to close the Load Generators dialog box. The load generator name you entered appears in the Load Generators list; its status is set to Down.

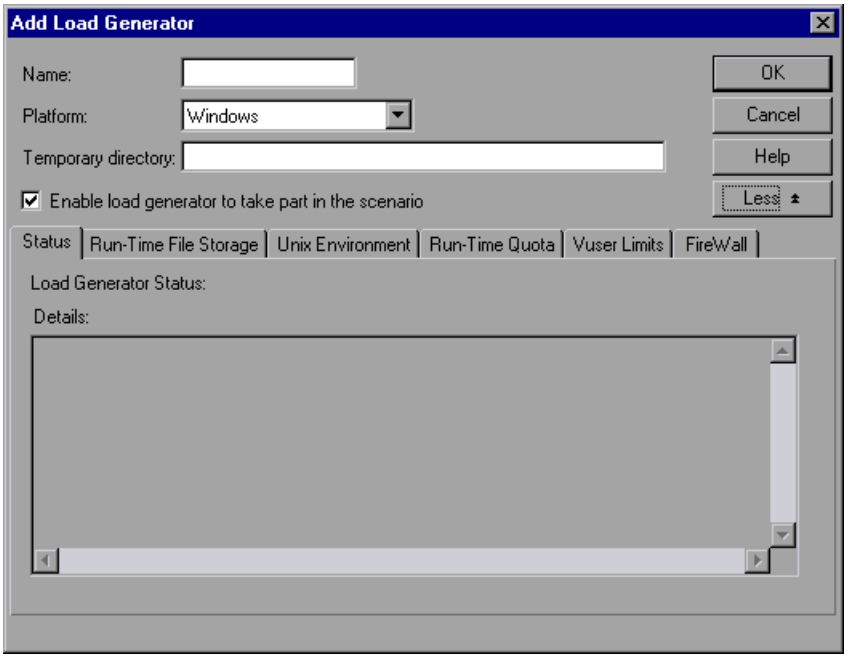
Configuring Load Generator Settings

You can configure additional settings for individual load generators using the tabs in the Add Load Generator or Load Generator Information dialog boxes. The settings that can be configured are: Run-Time File Storage, UNIX Environment, Run-Time Quota, Vuser Limits, Connection Log (Expert mode), and FireWall.

You can configure global settings for all load generators participating in the scenario, using the Options dialog box. For more information, see Chapter 9, "Configuring a Scenario."

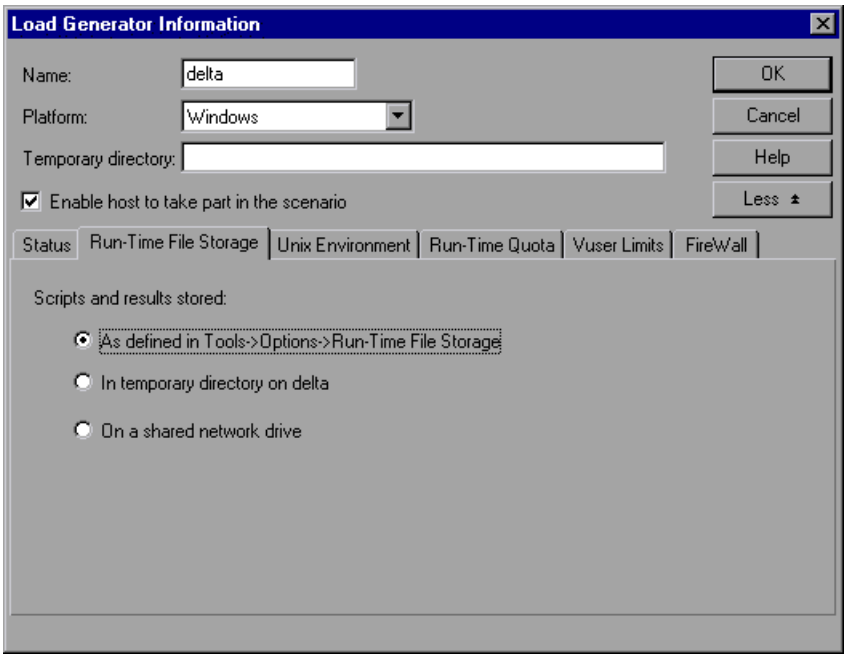
To configure load generator settings:

1 From the Add Load Generator or Load Generator Information dialog box, click **More** to expand the box and show the Status, Run-Time File Storage, UNIX Environment, Run-Time Quota, and Vuser Limits tabs.



2 Select the **Status** tab to display details of the Load Generator Status.

3 Select the **Run-Time File Storage** tab to specify the result directory for the performance data that LoadRunner gathers from each load generator during a scenario.

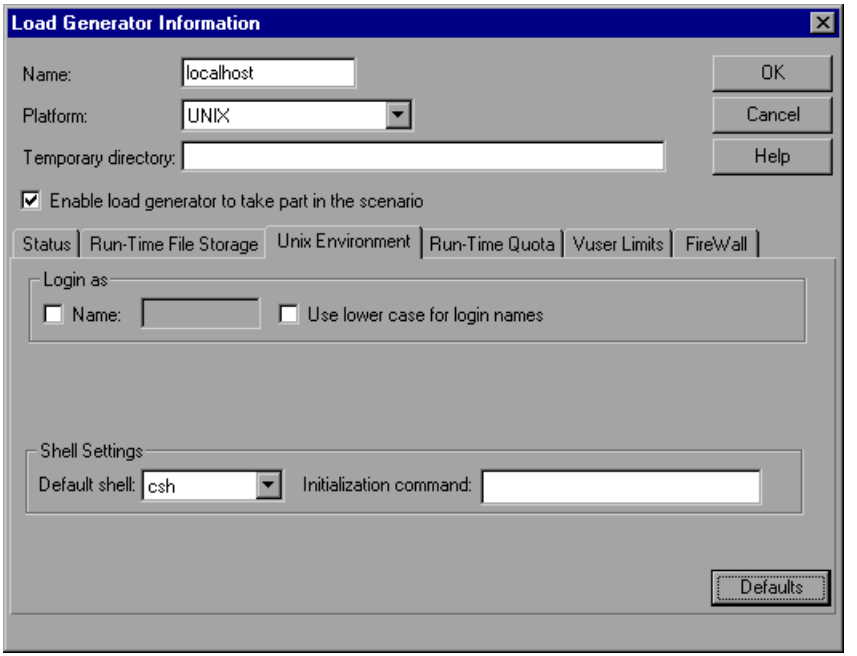


To store the results as specified in the global settings, click **As defined in Tools > Options > Run-Time File Storage.** To store the results temporarily on a hard drive of the load generator computer, click **In temporary directory on <*load generator name*>**. To store the scenario scripts or results on a shared network drive, click **On a shared network drive**. To set the network location for the results, see Chapter 10, "Preparing to Run a Scenario."

Note: If the load generator is *localhost*, then LoadRunner stores the scripts and results on a shared network drive, and the checkboxes and radio buttons for setting the location are all disabled.

---

4 Select the **UNIX Environment** tab to configure the login parameters and shell type for each UNIX load generator.



To specify a login name other than the current Windows user, select the **Name** check box and specify the desired UNIX login name. To login with lower case characters, select the **Use lower case for login names** check box.

Note: For information on the Local User setting available in Expert mode, see "Working in Expert Mode" on page 327.

In the Default Shell box, select **csh** (C Shell—the default), **bsh** (Bourne Shell), or **ksh** (Korn Shell).

To allow LoadRunner to run your application under the Korn shell, you first need to make sure that the *.profile* file contains all of the LoadRunner environment settings—for example, the M_LROOT definition and the LicenseManager variable. These environment settings already exist in your

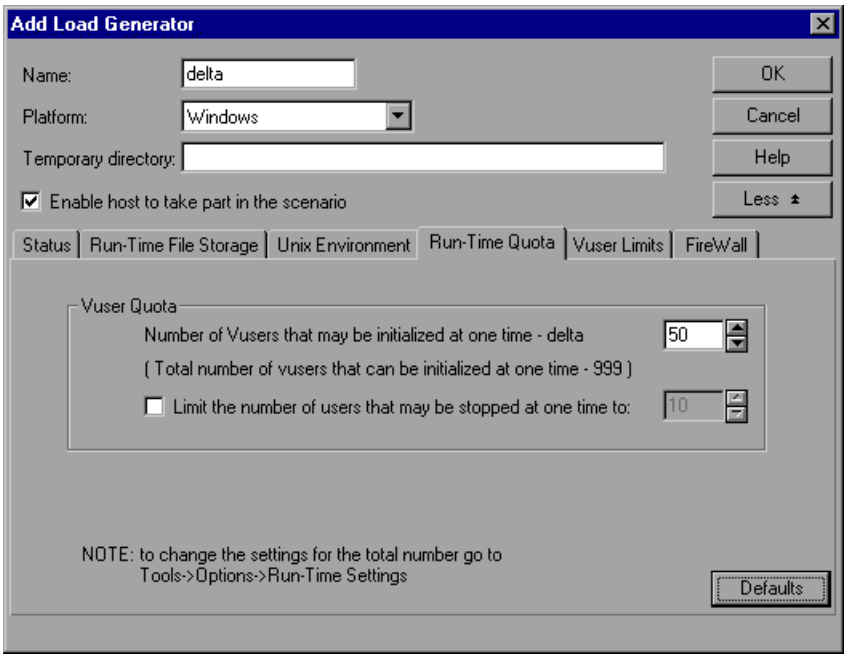
.cshrc file. Your UNIX $M_LROOT/templates directory contains a template for the *.profile* file, called *dot profile*. Use the template as a guide for modifying your *.profile* file with the LoadRunner environment settings.

Note: If you are using a Korn shell (ksh), you must delete all LoadRunner settings from the *.cshrc* file (e.g. M_LROOT) before executing the scenario.

In the Initialization Command box, enter any command line options that LoadRunner will use when logging on to a UNIX system. This initialization command will run as soon as the shell opens.

For example, you could select ksh and use the following initialization command:
. .profile;

5 Select the **Run-Time Quota** tab to specify the maximum number of Vuser types that the load generator will initialize or stop simultaneously.

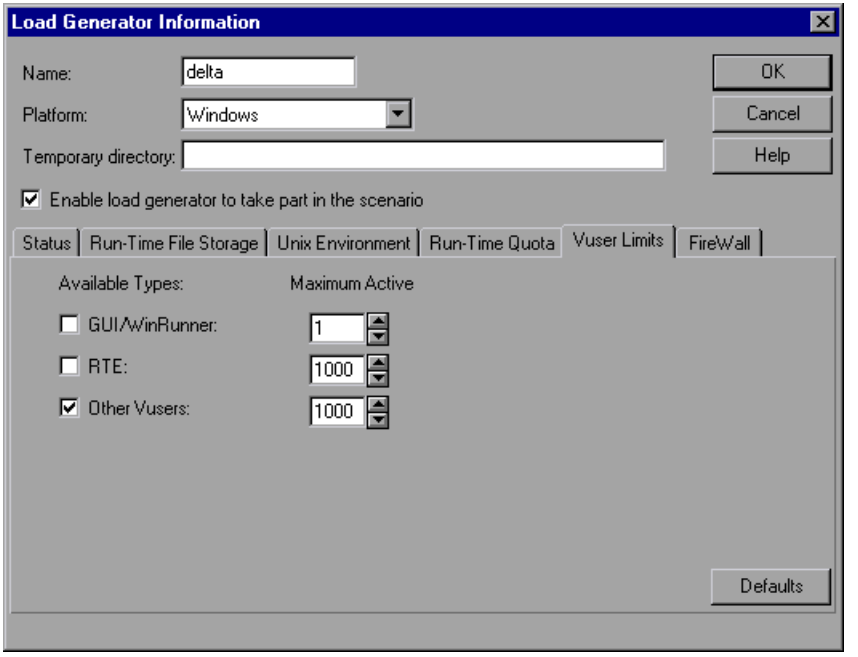


Click **Defaults** to use the Default values.

Initializing or stopping a large number of Vusers simultaneously places large stress on a load generator. To reduce stress on a load generator, you can initialize or stop smaller batches of Vusers.

You can set run-time quotas for an entire scenario using the Run-Time Settings tab in the Options dialog box. For information on setting quotas globally for an entire scenario, see Chapter 9, "Configuring a Scenario."

6 Select the **Vuser Limits** tab to modify the maximum number of GUI, RTE, and other Vusers that a load generator can run.

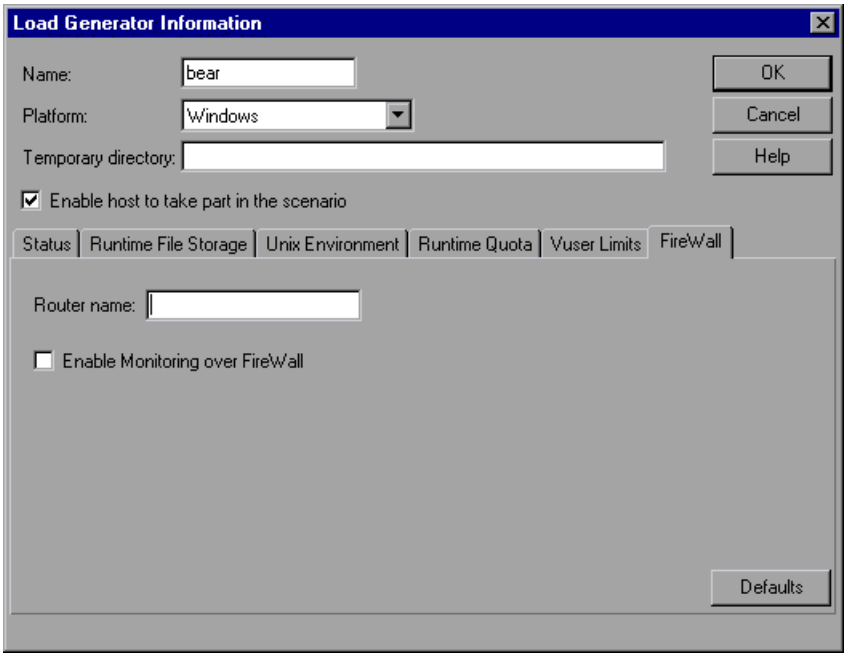


In the Maximum Active boxes enter the maximum number of Vusers of each type that the load generator can run.

Note: The maximum number of active Vusers that you specify must not exceed the number of Vusers that you are licensed to run. To check your Vuser licensing limitations, choose **Help** > **About LoadRunner**.

---

7 Select the **FireWall** tab to enable monitoring to occur through a firewall.

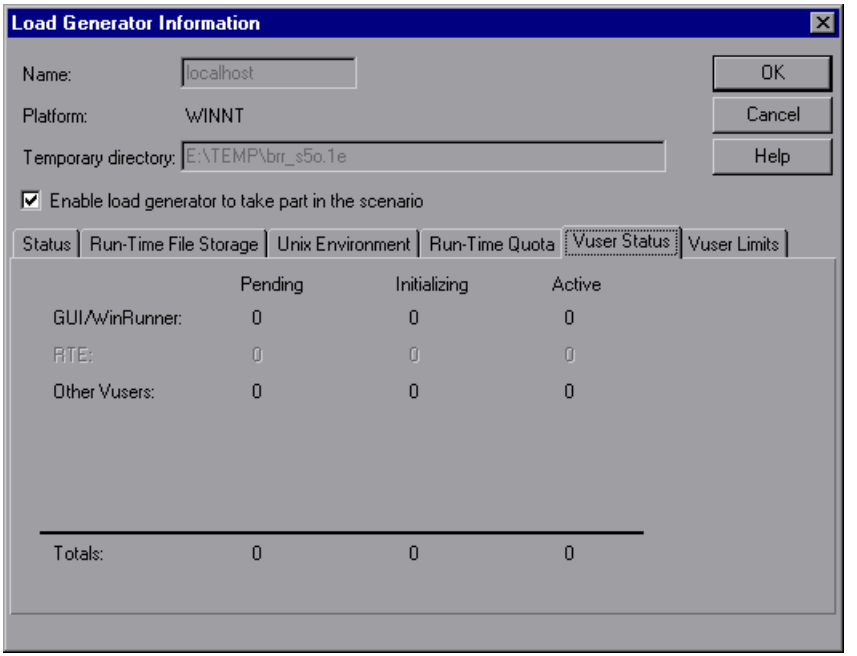


Note: If the load generator is connected, you cannot change values in the **FireWall** tab.

Specify the name of the router the load generator is using, and check the **Enable Monitoring over FireWall** check box to enable LoadRunner to monitor the load generator machine through a firewall.

Note: If you change the name of the load generator to the name of the local host, you cannot set any values in the **FireWall** tab.

8 If the load generator machine is connected, you can view the **Vuser Status** tab, which displays the number of GUI-WinRunner, RTE, and other Vusers that are *Pending*, *Initializing*, and *Active* on the selected load generator machine.



Note: For information on the Connection Log tab available in Expert mode, see "Working in Expert Mode" on page 327.

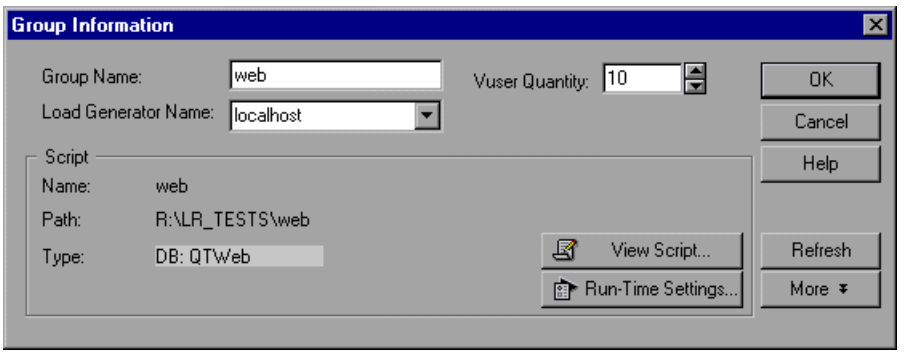
9 Click **OK** to close the Add Load Generator or Load Generator Information dialog box and save your settings.

Configuring Scripts

Once you have selected a script for a Vuser or Vuser group, you can edit the script, or view the details of the script you selected, from the Vusers or Group Information dialog boxes.

To edit and view the details of a script used by a Vuser group:

1 Select the Vuser group whose script you want to modify, and click the **Details** button on the right of the Scenario Groups window, or right-click the Vuser group and select **Details**. The Group Information dialog box opens displaying the Name, Path, and Type of the script.



2 Click **Run-Time Settings** to set the script's run-time settings (optional). These settings let you customize the way the Controller executes a Vuser script. You can also modify the run-time settings using the Vuser Generator (VuGen). Note that several protocols, such as Web and Java, have specific settings. For information on configuring the run-time settings, refer to the *Creating Vuser Scripts* guide.

Note: When you open the Run-Time Settings dialog box from the Controller, the settings you previously set using VuGen appear. If you modify these settings from the Controller, LoadRunner runs the script using the modified settings.

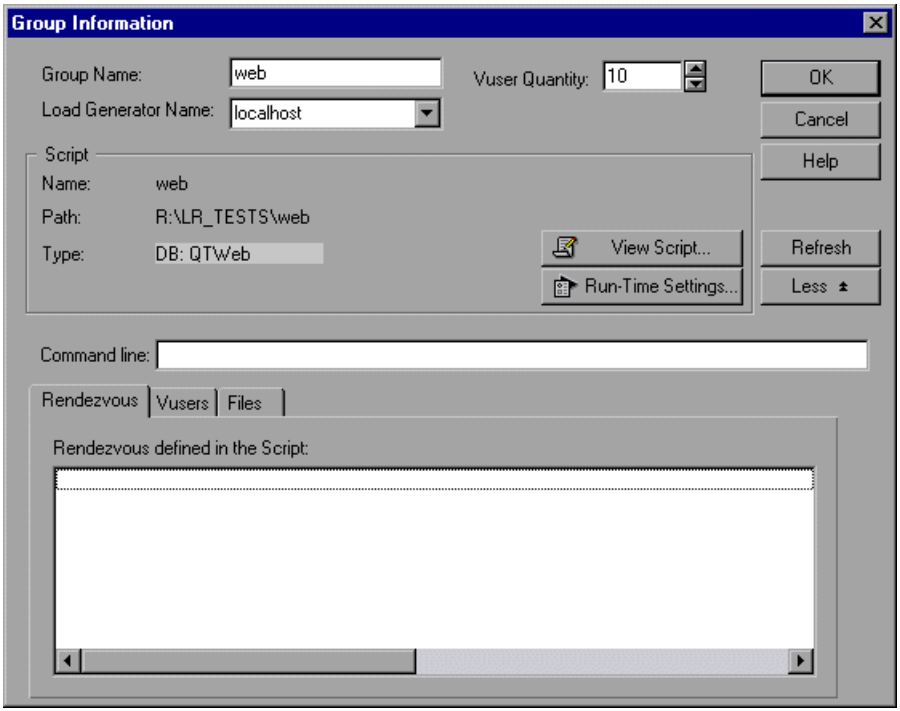
---

3 To edit the script, click **View Script**. The script generation tool, VuGen, opens. For more information on editing scripts, see the *Creating Vuser Scripts* guide.

Note: If you use VuGen to make changes to a script while the Controller is running, click the **Refresh** button in the Controller's Group Information dialog box to update the script details in the scenario.

---

4 Click **More** to expand the Group Information dialog box and view additional script information.



5 In the Command Line box, type any command line options to use when running the script. For example: -x value -y value

For information about passing command line argument values to a script, refer to the *Creating Vuser Scripts* guide.

6 To see the rendezvous points declared in the selected script, click the **Rendezvous** tab.

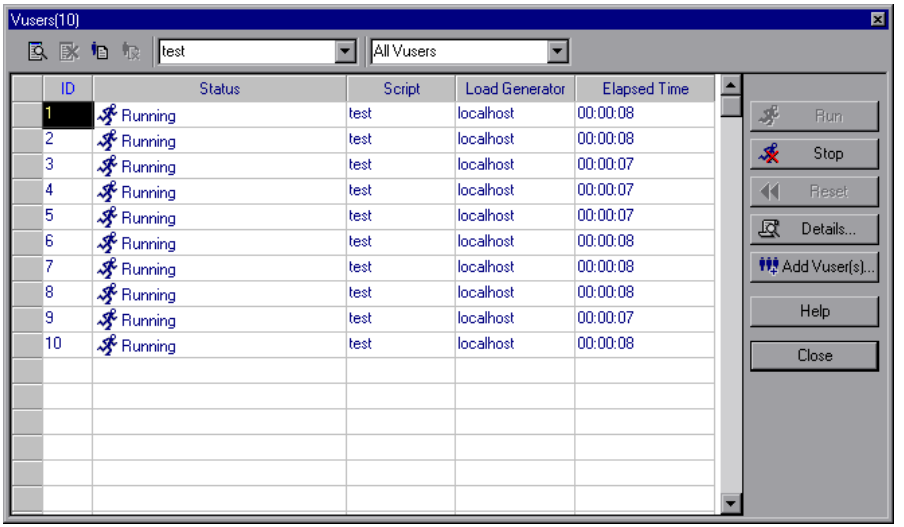
7 To see the list of Vusers associated with the selected script, click the **Vusers** tab. If you have not yet created Vusers, the box will be empty.

8 To see the list of files used by the script, click the **Files** tab. By default this list shows all files in the script's directory (only after your script has been added to the script list). These files include the configuration settings file, the init, run, and end portions of the script, the parameterization definitions file, and the *usr* file. To add a file to the list, click **Add** and add the file name. Note that you can delete the files that you add, but not the other files listed.

9 Click **OK** to close the Group Information dialog box.

To edit and view the details of a script used by an individual Vuser:

1 Click the **Vusers** button on the right of the Scenario Groups window. The Vusers dialog box opens.



To view details of a script, click **Details**. The script's name and path are displayed in the Vuser Information dialog box. To select a different script, click the **Browse** button and select the path and file name of the new script. To select a VB Vuser script, browse to locate the .usr file.

Note: When you specify the location of a script, you can specify a location that is relative to the current scenario directory. For details, see "Using Relative Paths for Scripts" on page 70.

---

2 To edit a script, right-click the script in the Vusers dialog box, and select **View Script**. The script generation tool, VuGen, opens. For more information on editing scripts, see the *Creating Vuser Scripts* guide.

3 To modify the run-time settings you specified while recording a script using VuGen, right-click the script in the Vusers dialog box, and select **Run-Time Settings**. Note that modifying the run-time settings for one Vuser will modify the run-time settings for all the Vusers in the group that are using the same script. For more information about run-time settings, see the *Creating Vuser Scripts* guide.

Using Relative Paths for Scripts

When you specify the location of a script, you can specify a relative location. The location can be relative to the current scenario directory, or the LoadRunner installation directory.

You can specify a path relative to the current scenario directory by typing either of the following notations at the start of the script path:

.\	indicates that the path is relative to the location of the scenario directory.
..\	indicates that the path is relative to the location of the parent directory of the scenario directory.

For example, if the current scenario is located at F:\scenarios, to specify a script located at F:\scenarios\scripts\user.usr, you could type:

```
.\scripts\user1.usr
```

You can specify a path relative to the LoadRunner installation directory by typing a percent sign (%) at the beginning of the script path. For example, if the LoadRunner installation directory is located at F:\LoadRunner, to specify a script located at F:\LoadRunner\scripts\user1.usr, you could type:

```
%\scripts\user1.usr
```

Note: When specifying a relative path, you can include standard DOS notation (.\ and ..\) inside the path, as shown in the following example: M:\LR\my_tests\..\..\test.usr.

When you run a scenario, by default, the script is copied to a temporary directory on the Vuser group machine. This enables the Vuser group load generator to access the script locally instead of over a network.

You can instruct the Controller to store the script on a shared network drive (see Chapter 9, "Configuring a Scenario.") If you configure the Controller to save the script to a network drive, you must ensure that the Vuser load generator recognizes the drive. The Script window contains a list of all the Vuser scripts and their paths. A script's path is based on the Controller load generator's mapping of that location. If a Vuser load generator maps to the script's path differently, path translation is required. Path translation converts the Controller load generator's mapping to the Vuser load generator's mapping. For more information see Appendix B, "Performing Path Translation."

6

Scheduling a Manual Scenario

After you create the Vuser groups in a manual scenario, you can set the start and duration time of the scenario or of the Vuser groups within the scenario. In addition, you can gradually run the Vusers within the scenario or within a Vuser group.

This chapter describes:

- Selecting a Schedule
- Scheduling a Scenario
- Delaying the Start of a Scenario
- Scheduling Vuser Groups

About Scenario Scheduling

An important factor in the creation of a manual scenario, is developing a test that accurately portrays user behavior—the types of actions and the timing of those actions. The actions are represented by Vuser scripts. You set the timing aspect of the manual scenario using the LoadRunner Schedule Builder.

You can instruct LoadRunner to begin executing a scenario or Vuser group with a delay. You specify the number of minutes to wait from the time a *Run* command is issued, until the scenario or Vuser group begins.

You can limit the time duration of a scenario or Vuser group. You specify the number of minutes a scenario or Vuser group should be in the Running state. When the scenario or group reaches its time limitation, it finishes.

You can stipulate how many Vusers LoadRunner runs within a certain time frame during a scenario or Vuser group. You specify whether LoadRunner should run all Vusers simultaneously, or run only a certain number of Vusers within a specified amount of time.

Note: Rendezvous points in a Vuser script interfere with a scheduled scenario. If you have rendezvous points in your script, therefore, your scenario will not run as scheduled.

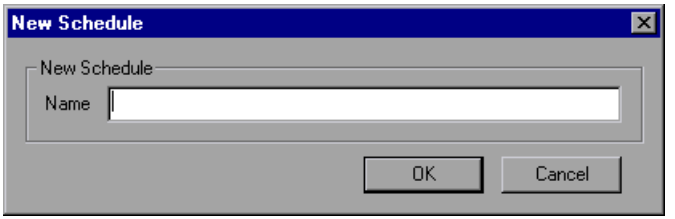
Selecting a Schedule

You select the schedule you want to use for your manual scenario from the Scenario Name box in the Scenario Schedule window. You can select one of the existing schedules—Slow Ramp Up or Ramp Up—or New Schedule, if you want to create a schedule with new properties using the Schedule Builder.

Note that you can also change the properties for one of the three existing schedules using the Schedule Builder.

To create a new schedule:

1 Select **<new schedule>** from the Scenario Name box in the Scenario Schedule window. The New Schedule dialog box opens.

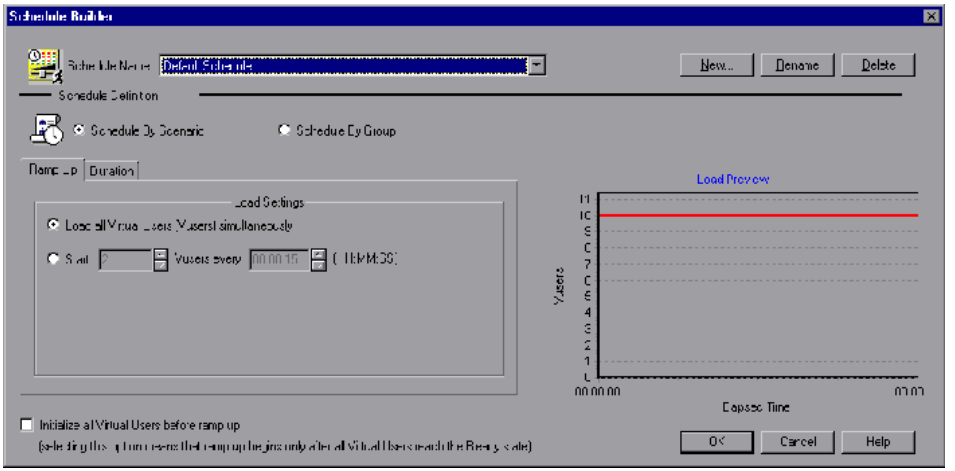


2 In the **Name** text box, type the name of the New Schedule and click **OK**. The Schedule Builder dialog box opens.

To modify the properties of an existing schedule:

1 Select **Slow Ramp Up** or **Ramp Up** from the Scenario Name box in the Scenario Schedule window.

2 Select **Scenario** > **Schedule Builder**, or click the **Edit Schedule** button. The Schedule Builder dialog box opens.



To rename a schedule, click **Rename**. Enter the new name you want to use in the dialog box that opens. To delete a schedule, click **Delete**.

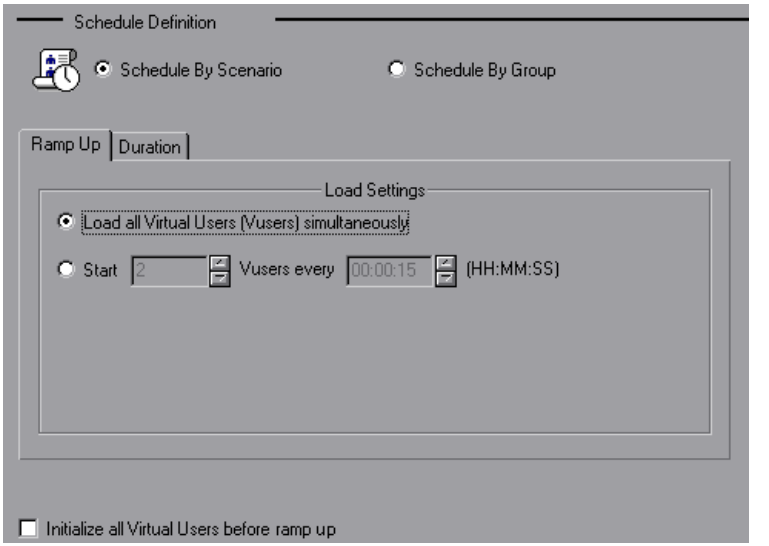
Scheduling a Scenario

Using the Schedule Builder, you can control the execution of your scenario by:

- limiting the scenario duration
- gradually running Vusers within a scenario

To set the scheduling options for a scenario:

1 Select the **Scenario Scheduling** option.



2 In the Ramp Up tab:

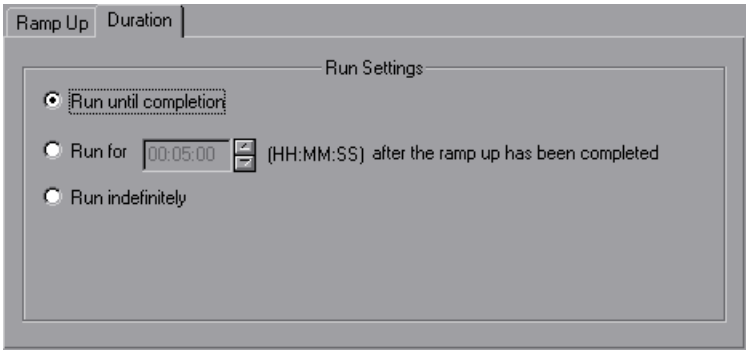
➤ Select **Load all Virtual Users (Vusers) simultaneously** if you want to run all the Vusers at once.

➤ To gradually run the Vusers, select the number of Vusers you want to begin running concurrently and the amount of time you want LoadRunner to wait between Vuser ramp ups.

Note: You can add Vusers to a scenario while the scenario is running. However, if you add a Vuser after all the Vusers in the scenario have been ramped up, the new Vuser will not run in the scenario.

---

3 To set the duration of the scenario, click the **Duration** tab.



Choose one of the following options:

➤ Run until completion.

➤ Specify the amount of time for which you want to run the scenario, once all the Vusers have been ramped up.

➤ Run indefinitely.

Note: The duration setting overrides the Vuser iteration settings. This means that if the duration is set to five minutes, the Vusers will continue to run as many iterations as required in five minutes, even if the run-time settings specify only one iteration.

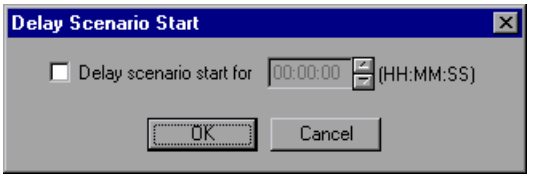
---

4 To instruct LoadRunner to initialize Vusers before beginning to load them, select **Initialize all Vusers before Ramp-Up**.

5 Click **OK** to close the Schedule Builder and save your settings.

Delaying the Start of a Scenario

You can instruct LoadRunner to start running a scenario at a later point in time. To delay the start of the scenario, select **Scenario** > **Start Time.** The Delay Scenario Start dialog box opens.



Check **Delay scenario start for** and select the amount of time you want to delay the start of your scenario.

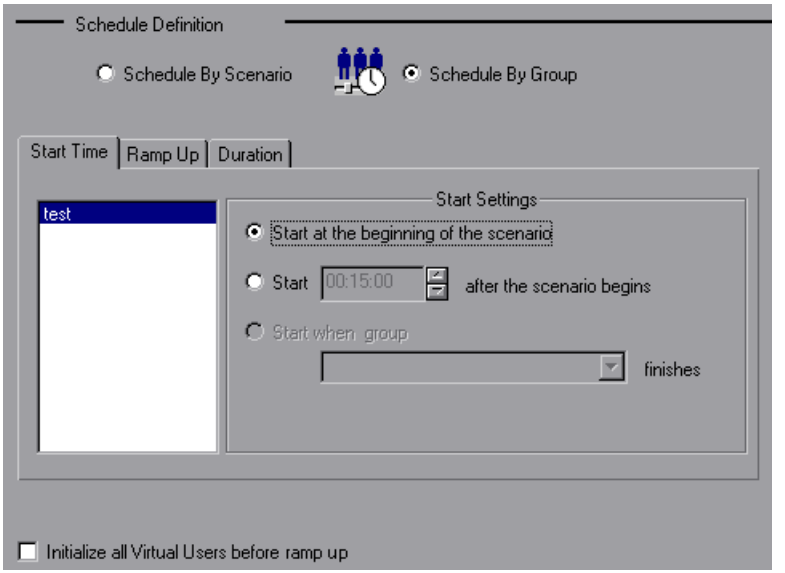
Scheduling Vuser Groups

After you create a Vuser group, you can schedule the group's script execution by setting:

- the amount of time after the start of the scenario that the group must wait before it starts running
- the number of Vusers that will run within a specified period of time
- the amount of time the group will run

To schedule a Vuser Group:

1 Select the **Group Scheduling** option.

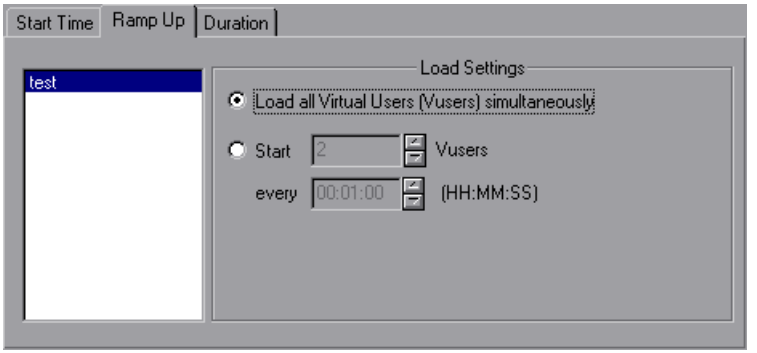


2 Select a group from the box on the left.

3 To set the start time for the group, click the **Start Time** tab. Choose one of the following three options:

- Start the group at the beginning of the scenario.
- Specify the amount of time that you want to elapse before the group begins running.
- Specify that you want to begin the group after another group finishes running.

4 To set the ramp up for the group, click the **Ramp Up** tab.



Choose one of the following options:

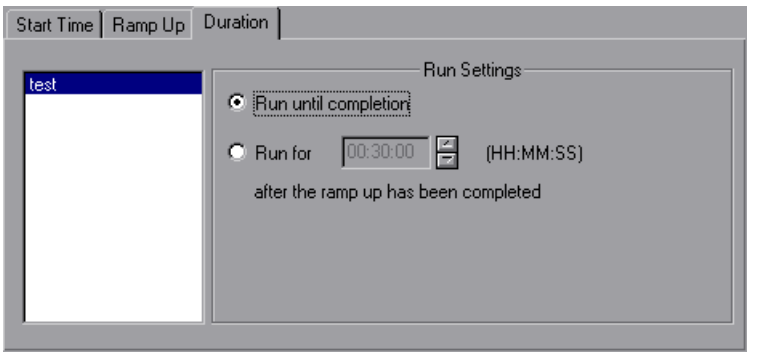
➤ Load all of the Vusers simultaneously.

➤ Specify the number of Vusers you want to begin running concurrently and the amount of time you want LoadRunner to wait between Vuser ramp ups.

Note: You can add Vusers to a Vuser group while the scenario is running. However, if you add a Vuser to a group after all the Vusers in the group have been ramped up, the new Vuser will not run in the scenario.

---

5 To set the duration of the group, click the **Duration** tab.



Choose one of the following options:

➤ Run until completion.

➤ Specify the amount of time for which you want to run the group, once all the Vusers have been ramped up.

Note: The duration setting overrides the Vuser iteration settings. This means that if the duration is set to five minutes, the Vusers will continue to run as many iterations as required in five minutes, even if the run-time settings specify only one iteration.

---

6 To instruct LoadRunner to initialize Vusers before beginning to load them, select **Initialize all Vusers before Ramp-Up**.

7 Click **OK** to close the Schedule Builder and save your settings.

7

Creating a Goal-Oriented Scenario

You build a goal-oriented scenario for an application by defining the goals you want your test to achieve. This chapter describes how to create a goal-oriented scenario.

This chapter discusses:

- Defining Scenario Goals
- Assigning Properties to Scripts
- Configuring Scripts

About Planning a Goal-Oriented Scenario

In a manual scenario, you create Vuser groups, assign them scripts, load generator machines, and a number of virtual users. You then build a schedule for LoadRunner to run these groups. In a goal-oriented scenario, you define the goals you want your test to achieve, and LoadRunner automatically builds a scenario for you, based on these goals.

You can define four types of goals in a goal-oriented scenario: the number of virtual users, the number of hits per second (Web Vusers only), the number of transactions per second, or the transaction response time you want your scenario to reach. You define these goals using the Edit Scenario Goal dialog box. For more information on this dialog box, see “Defining Scenario Goals” on page 86.

Note: To run a Transactions per Second or Transaction Response Time goal type, your script must contain transactions. For each of these goal types, you define the transaction in your script that you want to test.

Virtual Users Goal Type

If you want to test how many Vusers your application can run simultaneously, it is recommended that you define a Virtual Users goal type. Running this type of goal-oriented scenario is similar to running a manual scenario. For more information on defining this goal type, see "Defining Scenario Goals" on page 86.

Hits/Transactions per Second Goal Types

If you want to test the strength of your server, it is recommended that you define a Hits per Second or Transactions per Second goal type. Specify a minimum-maximum range of Vusers for LoadRunner to run, and a Transaction Name for the Transactions per Second goal type. The Controller attempts to reach the goal you defined using a minimum number of Vusers. Note that no hits will be registered during the first minute and a half of the scenario while the Vusers are stabilizing.

If the maximum number of Vusers defined is exhausted, or if the number of hits or transactions per second does not increase after a certain number of Vuser batches have been executed, LoadRunner will stop running the scenario before the desired goal is reached. If your goal cannot be reached with the maximum number of Vusers you specified, increase this number and execute your scenario again.

Transaction Response Time Goal Type

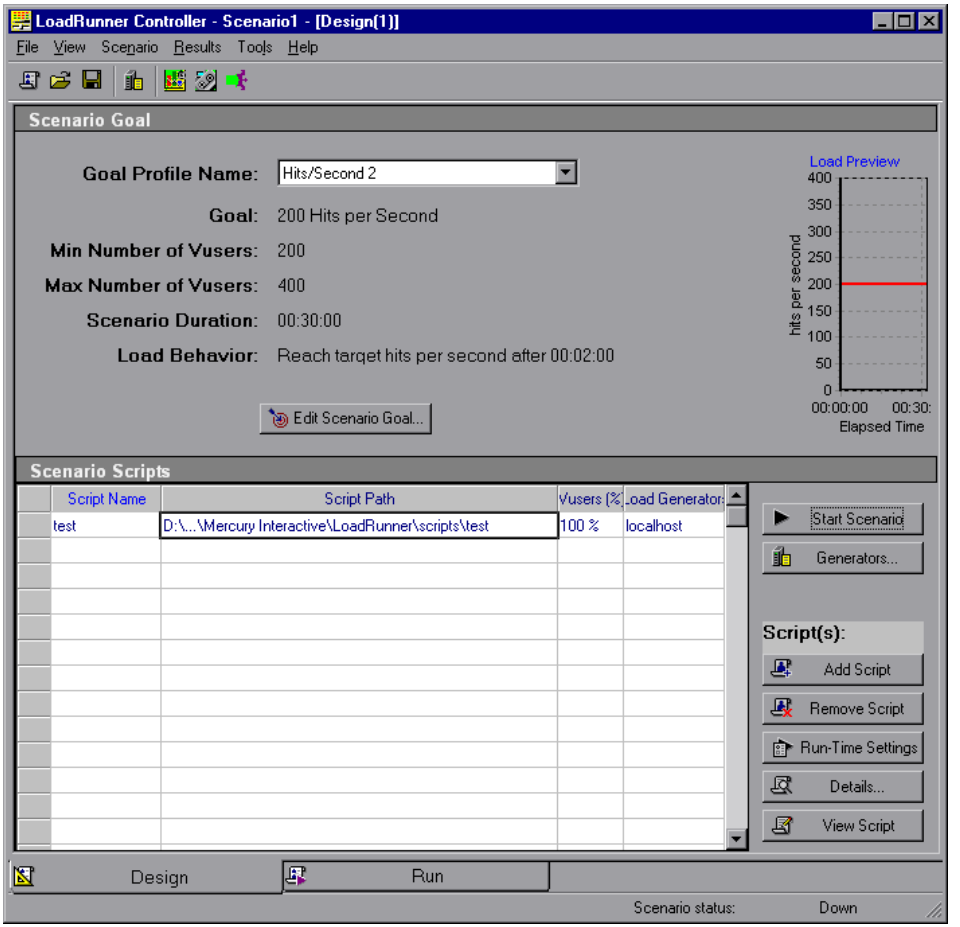
If you want to test how many Vusers can be run simultaneously without exceeding a desired transaction response time, it is recommended that you define a Transaction Response Time goal type. Specify the name of the transaction in your script that you want to test, and a minimum-maximum range of Vusers for LoadRunner to run. The transaction response time you specify should be a pre-defined threshold value. For example, if you do not want a customer to wait more than five seconds to log in to your e-commerce site, specify a maximum acceptable transaction response time of five seconds. Set the minimum and maximum number of Vusers to the minimum-maximum range of customers you want to be able to serve simultaneously.

If the scenario does not reach the maximum transaction response time that you defined, your server is capable of responding within a reasonable period of time to the number of customers you want to be able to serve simultaneously. If the defined response time is reached after only a portion of the Vusers have been executed, or if you receive a message that the defined response time will be exceeded if the Controller uses the maximum number of Vusers defined, you should consider revamping your application and/or upgrading your server software and hardware.

Note: In order for a Transaction Response Time goal-oriented scenario to be effective, you must choose your transaction carefully, ensuring that it performs effective hits on the server.

Defining Scenario Goals

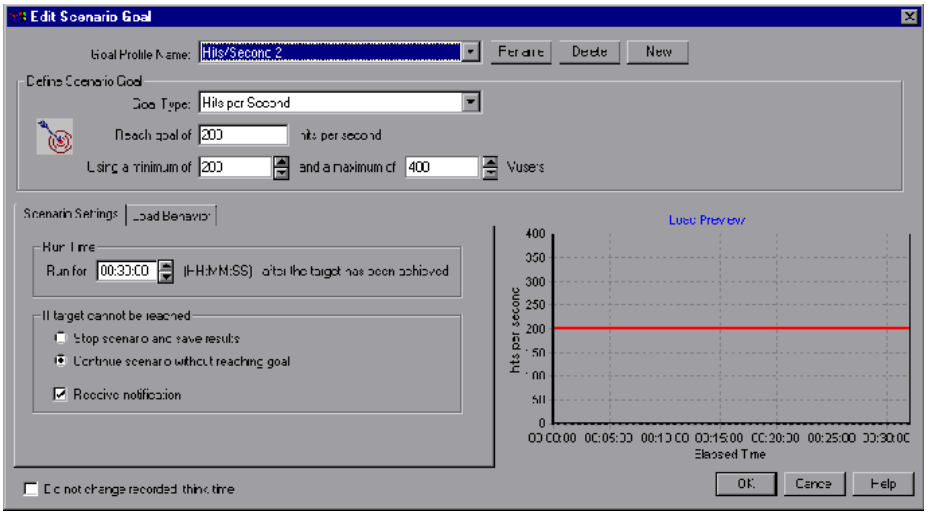
When you choose to create a goal-oriented scenario, the Controller displays the Scenario Goal and Scenario Scripts windows.



The Scenario Goal window contains basic scenario information, as defined in the Edit Scenario Goal dialog box.

To define a scenario goal:

1 Click the **Edit Scenario Goal** button. The Edit Scenario Goal dialog box opens.



2 Select a **Goal Profile Name**. To enter a new name, click **New**, type the new goal profile name in the New Goal Profile dialog box, and click **OK**. The new goal profile name appears in the selector.

To rename a goal profile, click **Rename** and enter the new goal profile name in the New Goal Profile dialog box.

To delete a goal profile, select it and click **Delete**.

3 In the Define Scenario Goal box, select a **Goal Type**.

➤ If you select **Virtual Users**, enter a target number of virtual users that you would like your scenario to reach.

➤ If you select **Hits per Second**, enter a target number of hits per second (HTTP requests per second) that you would like your scenario to reach, and select a minimum and maximum number of Vusers for the scenario.

➤ If you select **Transactions per Second**, enter a target number of transactions per second that you would like your scenario to reach, and select a minimum and maximum number of Vusers for the scenario. In addition, select a static script transaction for your scenario to test, or

enter the name of an automatic script transaction that you have recorded in the Transaction Name box.

Note: VuGen automatically defines each *Init, Action,* and *End* unit as a transaction. In a Web protocol, each recorded Web page is also defined automatically as a transaction. In addition, you can insert a static transaction in your script using the Start Transaction and End Transaction functions.

➤ If you select **Transaction Response Time**, enter a target transaction response time that you would like your scenario to reach, and select a minimum and maximum number of Vusers for the scenario. In addition, select a static script transaction for your scenario to test, or enter the name of a dynamic script transaction that you have recorded in the Transaction Name box.

4 In the Scenario Settings tab, select the amount of time that you want your scenario to run after your target has been reached.

5 Choose whether you want to stop the scenario and save the scenario results or continue the scenario, if LoadRunner does not succeed in reaching the target that you defined. If you want LoadRunner to send you an error message indicating that your target was not reached, select **Receive notification**.

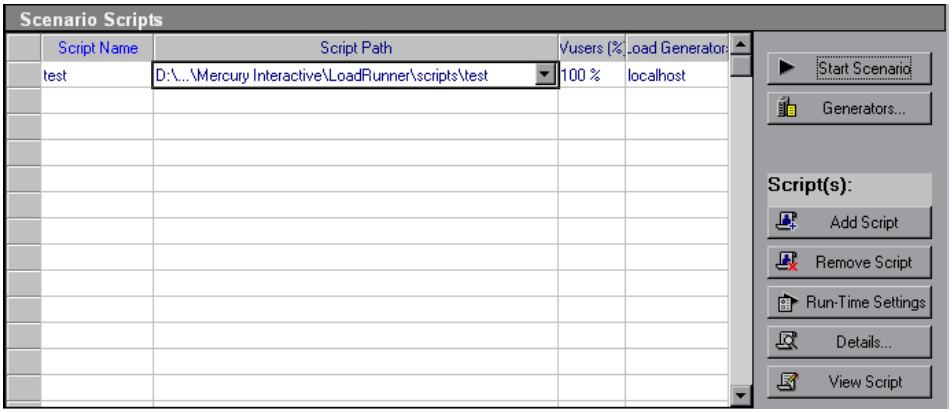
6 Select the **Load Behavior** tab. If you selected the Transactions per Second or Transaction Response Time goal types, choose whether you want LoadRunner to reach your target by automatically running a default number of Vusers in every batch, or after a certain period of the scenario has elapsed. If you selected the Virtual Users or Hits per Second goal types, choose whether you want LoadRunner to reach your target by automatically running a default number of Vusers in every batch, after a certain period of the scenario has elapsed, or by gradation (x number of Vusers/hits every x amount of time).

7 Select **Do not change recorded think time** if you want LoadRunner to run the scenario using the think time recorded in your script. Note that if you select this option, you may need to increase the number of Vusers in your scenario in order to reach your target.

8 Click **OK** to close the Edit Scenario Goal dialog box. The scenario target information you entered appears in the Scenario Goal window.

Assigning Properties to Scripts

The Scenario Scripts window displays a list of scripts you selected for the scenario.



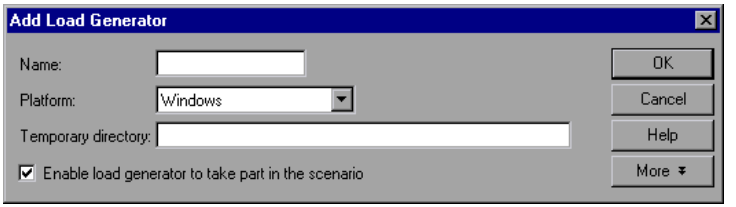
To each script you assign a load generator or load generators, as well as a percentage of the overall target number of Vusers, hits per second, transactions per second, or transaction response time that you defined in the Edit Scenario Goal dialog box.

Note: The Controller monitors a load generator machine's CPU usage and automatically stops loading Vusers on a load generator when it becomes overloaded.

To assign a load generator or load generators to a script:

1 Select **Any Load Generator**, or choose the name of the machine you are using (localhost) from the Load Generator Name list, and click **OK**.

2 Alternatively, you can choose **Add** to add a load generator to the list. The Add Load Generator dialog box opens:



Type the name of the load generator in the Name box.

By default, LoadRunner stores temporary files on the load generator during scenario execution, in a temporary directory specified by the load generator's TEMP or TMP environment variables. To override this default for a specific load generator, type a location in the Temporary Directory box.

Click **More** to expand the dialog box and show the Add Load Generator tabs. For information on configuring settings for each load generator, see "Configuring Load Generator Settings" on page 60.

Click **OK** to close the Add Load Generator dialog box. LoadRunner adds the new load generator to the Load Generator Name list. To include the new load generator in your scenario, select it from the Load Generator Name list, and click **OK**. Note that you can select multiple load generators.

Repeat the above procedure for each load generator you want to add to your scenario.

Configuring Load Generators

You can set a load generator's attributes while adding it to the load generator list, or modify the attributes of an existing load generator at any time, using the Load Generators dialog box. You can also use the Load Generators dialog box to indicate which load generators will run Vusers in the scenario. For example, if a load generator is unavailable for a particular scenario run, you can use the Load Generators dialog box to exclude it temporarily instead of removing it entirely from your list of load generators. For instructions on using the Load Generators dialog box, see "Configuring Load Generators" on page 57. To configure additional load generator settings, see "Configuring Load Generator Settings" on page 60.

To configure global settings for all load generators participating in the scenario, use LoadRunner's Options dialog box. For more information, see Chapter 9, "Configuring a Scenario."

Assigning a Percentage of your Defined Target to Scripts

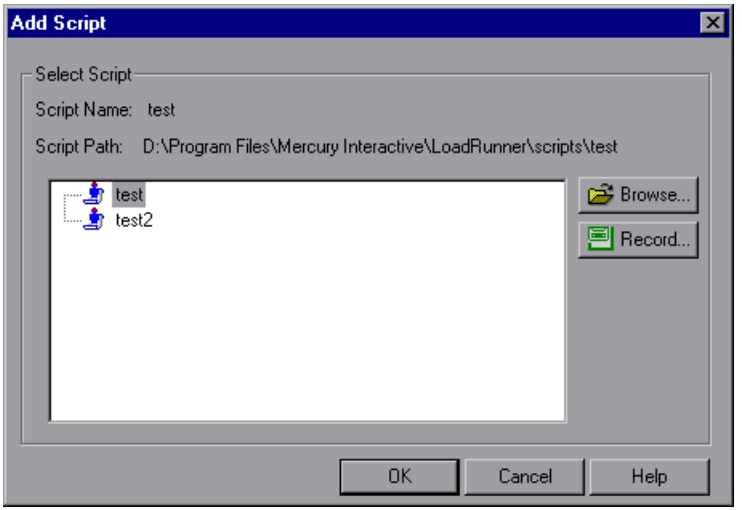
In the Edit Scenario Goal dialog box, you specify the maximum number of Vusers, hits per second, transactions per second, or transaction response time you want LoadRunner to reach during the scenario. In the Scenario Scripts window, you stipulate the percentage of this total that each script in the scenario should reach. Enter a percentage in the **% of Target** column for each script included in the scenario.

Configuring Scripts

You can add a script to the Scenario Scripts list using the Add Script dialog box. Once you have added the script to the list, you can view the details of the script you selected, edit the script, or change its run-time settings.

To add a script:

1 Click the **Add Script** button on the right of the Scenario Scripts window, or right-click within a column and select **Add Script**. The Add Script dialog box opens.



2 Click the **Browse** button to the right of the path box. The Open Test dialog box opens.

Select the path and file name of the new script. To select a VB Vuser script, browse to locate the .usr file.

Note: When you specify the location of a script, you can specify a location that is relative to the current scenario directory. For details, see "Using Relative Paths for Scripts" on page 70.

---

3 Click **Open** to select the files. The Open Test dialog box closes, and the new script name appears in the Add Script dialog box.

4 Click **OK** to close the Add Script dialog box and enter the new script information in the Scenario Scripts window.

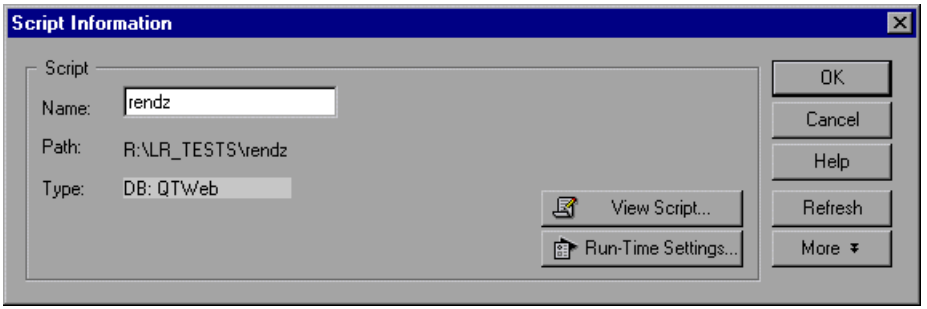
Note: A script's rendezvous points are disabled in a goal-oriented scenario.

To delete a script:

Click the **Remove Script** button on the right of the Scenario Scripts window, or right-click the script and select **Remove Script**.

To view script details:

1 Click the **Details** button on the right of the Scenario Scripts window, or right-click a script and select **Details**. The Script Information dialog box opens, displaying the Path, Name, and Type of the script you selected.



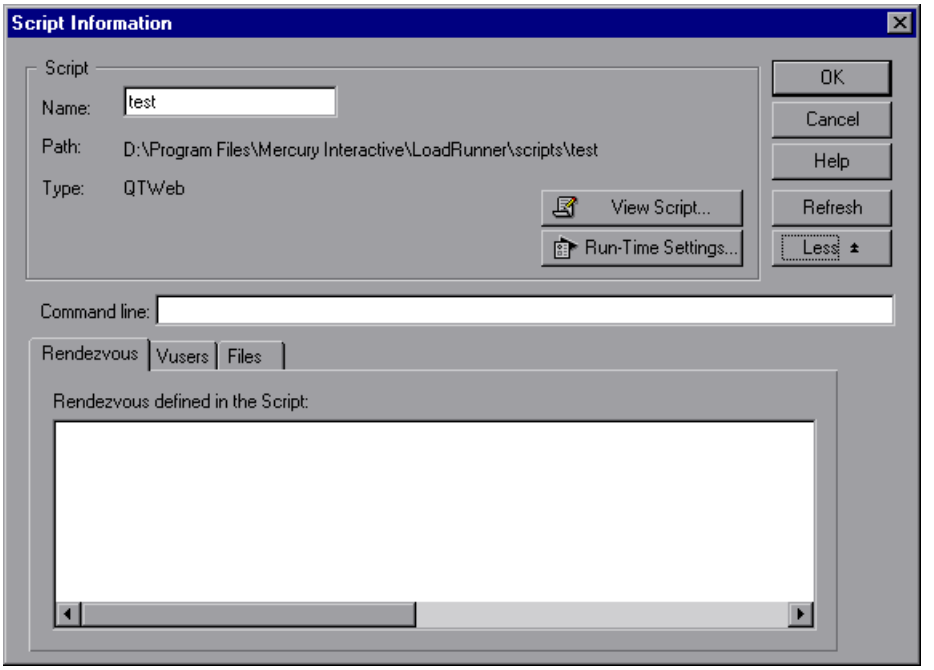
2 Click **Run-Time Settings** to set the script's run-time settings (optional). These settings let you customize the way the Controller executes a Vuser script. You can also modify the run-time settings using the Vuser Generator (VuGen). Note that several protocols, such as Web and Java, have specific settings. For information on configuring the run-time settings, refer to the *Creating Vuser Scripts* guide.

Note: When you open the Run-Time Settings dialog box from the Controller, the settings you previously set using VuGen appear. If you modify these settings from the Controller, LoadRunner runs the script using the modified settings.

---

3 To edit the script, click **View Script**. The script generation tool, VuGen, opens. For more information on editing scripts, see the *Creating Vuser Scripts* guide.

4 Click **More** to expand the Script Information dialog box and view additional script information.



5 In the Command Line box, type any command line options to use when running the script. For example: -x value -y value

For information about passing command line argument values to a script, refer to the *Creating Vuser Scripts* guide.

6 To see the rendezvous points declared in the selected script, click the **Rendezvous** tab.

7 To see the list of Vusers associated with the selected script, click the **Vusers** tab. If you have not yet created Vusers, the box will be empty.

8 To see the list of files used by the script, click the **Files** tab. By default this list shows all files in the script's directory (only after your script has been added to the script list). These files include the configuration settings file, the init, run, and end portions of the script, the parameterization definitions file, and the *usr* file. To add a file to the list, click **Add** and add the file name. Note that you can delete the files that you add, but not the other files listed.

9 Click **OK** to close the Script Information dialog box.

Note: If you use VuGen to make changes to a script while the Controller is running, click the **Refresh** button in the Controller's Script Information dialog box to update the script details in the scenario.

8

Using Rendezvous Points

LoadRunner allows you to check your system's response under specific load. To do this, you can use *rendezvous points* to cause multiple Vusers to perform tasks at exactly the same time, thereby creating intense user load on the server.

This chapter describes:

- Setting the Rendezvous Attributes
- Setting the Rendezvous Behavior
- Disabling and Enabling Rendezvous Points
- Disabling and Enabling Vusers at Rendezvous Points
- Viewing Rendezvous Information

About Using Rendezvous Points

During a scenario run you can instruct multiple Vusers to perform tasks simultaneously by using rendezvous points. A rendezvous point creates intense user load on the server and enables LoadRunner to measure server performance under load.

Suppose you want to measure how a web-based banking system performs when ten Vusers simultaneously check account information. In order to emulate the required user load on the server, you instruct all the Vusers to check account information at exactly the same time.

You ensure that multiple Vusers act simultaneously by creating a *rendezvous point*. When a Vuser arrives at a rendezvous point, it is held there by the Controller. The Controller releases the Vusers from the rendezvous either

when the required number of Vusers arrive, or when a specified amount of time has passed. For details on the release criteria, see "Setting the Rendezvous Behavior," on page 100.

You define rendezvous points in the Vuser script. For information about inserting rendezvous points into Vuser scripts, refer to the *Creating Vuser Scripts* guide.

Using the Controller, you can influence the level of server load by selecting:

- which of the rendezvous points will be active during the scenario
- how many Vusers will take part in each rendezvous

For example, to test a bank server you could create a scenario that contains two rendezvous points. The first rendezvous ensures that one thousand Vusers simultaneously deposit cash. The second rendezvous ensures that another thousand Vusers simultaneously withdraw cash. If you want to measure how the server performs when only five hundred Vusers deposit cash, you can deactivate (disable) the "withdraw" rendezvous, and instruct only five hundred Vusers to participate in the "deposit" rendezvous.

The following procedure outlines how to control load peaks on the server:

1. **Create the Vuser scripts, inserting the necessary rendezvous points**.
2. **Create a scenario.**

   When you add a Vuser group to a scenario, LoadRunner scans the group's associated script for the names of the rendezvous points and adds them to the list in the Rendezvous window (**Scenario** > **Rendezvous**). If you create another Vuser group that runs the same script, the Controller adds the new Vusers to the rendezvous and updates the list.
3. **Set the level of emulated user load.**

   You determine the exact level of load by selecting the rendezvous points that will take part in the scenario, and how many Vusers will participate in each rendezvous.

4 Set the attributes for the rendezvous (optional).

For each rendezvous you can set the *timeout* and *policy attributes*. For more information, see "Setting the Rendezvous Behavior," on page 100.

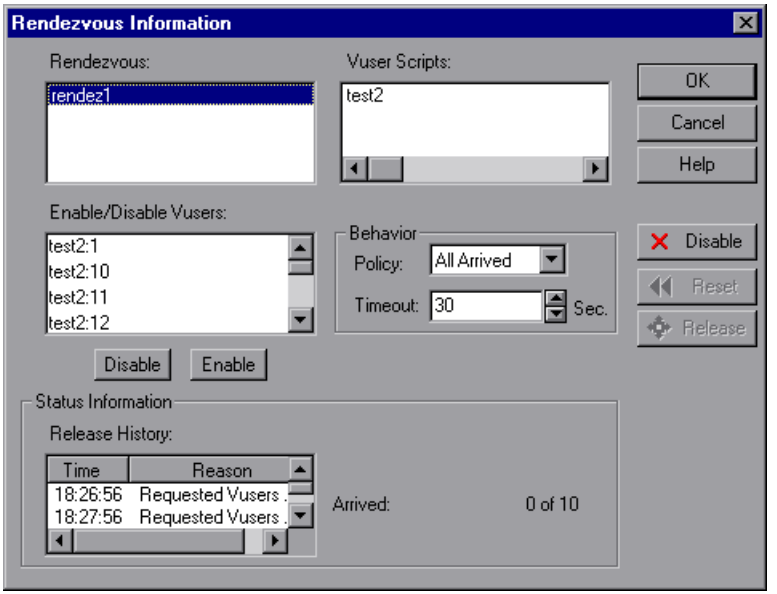
5 Run the scenario.

Setting the Rendezvous Attributes

You can set the following rendezvous attributes from the Rendezvous Information dialog box (**Scenario** > **Rendezvous**):

- Timeout
- Rendezvous Policy
- Enabling and Disabling Rendezvous
- Enabling and Disabling Vusers

In addition, the dialog box displays general information about the rendezvous point: which script is associated with the rendezvous and release history.



For information on manipulating the Vusers during scenario execution using the Release command, see Chapter 12, "Running a Scenario."

Setting the Rendezvous Behavior

Setting the rendezvous behavior determines how the Vusers handle a rendezvous point. You set the following behavior attributes for each rendezvous:

timeout	how long the Controller waits before releasing Vusers from a rendezvous.
policy	sets how many Vusers will be released from a rendezvous at a time.

Setting the Timeout Behavior Attribute

After each Vuser arrives at the rendezvous point, LoadRunner waits up to the maximum *timeout* period you set for the next Vuser to arrive. If the next Vuser does not arrive within the *timeout* period, then the Controller releases all the Vusers from the Rendezvous. Each time a new Vuser arrives, the timer is reset to zero. The default *timeout* is thirty seconds. You set a *timeout* for each rendezvous point.

To set a timeout:

1. Choose **Scenario** > **Rendezvous**. The Rendezvous Information dialog box opens.
2. Select the rendezvous for which you want to set a timeout from the Rendezvous list.
3. In the Behavior section, locate the Timeout box. Enter a timeout value—the default is 30 seconds.
4. Click **OK** to close the dialog box and set the timeout for the rendezvous.

Setting the Release Policy Attribute

The policy attribute determines how the Controller releases Vusers from the rendezvous. For each rendezvous you can set the following policies:

All Arrived	Instructs the Controller to release the Vusers from the rendezvous only when all the Vusers included in the rendezvous arrive. All the Vusers are released simultaneously. The default policy is *All Arrived*.
Quota	Sets the number of Vusers that must arrive at a rendezvous point before the Controller releases the Vusers. For instance, suppose that you are testing a scenario of fifty Vusers and that you want a particular operation to be executed simultaneously by ten Vusers. You can designate the entire scenario as participants in the rendezvous and set a quota of ten Vusers. Every time ten Vusers arrive at the rendezvous, they are released.

To set the policy attribute:

1. Choose **Scenario** > **Rendezvous**. The Rendezvous Information dialog box opens.
2. Click a rendezvous in the Rendezvous list.
3. In the Behavior section, locate the Policy box.
4. To set the policy to All Arrived, select **All Arrived** from the list. The new policy appears in the Policy field.
5. To set the policy to Quota, select **Quota** from the list. The Rendezvous - Quota dialog box opens.

Enter the number of Vusers in the Quota box.

6. Click **OK** to close the dialog box. The new policy appears in the Policy field.

Disabling and Enabling Rendezvous Points

You can temporarily disable a rendezvous and exclude it from the scenario. By disabling and enabling a rendezvous, you influence the level of server load.

You use the Disable and Enable buttons in the Rendezvous Information dialog box, to change the status of a rendezvous.

To disable a rendezvous:

1. Choose **Scenario** > **Rendezvous**. The Rendezvous Information dialog box opens.
2. Select the rendezvous you want to disable.
3. 

Click the **Disable** button. The button changes to **Enable** and the rendezvous becomes disabled.

To enable a rendezvous:

1. Choose **Scenario** > **Rendezvous**. The Rendezvous Information dialog box opens.
2. Select the disabled rendezvous that you want to enable.
3. 

Click the **Enable** button. The button changes to **Disable** and the rendezvous becomes enabled.

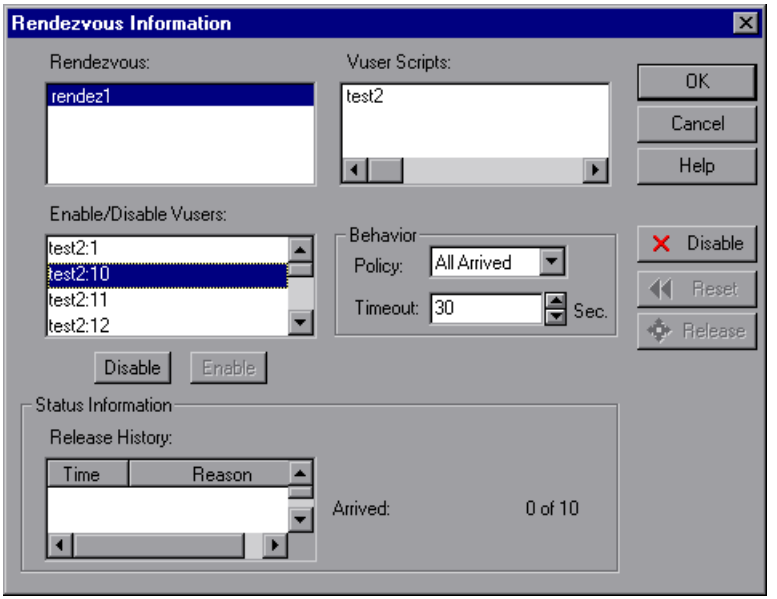
Disabling and Enabling Vusers at Rendezvous Points

In addition to disabling a rendezvous point for all Vusers in a scenario, LoadRunner lets you disable it for specific Vusers. By disabling Vusers at a rendezvous, you temporarily exclude them from participating in the rendezvous. Enabling disabled Vusers returns them to the rendezvous. You use the Disable and Enable commands to specify which Vusers will take part in a rendezvous.

To disable a Vuser in a rendezvous:

1 Choose **Scenario** > **Rendezvous**. The Rendezvous Information dialog box opens.

2 Select the rendezvous for which you want to disable Vusers from the Rendezvous list.



3 In the section Enable/Disable Vusers, select the Vuser(s) you want to exclude from the rendezvous. Select multiple Vusers using the CTRL key.

4 Click **Disable** (the button directly below the Vuser list). The disabled Vusers change from black to gray and will not take part in the rendezvous.

5 To enable a Vuser, select it and click **Enable**.

Viewing Rendezvous Information

During and after a scenario, you can view the rendezvous status in the Rendezvous Information dialog box. The following information is provided:

Time: The time the rendezvous was released.

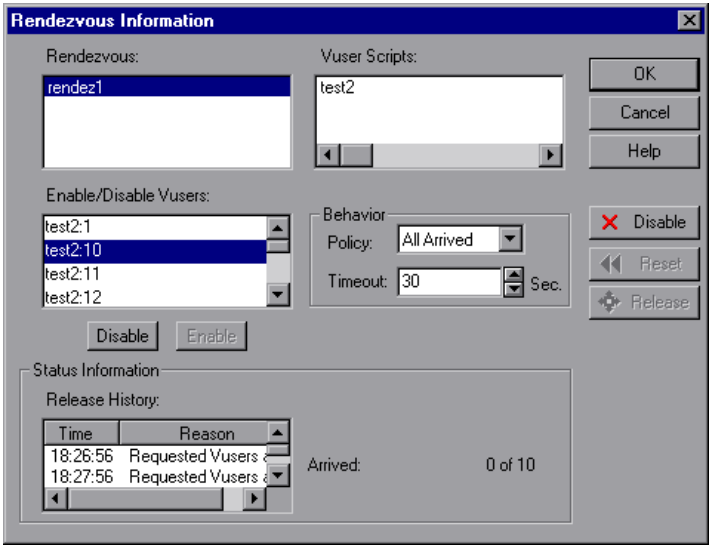
Reason: The reason the rendezvous released the Vusers. The possible values are *Timeout* or *Arrived*.

Arrived: The number of Vusers that arrived at the rendezvous point, out of the total number of Vusers assigned to the rendezvous.

To view rendezvous information:

1 Choose **Scenario** > **Rendezvous**. The Rendezvous Information dialog box opens.

2 Select the rendezvous whose information you want to view. The rendezvous status is displayed in the Status Information section.



9

Configuring a Scenario

You can configure how load generators and Vusers behave when you run a scenario so that the scenario accurately emulates your working environment.

This chapter describes:

- Configuring Scenario Run-Time Settings
- Saving Messages to the Output Files
- Setting Timeout Intervals
- Setting the Run-Time File Location
- Specifying Path Translation

About Configuring a Scenario

Before you run a scenario, you can configure both the load generator and Vuser behaviors for the scenario. Although the default settings correspond to most environments, LoadRunner allows you to modify the settings in order to customize the scenario behavior. The settings apply to all future scenario runs and generally only need to be set once.

The settings described in this chapter apply to all the load generators in a scenario. To change the settings for individual load generator machines, refer to Chapter 5, "Creating a Manual Scenario." If the global scenario settings differ from those of an individual load generator, the load generator settings override them.

The settings discussed in this chapter are unrelated to the Vuser run-time settings. These settings, which apply to individual Vusers or scripts, contain

information about logging, think time, and the network, the number of iterations, and the browser. For information on setting the run-time settings, see the *Creating Vuser Scripts* guide.

Note: Vuser scripts have individual run-time setting defaults for VuGen and the Controller, to support the debugging environment of VuGen and the load testing environment of the Controller.

These are the default settings for Vuser scripts in VuGen and the Controller:

Think Time - Off in VuGen and 'Replay as recorded' in the Controller.
Log - 'Brief' in VuGen and off in the Controller.
Enable Loading of Web Resources - On in both VuGen and the Controller.

For information on setting the options for online monitors, see Chapter 14, "Online Monitoring."

The LoadRunner Expert mode allows you to configure additional settings for the LoadRunner agent and other LoadRunner components. For more information, see Appendix C, "Working in Expert Mode."

Configuring Scenario Run-Time Settings

The scenario run-time settings relate to:

- Vuser Quotas
- Random Sequence Seed

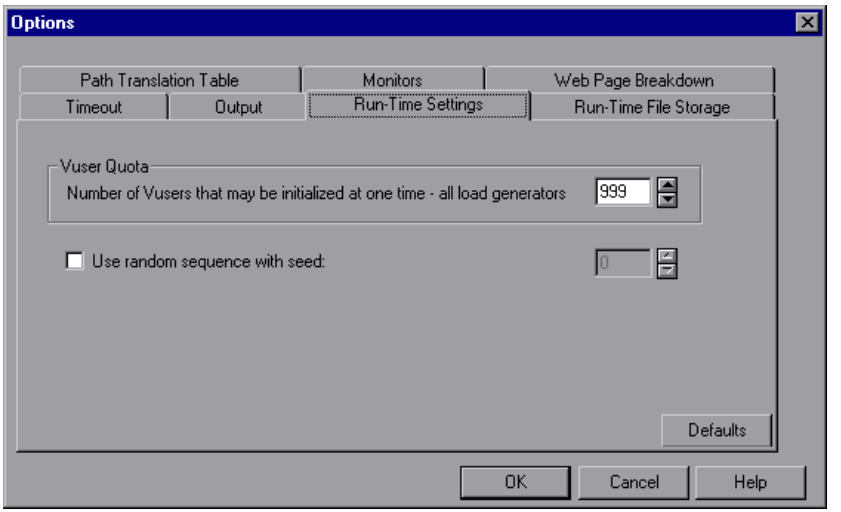
Vuser quotas: To prevent your system from overloading, you can set quotas for Vuser activity. The Vuser quotas apply to Vusers on all load generators. You can limit the number of Vusers initialized at one time (when you send an Initialize command).

Random sequence seed: LoadRunner lets you set a seed number for random sequencing. Each seed value represents one sequence of random values used for test execution. Whenever you use this seed value, the same sequence of

values is assigned to the Vusers in the scenario. This setting applies to parameterized Vuser scripts using the Random method for assigning values from a data file. Enable this option if you discover a problem in the test execution and want to repeat the test using the same sequence of random values. You can set a seed value for LoadRunner's Random Sequencing.

To set the scenario run-time settings:

1 Choose **Tools** > **Options**. The Options dialog box opens. Click the **Run-Time Settings** tab.



2 To set a Vuser quota, specify the desired value.

3 To specify a seed value for a random sequence, select the **Use random sequence with seed** check box and enter the desired seed value.

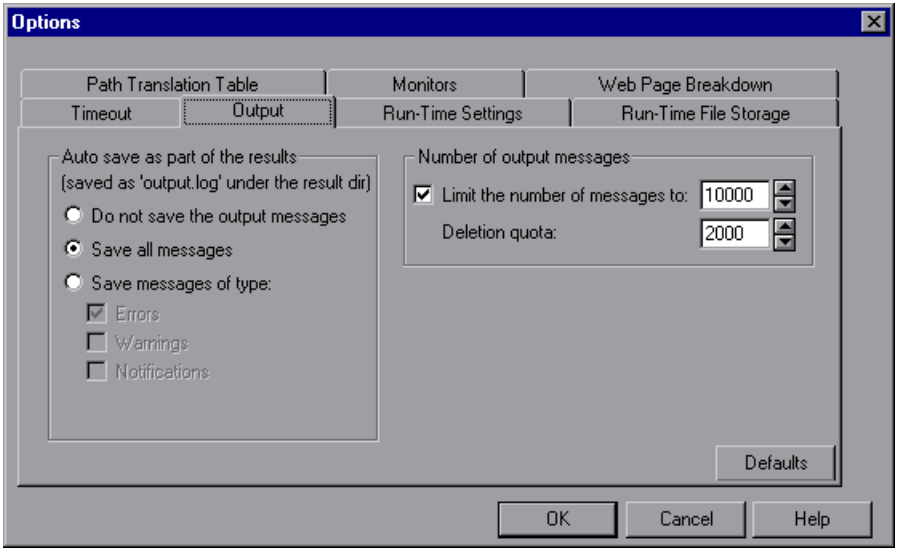
Saving Messages to the Output Files

By default, LoadRunner generates an output file, *output.log*, and stores it in the script directory. This file contains error, warning, and notification messages issued by the Controller during scenario execution. Using the Options dialog box, you can select the type of messages to save to the output file. In addition, you can disable the logging entirely.

You can specify the number of messages that will appear in the output window. If the number of messages exceeds the limit, the messages are deleted. A deletion quota value specifies the number of messages that may be deleted from the output. The deletion quota overrides the limit number. For example, assume that you limit the number of messages to 500 and set the deletion quota to 50. If there are 600 messages, 550 will appear in the output.

To configure the output logging:

1 Choose **Tools** > **Options**. The Options dialog box opens. Click the **Output** tab.



2 Select **Do not save the output messages** to disable writing to the *output.log* file.

3 Select **Save all messages** to instruct LoadRunner to save message of all types to the log file.

4 Select **Save messages of type** to exclude a specific message type. Then select the message type(s) to include in the log file.

5 To limit the number of output messages, select **Limit the number of messages to** and specify a value.

6 To set a deletion quota, specify a value in the Deletion quota box.

The LoadRunner Expert mode allows you to configure additional output-related settings. For more information, see Appendix C, "Working in Expert Mode."

Setting Timeout Intervals

LoadRunner enables you to set the timeout interval for commands and Vuser elapsed time.

The command timeouts are the maximum time limits for various LoadRunner commands. When a command is issued by the Controller, you set a maximum time for the load generator or Vuser to execute the command. If it does not complete the command within the timeout interval, the Controller issues an error message.

The command timeouts relate to load generators and Vusers. The load generator commands for which you can specify a timeout interval are Connect and Disconnect. The Vuser commands for which you can specify a timeout interval are Init, Run, Pause, and Abort.

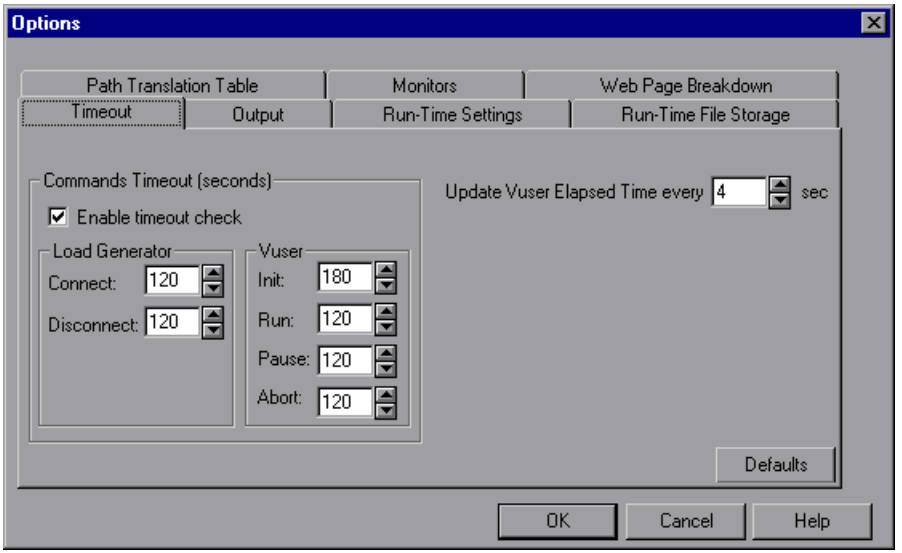
For example, the default *Init* timeout is 180 seconds. If you select a Vuser and click the **Initialize** button, LoadRunner checks whether the Vuser reaches the Ready state within 180 seconds; if it does not, the Controller issues a message indicating that the Init command timed out.

In the Vuser view, the Elapsed column (the last column) indicates the amount of time that elapsed from the beginning of the scenario. You can specify the frequency in which LoadRunner updates this value. The default is 4 seconds.

Note: LoadRunner's calculations consider the number of active Vusers and their influence on the timeout values. For example, 1000 Vusers trying to initialize will take much longer than 10 Vusers. LoadRunner adds an internal value, based on the number of active Vusers, to the specified timeout value.

To set timeout intervals:

1 Choose **Tools** > **Options**. The Options dialog box opens. Click the **Timeout** tab.



2 Clear the **Enable timeout check** check box to disable the timeout test. LoadRunner waits an unlimited time for the Initialize, Run, Pause, and Stop commands to be executed.

3 To specify a command timeout interval, select the **Enable timeout check** check box and specify the appropriate timeouts.

4 Specify the frequency at which LoadRunner updates the Elapsed time, in the **Update Vuser Elapsed Time every** box.

Setting the Run-Time File Location

When you run a scenario, by default the run-time files are stored locally on each Vuser load generator (the machine running the Vuser script). The default location of the files is under the temporary directory specified by the load generator's environment variables (on Windows, TEMP or TMP and on UNIX, $TMPDIR or $TMP). If no environment variable is defined, the files are saved to the /tmp directory.

Note: The run-time file storage settings that are described in this chapter apply to all the load generators in a scenario. You can change the settings for individual load generator machines as described in "Configuring Load Generators" on page 57.

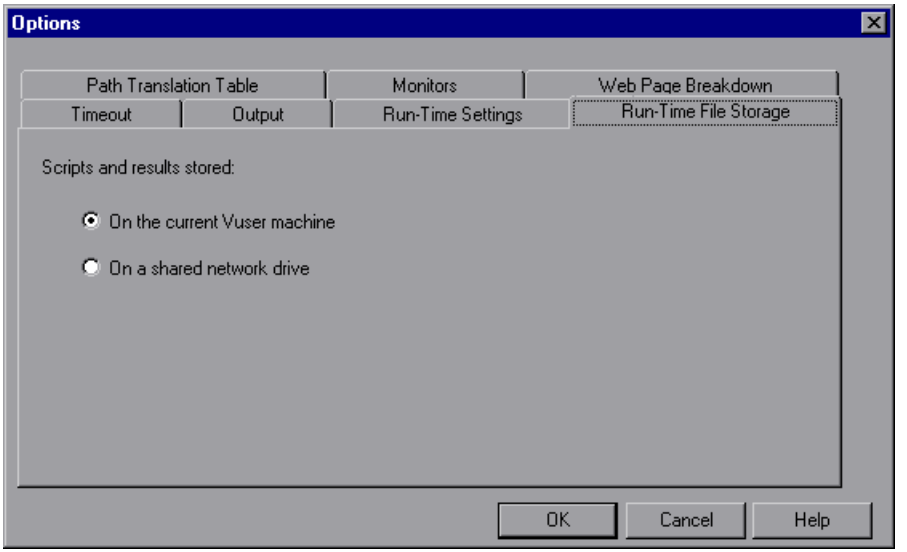
The primary run-time files are Vuser script and result files:

Script files: When you run a Vuser, the Controller sends a copy of the associated Vuser script to the Vuser load generator. The script is stored in the load generator's temporary run-time directory.

Result files: While you run a scenario, the participating Vusers write their results to the temporary run-time file directory. After scenario execution, these result files are collated or consolidated—results from all of the load generators are transferred to the results directory. You set the location of the results directory as described in Chapter 12, "Running a Scenario." After collating the results, the temporary run-time directory is deleted.

To specify where LoadRunner stores run-time files:

1 Choose **Tools** > **Options**. The Options dialog box opens. Click the **Run-Time File Storage** tab.



By default, the **On the current Vuser machine** option is selected. This means that all run-time files—including result files and script files—are stored on the Vuser load generators. The only exception is for Vusers running on the local load generator (Controller machine), where you must use the shared drive option.

2 To store script and result files on a shared network drive, click **On a shared network drive**. To set the exact location on the network drive, see Chapter 10, "Preparing to Run a Scenario."

If you select to save results to a shared network drive, you may need to perform path translation. Path translation ensures that the specified results directory is recognized by the remote load generator. For information about path translation see Appendix B, "Performing Path Translation."

If you specify that all Vusers access their Vuser scripts directly at some shared location, no transfer of script files occurs at run time. This alternative method may be useful in either of the following situations:

- The file transfer facility does not work.
- The Vuser script files are large and therefore take a long time to transfer. Remember that Vuser script files are transferred only once during a scenario.

This alternate method often necessitates path translation. For details, see Appendix B, "Performing Path Translation."

3 Click **OK** to close the dialog box.

Note: If you choose to save result files on the Vuser load generators, you must collate the results before you can perform any analysis. You can wait for LoadRunner to collate the results when you launch the Analysis tool, or you can collate results by selecting **Results** > **Collate Results**. Alternatively, select **Results** > **Auto Collate Results** to automatically collate the results at the end of each scenario run.

Specifying Path Translation

If you specified a shared network drive for run-time file storage, (see "Setting the Run-Time File Location" on page 111), you may need to perform *path translation*. Path translation is a mechanism used by LoadRunner to convert a remote path names. A typical scenario may contain several load generator machines that map the shared network drive differently. For more information, see the Appendix B, "Performing Path Translation."

10

Preparing to Run a Scenario

Before you run a scenario, you specify a location for the scenario results and other run-time related settings.

This chapter describes:

- Specifying a Results Location
- Results Directory File Structure
- Collating Results
- Setting Scenario Summary Information

About Preparing to Run a Scenario

Before you run a scenario, you need to specify the location of the results (mandatory), assign a name to the results, schedule the scenario, and provide scenario summary information. In addition, you can specify the applications to invoke at the start of a scenario.

Although most of the pre-scenario settings are optional, by using them you can enhance the testing process. These values are scenario specific—you can set different values for each LoadRunner scenario.

For information on one-time configuration settings such as timeout, output, and quotas, see Chapter 9, "Configuring a Scenario."

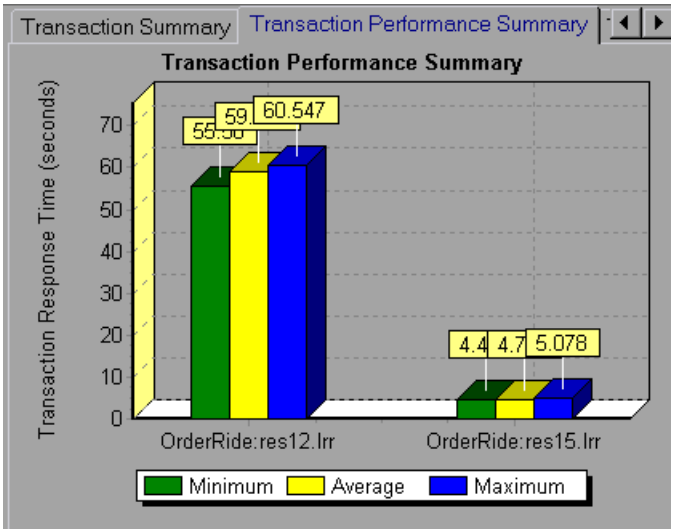
Specifying a Results Location

When you run a scenario, by default the run-time files are stored locally on each load generator. After the scenario, the results are collated together and processed on the Controller machine. Alternatively, you can instruct LoadRunner to save the results on a shared network drive. For information about specifying a file storage method, see the Run-Time File Storage settings in Chapter 9, "Configuring a Scenario."

You can also use Mercury Interactive's test management program, TestDirector, to store results to a project. For information on connecting to a TestDirector database, see Chapter 11, "Managing Scenarios Using TestDirector."

LoadRunner allows you to give descriptive names to each result set. This is especially useful for cross results analysis, in which LoadRunner superimposes the results of several scenario runs in a single graph and lets you compare the results of multiple scenario runs. The descriptive graph names enable you to distinguish between the results of the multiple runs.

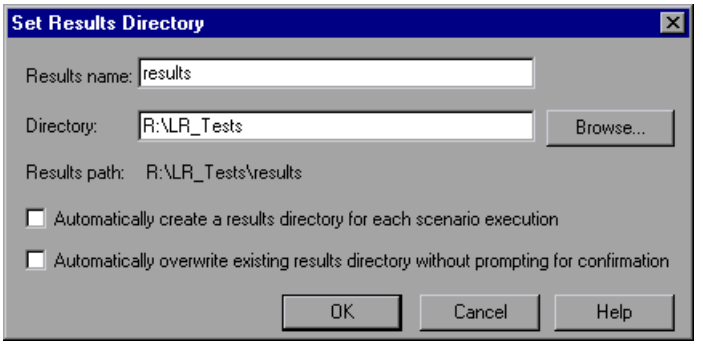
In the example below, the results of two scenario runs are superimposed. The result sets are *res12*, and *res15*.



For more details on cross result graphs, see the *LoadRunner Analysis User's Guide*.

To specify where results are stored (not in TestDirector):

1 Choose **Results** > **Results Settings**. The Set Results Directory dialog box opens.



2 In the Results Name box, enter a name for the results. Avoid using the same name with different paths, since the names will appear identical on the graphs.

3 In the Directory box, type the full path of the results directory. If you are using the default file storage setting (local machine), specify a directory in which to store all of the collated results after the scenario run. If you specified a shared network drive as the file storage method, specify the directory to which Vuser groups should write during scenario execution.

Using the results name from step 2, the Controller creates a subdirectory within the results directory. All results are saved within this subdirectory.

4 Select the appropriate check box for subsequent executions: **Automatically create a results directory for each scenario execution** or **Automatically overwrite existing results directory without prompting for confirmation**.

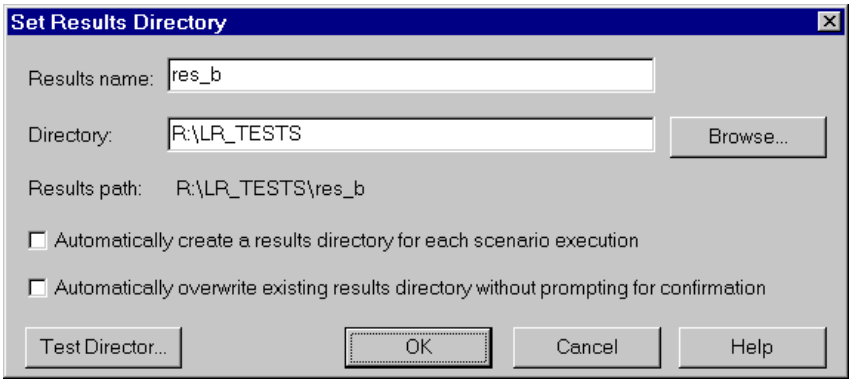
5 Click **OK** to save the results directory setting.

To specify where results are stored (using TestDirector):

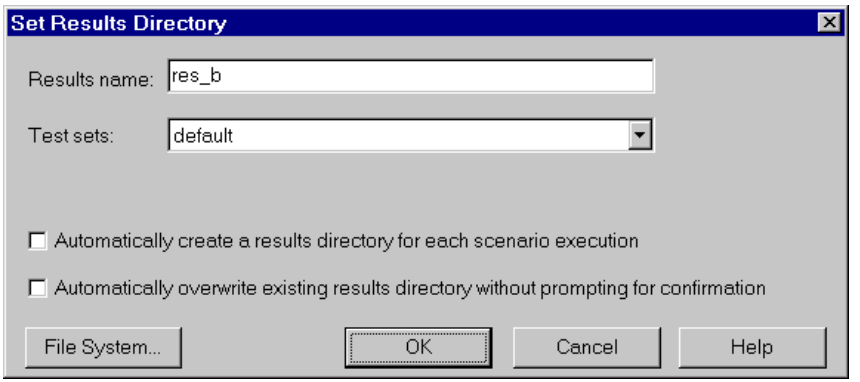
1 Choose **Tools** > **TestDirector Connection** to open a connection to a TestDirector server and database.

2 Open a scenario from the database (**File** > **Open**), or save the current scenario to the TestDirector database (**File** > **Save As**).

3 Choose **Results** > **Results Settings**. The Set Results Directory dialog box opens.



4 Click **TestDirector**. The Directory box changes to Test sets.



5 In the Results Name box, enter a name for the results. Avoid using the same name with different paths, since the names will appear identical on the graphs.

6 Select an existing test set from the Test Sets list, or specify a new one.

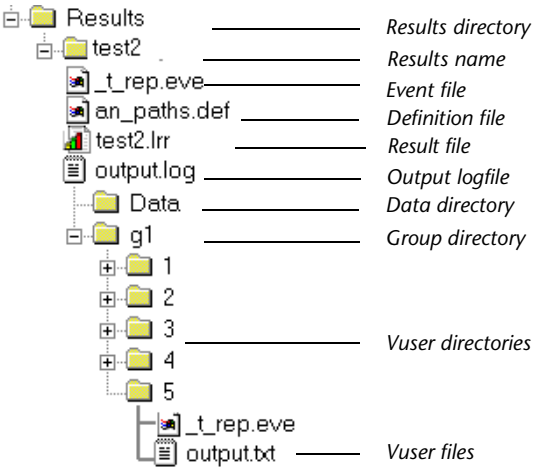
7 Select the appropriate check box for subsequent executions: **Automatically create a results directory for each scenario execution** or **Automatically overwrite existing results directory without prompting for confirmation**.

8 Click **OK** to save the results settings.

Results Directory File Structure

When you set the results directory, you also specify a results name. LoadRunner creates a subdirectory using the results name, and places all of the data it gathers in that directory. Every set of results contains general information about the scenario in a result file (*.lrr*) and an event (*.eve*) file.

During scenario execution, LoadRunner also gathers data from each Vuser and stores it in an event file *_t_rep.eve* and an output file *output.txt*. LoadRunner creates a directory for each group in the scenario and a subdirectory for each Vuser. A typical result directory has the following structure:



- *t_rep.eve* in the main result directory contains Vuser and rendezvous information.
- **.def* are definition files for graphs that describe the online and other custom monitors.
- *results_name.lrr* is the LoadRunner Analysis document file.

- *output.log* contains output information about the scenario generated during test execution.
- The *Data* directory contains the database created by the Analysis (from the results files).
- *g1* is a group directory. A separate directory exists for each Vuser group that runs in the scenario. Each group directory consists of Vusers subdirectories.
- *t_rep.eve* in each Vuser directory contains transaction information.
- *output.txt* in each Vuser directory contains output information generated during replay.

When you generate analysis graphs and reports, the LoadRunner Analysis engine copies all of the scenario result files (*.eve* and *.lrr*) to a database. Once the database is created (and stored in the *Data* directory), the Analysis works directly with the database and does not use the result files.

For information on LoadRunner Analysis, see the *LoadRunner Analysis User's Guide*

Collating Results

When you run a scenario, by default all Vuser information is stored locally on each load generator. After scenario execution, the results are automatically collated or consolidated—results from all of the load generators are transferred to the results directory. You set the location of the results directory as described in "Specifying a Results Location," on page 116.

To disable automatic collation, choose **Results** > **Auto Collate Results** to clear the check mark adjacent to the option. To manually collate results, choose **Results** > **Collate Results**. If your results have not been collated, then LoadRunner will automatically collate the results before generating the analysis data.

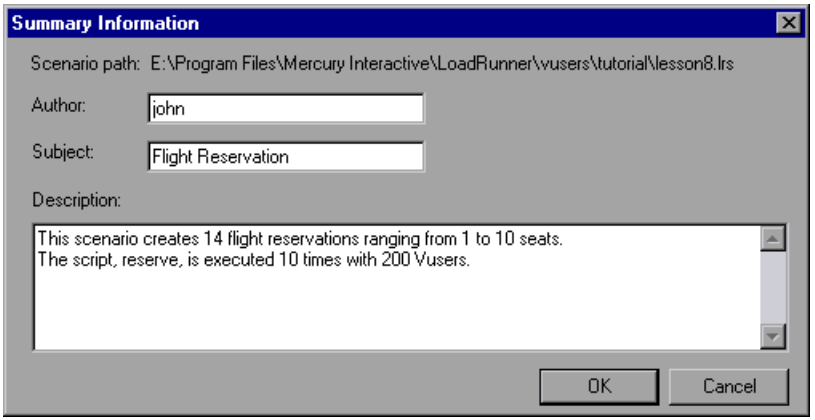
Note: If you have selected to store all the scenario results directly to a shared network drive, then collation of the results is not required. See "About Configuring a Scenario," on page 105 for details on changing how results are stored.

Setting Scenario Summary Information

The Controller allows you to provide a detailed description of the scenario. In addition, you can specify the author's name and a subject title for the scenario. Whenever you open this scenario, the summary information is available to you.

To set the scenario summary information:

1 Choose **Scenario** > **Summary Information**. The Summary Information box opens.



2 In the Author box, enter the name of the author.

3 In the Subject box, enter a subject name or short title for the scenario.

4 In the Description box, enter a detailed description about the scenario.

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

11

Managing Scenarios Using TestDirector

LoadRunner integration with TestDirector lets you manage LoadRunner scenarios using TestDirector. TestDirector helps you organize and manage all scripts, scenarios, and results using a repository.

This chapter describes:

- Opening a Connection to a TestDirector Project
- Opening a Scenario Using TestDirector
- Saving Scenarios to the TestDirector Project
- Saving Results to the TestDirector Project
- Adding Vuser Scripts with TestDirector

For detailed information on using TestDirector to manage your scenarios, refer to the *TestDirector User's Guide*.

About Managing Scenarios Using TestDirector

LoadRunner works together with TestDirector, Mercury Interactive's test management tool. TestDirector provides an efficient method for storing and retrieving scenarios and collecting results. You store scenarios and results in a TestDirector repository and organize them into unique groups.

The following sections describe how to use the TestDirector repository to:

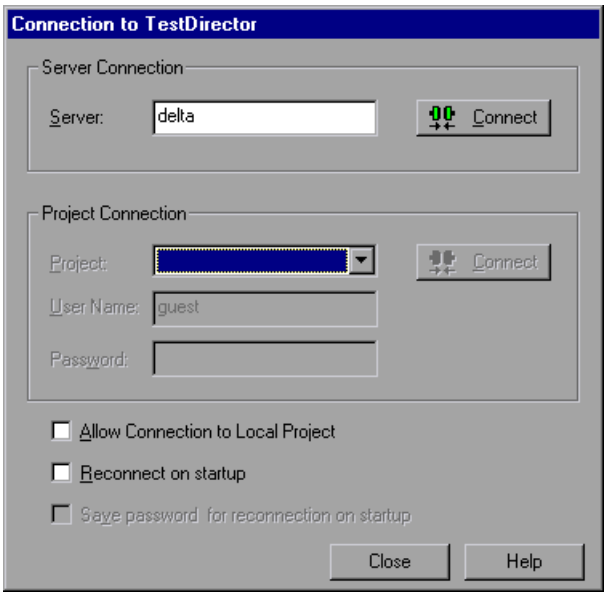
- open scenarios
- save scenarios
- save results
- open scripts

Opening a Connection to a TestDirector Project

Before you can work with a TestDirector project, you must open a connection to the server hosting the project.

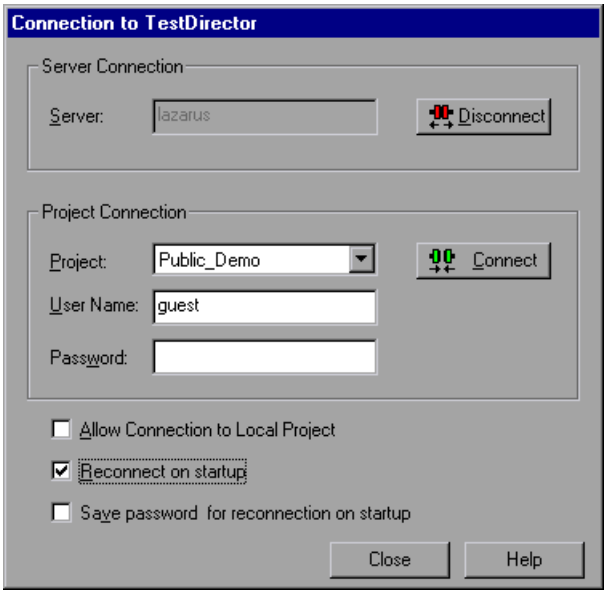
To open a connection to a TestDirector project:

1 Choose **Tools** > **TestDirector Connection** from the Controller.



2 In the Server box, type the name of the TestDirector server and click **Connect**. A message appears indicating that LoadRunner is connecting to the server.

3 In the Project Connection box, select one of the available projects.



4 Provide the user name and password.

5 Select the **Reconnect on Startup** check box to automatically open the connection to the TestDirector server and project when you start LoadRunner.

6 If you select **Reconnect on Startup**, you can save the specified password to automate the login process. Select the **Save Password for Reconnection on Startup** check box.

7 Click **Connect**. A message appears indicating that LoadRunner is connecting to the project.

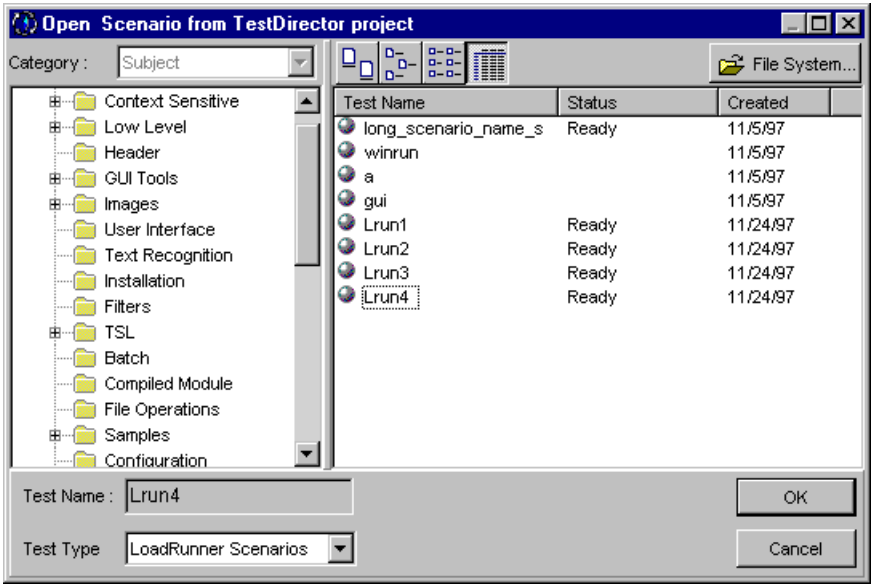
8 Click **Close** to close the Connection to TestDirector dialog box.

Opening a Scenario Using TestDirector

You can open a scenario through the standard file system or the TestDirector repository. Before opening a scenario through TestDirector, make sure you have opened a connection to the project.

To open a scenario from the TestDirector repository:

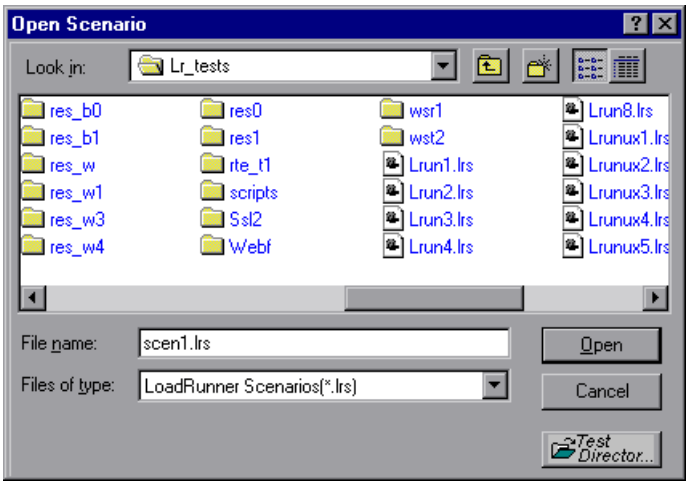
1 Choose **File** > **Open** or click the **File Open** button. The Open Scenario from TestDirector Project dialog box opens.



2 Select a LoadRunner scenario from the project.

3 Click **OK**. LoadRunner loads the scenario. The name of the scenario appears in the Controller's banner. The Design tab shows the scripts, the load generators, and the Vusers and Vuser groups in the scenario.

4 To select a scenario file from the standard file system, click the **File System** button in the upper right corner of the dialog box. The Open Scenario dialog box opens.



5 To return to the Open from TestDirector Project dialog box, click the **TestDirector** button in the bottom right of the dialog box.

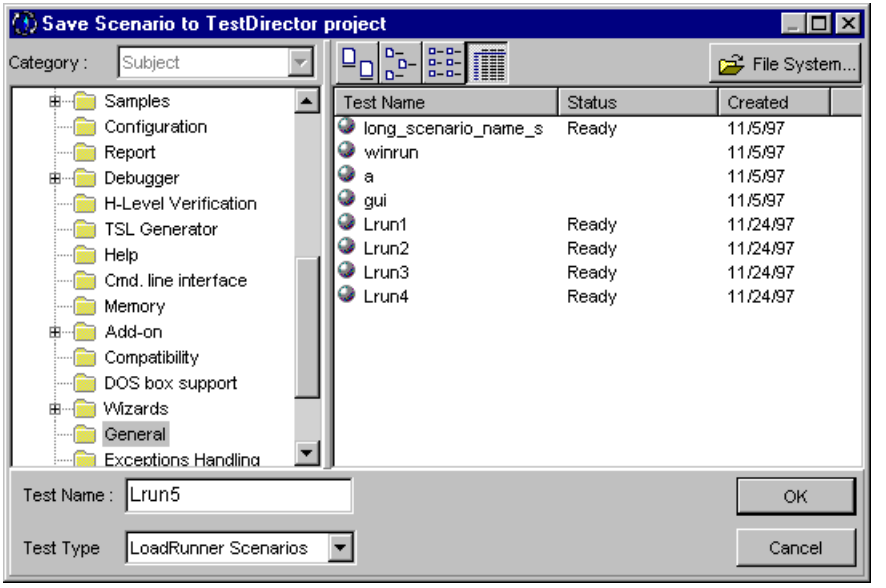
Saving Scenarios to the TestDirector Project

You can save a new scenario, or one that previously existed as a local file, into the TestDirector project.

To save a scenario to the TestDirector project:

1 Connect to the TestDirector server (see "Opening a Connection to a TestDirector Project," on page 124.)

2 Choose **File** > **Save As**. The Save Scenario to TestDirector Project dialog box opens.



3 Type a scenario name, select a folder, and click **OK**.

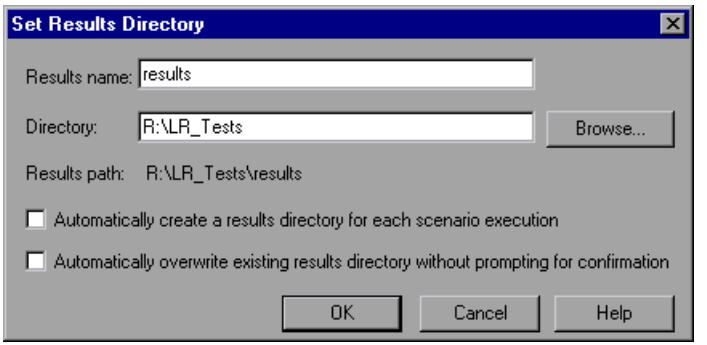
4 To save the scenario to disk, click **File System** and save the file as you would on a standard file system.

Saving Results to the TestDirector Project

Before you run a scenario, you set the results location. If you have a connection open to the TestDirector project, results are saved to the default test set. You can also save the results to a new or existing test set, or to disk using the standard file system.

To specify a results location:

1 Choose **Results** > **Results Setting**. The Set Results Directory dialog box opens.



If you have a connection to TestDirector, a TestDirector button also appears.

2 Type a name for the results in the Results name box.

3 Specify a directory for the results, or click **TestDirector** to save the results in a TestDirector repository.

4 For a TestDirector project, specify the name of a test set.

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

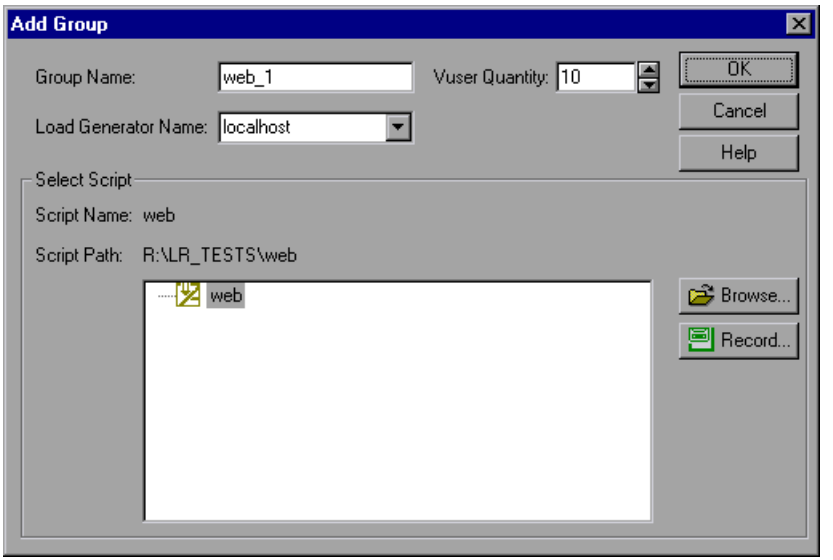
Adding Vuser Scripts with TestDirector

You can add scripts from the TestDirector project to the Controller's script list. If you are creating a manual scenario, you use the Add Group dialog box to add scripts. If you are creating a goal-oriented scenario, you use the Add Script dialog box to add scripts.

To add a script to the script list using the Add Group dialog box:

1 Connect to the TestDirector server (see "Opening a Connection to a TestDirector Project," on page 124).

2 Click the **Add Group** button to the right of the Scenario Groups window. The Add Group dialog box opens:



3 Click the **Browse** button to the right of the Path box. The Open Test dialog box opens.

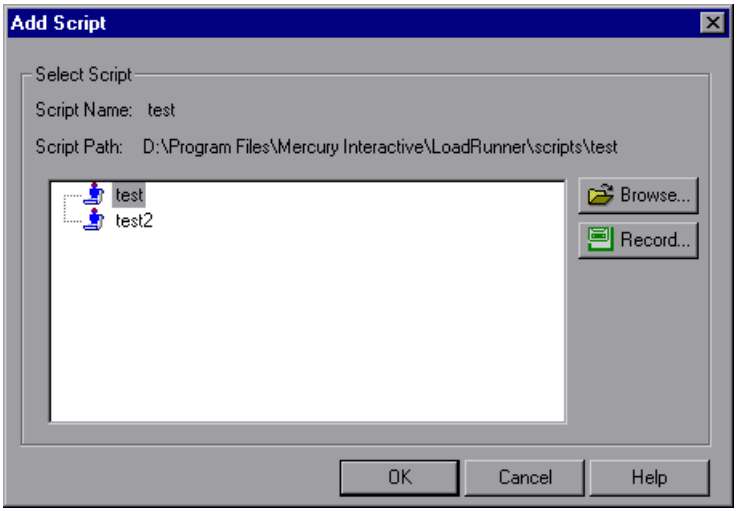
4 Locate the script in the TestDirector repository and click **OK**. The full path and name of the script appear in the Path box. TestDirector scripts are preceded by a "[TD]".

5 Click **OK** to close the Add Group dialog box. The script appears in the Scenario Groups window.

To add a script to the script list using the Add Script dialog box:

1 Connect to the TestDirector server (see "Opening a Connection to a TestDirector Project," on page 124).

2 Click the **Add Script** button to the right of the Scenario Scripts window. The Add Script dialog box opens.



3 Click the **Browse** button to the right of the Path box. The Open Test dialog box opens.

4 Locate the script in the TestDirector repository and click **OK**. The full path and name of the script appear in the Path box. TestDirector scripts are preceded by a "[TD]".

5 Click **OK** to close the Add Script dialog box. The script appears in the Script Path column in the Scenario Scripts window.

Part III

Executing a Scenario

12

Running a Scenario

When you run a scenario, LoadRunner generates load on the application you are testing, and measures the system's performance.

This chapter describes:

- Running an Entire Scenario
- Controlling Vuser Groups
- Controlling Individual Vusers
- Manually Releasing Vusers from a Rendezvous

About Running a Scenario

When you run a scenario, the Vuser groups are assigned to their load generators and execute their Vuser scripts. During scenario execution, LoadRunner:

- records the durations of the transactions you defined in the Vuser scripts
- performs the rendezvous included in the Vuser scripts
- collects error, warning, and notification messages generated by the Vusers

You can run an entire scenario unattended, or you can interactively select the Vuser groups and Vusers that you want to run. When the scenario starts running, the Controller first checks the scenario configuration information. Next, it invokes the applications that you selected to run with the scenario. Then, it distributes each Vuser script to its designated load generator. When

the Vuser groups are ready, they start executing their scripts. While the scenario runs, you can stop both Vuser groups and individual Vusers.

Note: When a test runs in VuGen, the full browser is used. This differs from a test run in the Controller, where only the browser basics are used. There may be occasions when a test passes its run in VuGen, but fails when it is run in the Controller. Before running a scenario in the Controller with multiple Vusers, run a single Vuser to ensure the test is bug free.

The scenario ends when all the Vusers have completed their scripts, when the duration runs out, or when you terminate it.

The following procedure outlines how to run a scenario:

1. Open an existing scenario or create a new one.
2. Configure and schedule the scenario.
3. Set the results directory.
4. Run and monitor the scenario.

While the scenario runs, you can stop both Vuser groups and individual Vusers, monitor each Vuser in the scenario, and view error, warning, and notification messages generated by the Vusers.

Running an Entire Scenario

You can run all the Vusers and Vuser groups in a scenario, or you can select the specific Vuser groups and Vusers that you want to run. Note that when you run an entire scenario, LoadRunner does not begin running Vusers until all of them have reached the READY state. However, if you run individual groups or Vusers, LoadRunner runs the Vusers as soon as they reach the READY state.

The following section describes how to run an entire scenario. "Controlling Vuser Groups," on page 137 and "Controlling Individual Vusers," on page 139 describe how to manipulate Vuser groups and individual Vusers.

To run an entire scenario:

1 Open an existing scenario or create a new one. Click the **Run** tab. The Scenario Groups window appears in the top left-hand corner of the screen.

2 Choose **Scenario** > **Start**, or click the **Start Scenario** button. The Controller starts initializing the Vusers and distributing them to their designated load generators—where they begin to execute their Vuser scripts.

If you have not specified a results directory for the scenario, the Set Results Directory dialog box opens.

During scenario execution you can manipulate individual Vusers and Vuser groups. This is described in "Controlling Vuser Groups," on page 137 and "Controlling Individual Vusers," on page 139.

3 Select **Scenario** > **Stop Ramp Up** to stop the ramp up process. Select **Scenario** > **Return to Ramp Up** to resume the ramping up of Vusers.

4 Choose **Scenario** > **Stop**, or click the **Stop** button to terminate the scenario.

5 Choose **Scenario** > **Reset**, or click the **Reset** button to reset all Vusers to their pre-scenario, DOWN status.

Controlling Vuser Groups

You can run an entire scenario as described above, or you can manipulate individual Vuser groups in the scenario. This section describes how to initialize, run, and stop Vuser groups.

Initializing Vuser Groups

Initializing a Vuser group distributes the Vusers in the group to their designated load generators so that they are ready to execute their script(s). By initializing all of the Vusers in a group before running them, you can ensure that they all begin executing the scenario at the same time.

To initialize a Vuser group:

1 Select the Vuser group or groups that you want to initialize.

2 Click the **Initialize Vusers** button , or right-click the Vuser group or groups that you want to initialize and select **Initialize Group/s**. The Vuser group's

status changes from DOWN to PENDING to INITIALIZING to READY. If a Vuser group fails to initialize, the Vuser group status changes to ERROR.

Running Vuser Groups

Running a Vuser group tells the Vuser group to execute its script.

To run a Vuser group:

1 Select the Vuser group or groups that you want to run.

2 Click the **Run Vusers** button , or right-click the Vuser group or groups that you want to run and select **Run Group/s**. The Vuser groups execute their scripts. If you run a Vuser group in the DOWN or ERROR state, LoadRunner initializes and then runs the Vuser group.

Pausing Vuser Groups

Pausing a Vuser group temporarily stops script execution. The Pause command changes the status of a Vuser group from RUNNING to PAUSED.

To pause a Vuser:

1 Select the Vuser group or groups that you want to pause.

2 Select **Pause** from the right-click menu. The Vusers groups temporarily stop script execution.

Stopping Vuser Groups

Stopping a Vuser group stops script execution. If you stop a Vuser group, the group still appears in the Vuser group list.

To stop a Vuser group:

1 Select the Vuser group or groups that you want to stop.

2 Click the **Stop Vusers** button , or right-click the Vuser group or groups that you want to stop and select **Stop**. The Vuser groups stop executing their scripts.

Resetting Vuser Groups

Resetting causes all of the Vusers in a group to revert to their pre-scenario, DOWN status.

To reset a Vuser group:

1 Select the Vuser group or groups that you want to stop.

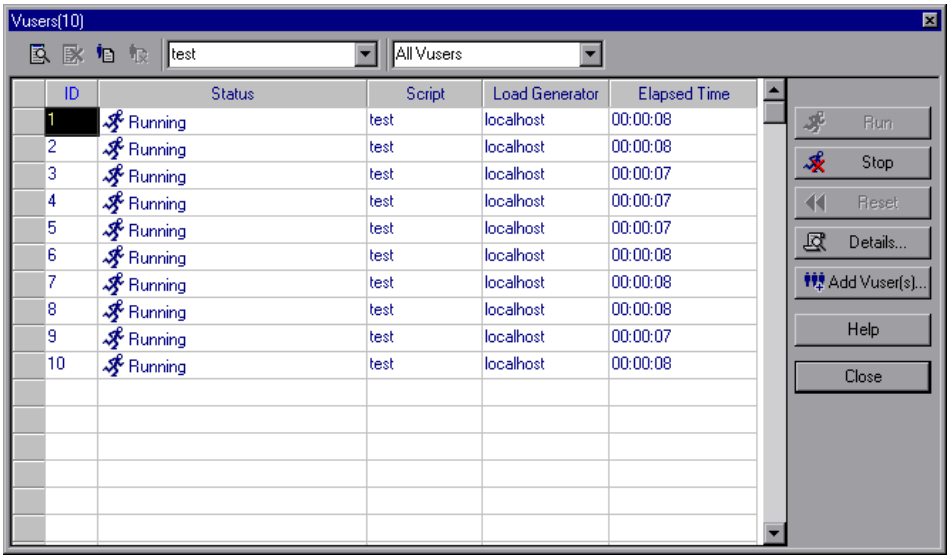
2 Right-click the Vuser group or groups that you want to stop, and select **Reset**. The Vuser groups revert to their pre-scenario, DOWN status.

Controlling Individual Vusers

You can also manipulate individual Vusers within the Vuser groups you have defined. This section describes how to initialize, run, and stop individual Vusers.

To control an individual Vuser:

1 Select a Vuser group and click the **Vusers** button. The Vusers dialog box opens, with a list of the ID, Status, Script, Load Generator, and Elapsed Time (since the beginning of the scenario) for each of the Vusers in the group.

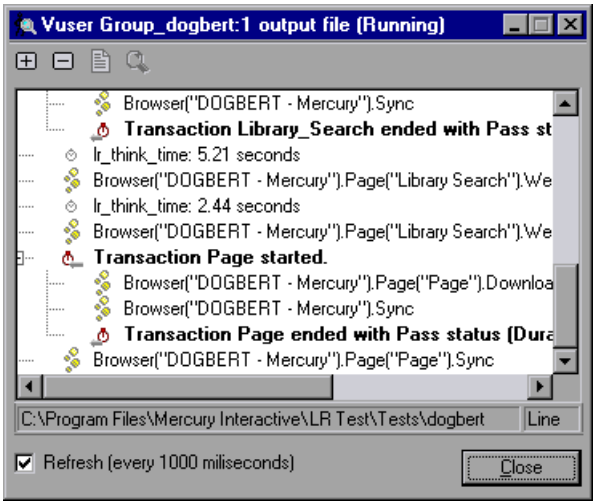


You can control an individual Vuser using the following utilities:

- Select a Vuser and click **Run** to run it.
- Select a Vuser and click **Stop** to stop it from running.
- To pause a Vuser, right-click it and select **Pause**.
- Select a Vuser and click **Reset** to revert its status to DOWN.
- To initialize a Vuser, right-click it and select **Initialize Vuser/s**.
- To renumber the Vusers in a group, right-click the Vusers you want to renumber and select **Renumber**.
- To filter the Vusers listed, right-click in one of the columns and select **Filter Vusers.** Select the way in which you want to filter the Vusers. Alternatively, you can select the filter option you want to use from the filter selector at the top of the Vusers dialog box.
- To sort the Vusers listed, right-click in one of the columns and select **Sort Vusers**. Select the way in which you want to sort the Vusers.
- To view a Vuser executing its assigned script, select the Vuser and click the **Show** button. A browser opens, allowing you to see the Vuser executing the script.

To close the browser, click the **Hide** button.

➤ To view the Vuser script log, click the **Vuser log** button. A script log, such as the following, appears.



To close the Vuser script log, click the **Close Vuser log** button.

2 Click **Close** to close the Vusers dialog box.

Manually Releasing Vusers from a Rendezvous

While you run a scenario, you can manually release Vusers from a rendezvous before the Controller releases them.

To manually release Vusers from a rendezvous:

1 Choose **Scenario** > **Rendezvous**. The Rendezvous Information dialog box opens.

2 Select a rendezvous from the Rendezvous list.

3 Click **Release**. The Vusers in the rendezvous are released.

13

Viewing Vusers During Execution

During scenario execution, you can view the actions that are performed by Vusers.

This chapter describes:

- Monitoring Vuser Status
- Viewing the Output Window
- Logging Execution Notes
- Viewing the Agent Summary

About Viewing Vusers During Execution

LoadRunner lets you view Vuser activity during a scenario:

- On the Controller load generator machines, you can view the Output window, monitor Vuser performance online, and check the status of Vusers executing the scenario.
- On remote machines, you can view the Agent summary with information about the active Vusers.

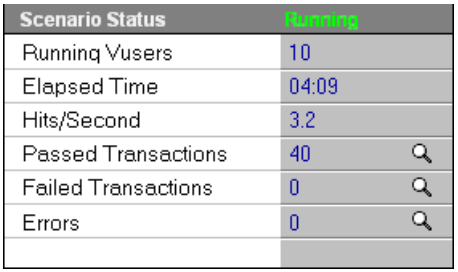
Monitoring Vuser Status

During scenario execution, you can use the Scenario Groups window in the Run view to monitor the actions of all the Vusers and Vuser groups in the scenario.

The Status field of each Vuser group displays the current state of each Vuser in the group. The following table describes the possible Vuser states during a scenario.

Status	Description
DOWN	The Vuser is down.
PENDING	The Vuser is ready to be initialized and is waiting for an available load generator, or is transferring files to the load generator. The Vuser will run when the conditions set in its scheduling attributes are met.
INITIALIZING	The Vuser is being initialized on the remote machine.
READY	The Vuser already performed the init section of the script and is ready to run.
RUNNING	The Vuser is running. The Vuser script is being executed on a load generator.
RENDEZVOUS	The Vuser has arrived at the rendezvous and is waiting to be released by LoadRunner.
DONE.PASSED	The Vuser has finished running. The script passed.
DONE.FAILED	The Vuser has finished running. The script failed.
ERROR	A problem occurred with the Vuser. Check the Status field on the Vuser dialog box or the output window for a complete explanation of the error.
EXITING	The Vuser has finished running or has been stopped, and is now exiting.
STOPPED	The Vuser stopped when the Stop command was invoked.

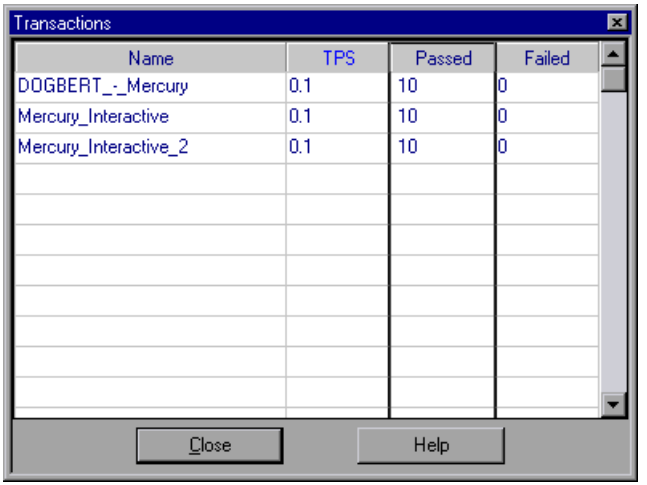
You can also view a synopsis of the running scenario in the box at the top right-hand corner of the Run view.



Status Summary	Description
SCENARIO STATUS	indicates whether the scenario is RUNNING or DOWN
RUNNING VUSERS	indicates how many Vusers are being executed on a load generator machine
ELAPSED TIME	indicates how much time has elapsed since the beginning of the scenario
HITS/SECOND	indicates how many hits (HTTP requests) there have been to the Web site being tested per second that each Vuser has been running
PASSED TRANSACTIONS	indicates how many transactions have been executed successfully
FAILED TRANSACTIONS	indicates how many transactions have been executed unsuccessfully
ERRORS	indicates how many problems have occurred with the Vusers

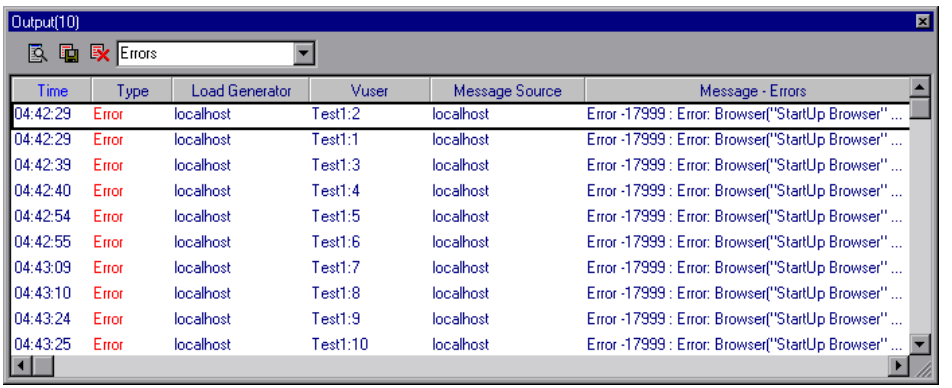
To view details of the transactions and errors:

1 Click the **Show Snapshot** button to the right of the Passed Transactions or Failed Transactions in the Scenario Status window. The Transactions dialog box opens.



The **Name** column lists the individual transactions in a script. For each transaction, the Transactions dialog box lists information concerning the number of **Transactions Per Second (TPS)**, the number of transactions **Passed**, and the number of transactions **Failed**.

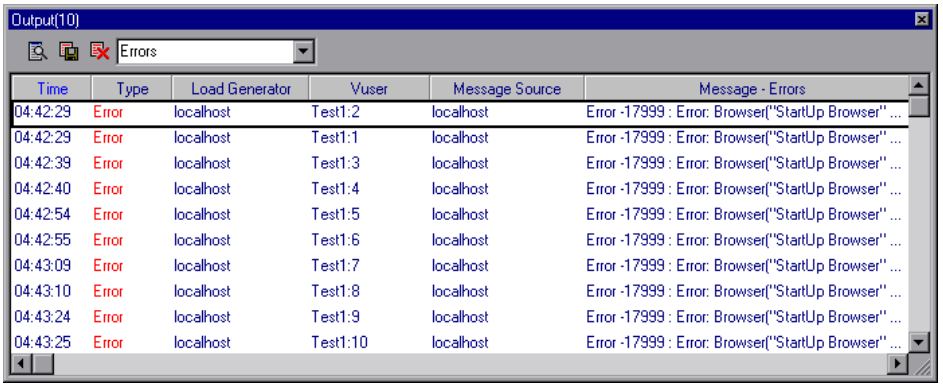
2 Choose **View** > **Show Output** or click the **Show Snapshot** button to the right of the Errors listing. The Output window opens.



The **Vusers** column lists the Vusers that did not execute properly. The Output window also lists the time at which the errors occurred, the load generator machine on which the errors took place, and the reason for each error. For more information on the Output window, see "Viewing the Output Window" on page 147.

Viewing the Output Window

While the scenario runs, the Vusers and load generators send error, notification, and warning messages to the Controller. You can view these messages in the Output window. LoadRunner clears the messages in this window at the start of each scenario execution.



The Output window provides the following scenario information:

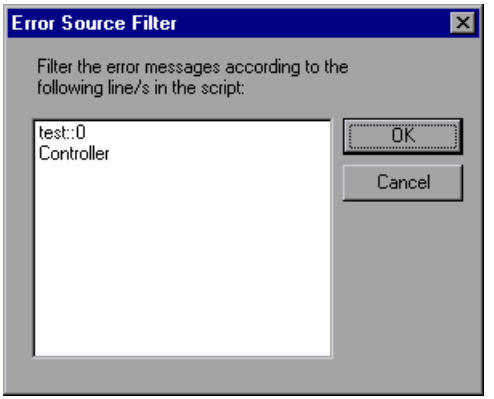
Column	Description
TIME	the time each message arrived
TYPE	the type of message sent: Error, Notify, Warning
LOAD GENERATOR	the Vuser's load generator
VUSER	the ID of the Vuser that sent the message
MESSAGE SOURCE	the name of the load generator that sent the message
MESSAGE - ALL MESSAGES	the text of the message

Note: You can limit the number of messages in the output window, and set a deletion quota for the number of messages that will be overwritten. For more information, see Appendix C, "Working in Expert Mode."

You can view and manipulate the messages using the following utilities:

- To show (or hide) the Output window, choose **View** > **Show Output**.
- To sort output messages, click the appropriate column header, or right-click within the Output window and select **Sort Messages**. Choose the method by which you want to sort the messages.
- To filter output messages to display only certain message types, choose the type of message you want to view in the selector, or right-click within the Output window and select **Filter Messages**. Choose the way in which you want to filter the messages.

  If you choose to filter by **Error Source**, the Error Source Filter dialog box opens.

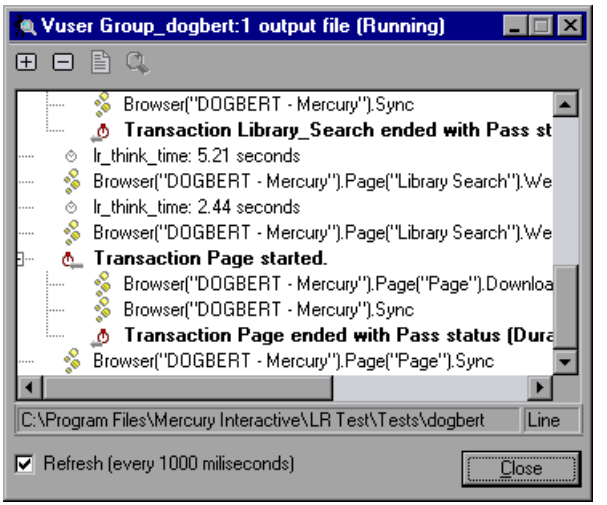


  Select a line in the script according to which you want LoadRunner to filter the error messages in the Output window.

- To view all of a message's details, right-click the message and select **Details**.
- To save messages, right-click within the Output window and select **Export To File**, or click the **Export to File** button.

➤ To clear scenario execution messages, right-click within the Output window and select **Remove All**, or click the **Remove All** button.

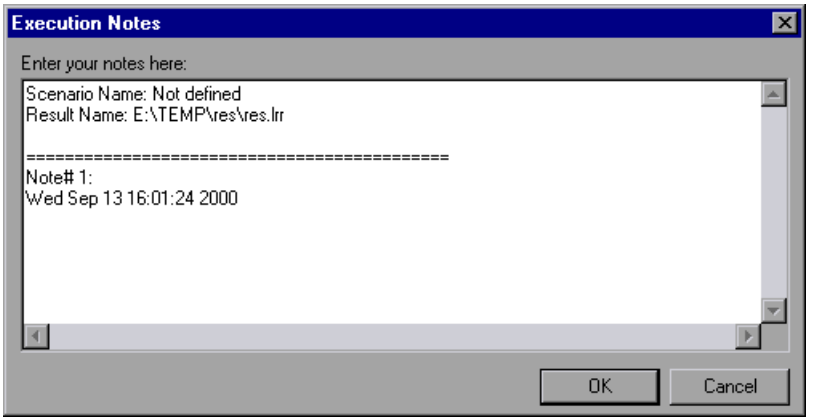
➤ To view the complete Vuser log, right-click a message and select **Show Vuser Log**, or click the **Show Vuser Log** button. This log contains run-time information about the Vuser and is refreshed every 1000 milliseconds. To disable the refreshing of this log, clear the **Refresh** check box.



➤ To view a snapshot of the Web page where an error occurred, highlight the error in the Vuser Log and click the **Show Snapshot** button.

Logging Execution Notes

The Controller provides you with a dialog box in which you can log comments while a scenario is running.



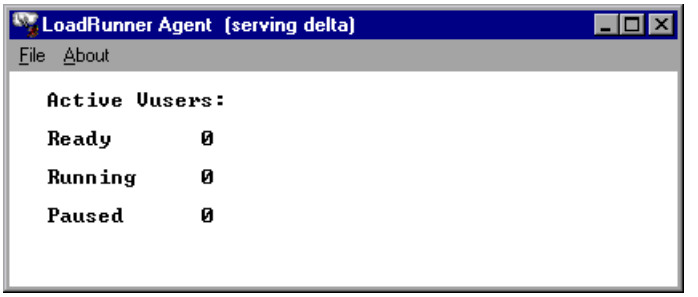
To log execution notes:

1. Select **Scenario** > **Execution Notes**. The Execution Notes dialog box opens.
2. Enter the note or notes that you want to log.
3. Click **OK** to close the dialog box. LoadRunner saves the note(s) you recorded.

Viewing the Agent Summary

When you run a scenario with non-GUI Vusers, the machine running the Vusers invokes an agent that controls the Vuser execution on that load generator. During scenario execution, the agent displays a summary of the Ready, Running, and Paused Vusers.

The Agent window comes forward at the start of the scenario. You can minimize and restore it at any time.



Part IV

Monitoring a Scenario

14

Online Monitoring

You can monitor scenario execution using the LoadRunner online run-time, transaction, system resource, Web resource, Web server resource, Web application server resource, database server resource, network delay, streaming media resource, firewall server resource, and ERP server resource monitors.

The specific monitors are discussed in the next few chapters. This chapter describes the online monitor user interface:

- Starting the Monitors
- Configuring Online Monitors
- Setting Monitor Options
- Configuring Online Graphs
- Opening Online Monitor Graphs
- Merging Graphs
- Understanding Online Monitor Graphs
- Configuring Online Measurements
- Exporting Online Monitor Graphs
- Viewing Data Offline

About Online Monitoring

LoadRunner provides the following online monitors:

- Run-Time
- Transaction
- System Resources
- Web Resources
- Web Server Resources
- Web Application Server Resources
- Database Server Resources
- Network Delay
- Streaming Media
- FireWall
- ERP Server Resources

The *Run-Time* monitor displays the number and status of Vusers participating in the scenario, as well as the number and types of errors that the Vusers generate. It also provides the Data Point graph that displays the real-time values for user-defined points.

The *Transaction* monitor displays the transaction rate and response time during scenario execution. For more information, see Chapter 15, "Run-Time and Transaction Online Monitors."

The *System Resource* monitors gauge the Windows, UNIX, TUXEDO, and SNMP resources used during a scenario. To activate the System Resource monitors, you must set the monitor options before you run your scenario. For information on setting these options, see Chapter 16, "System and Database Resource Monitoring."

The *Web Resource* monitor measures statistics at the Web server(s) during scenario runs. It provides information about the number of Web connections, throughput volume, HTTP responses, and downloaded pages during the scenario. For more information on the Web Resource monitor, see Chapter 17, "Web Resource and Web Server Resource Monitors."

The *Web Server Resource* monitors measure statistics at the Apache, Microsoft IIS, and Netscape Web servers during the scenario. To activate the Web Server Resource monitors, you must set up a list of resources to monitor before you run your scenario. For more information, see Chapter 17, "Web Resource and Web Server Resource Monitors."

The *Web Application Server Resource* monitors measure statistics at the Web application server(s) during the scenario. To activate the Web Application Server Resource monitors, you must set up a list of resources to monitor before you run your scenario. For more information, see Chapter 18, "Web Application Server Resource Monitors."

The *Database Server Resource* monitors measure statistics related to the SQL server and Oracle databases. To activate the Database Server Resource monitors, you must set up a list of measurements to monitor before you run your scenario. For more information, see Chapter 16, "System and Database Resource Monitoring."

The *Network Delay* monitor displays information about the network delays on your system. To activate the Network Delay monitor, you must set up the network paths to monitor before you run your scenario. For more information see Chapter 19, "Network Monitoring."

The *Streaming Media* monitors measure statistics at the Windows Media Server and RealPlayer audio/video servers, as well as the RealPlayer client. To activate the Streaming Media monitors, you must set up a list of resources to monitor before you run your scenario. For more information, see Chapter 20, "Streaming Media Monitoring."

The *FireWall* monitor measures statistics at the firewall servers during the scenario. To activate the FireWall monitor, you must set up a list of resources to monitor before you run your scenario. For more information, see Chapter 21, "FireWall Server Performance Monitoring."

The *ERP Server Resource* monitor measures statistics at the ERP servers during the scenario. To activate the ERP Server Resource monitor, you must set up a list of resources to monitor before you run your scenario. For more information, see Chapter 22, "ERP Server Resource Monitoring."

All of the monitors allow you to view a summary of the collected data at the conclusion of the scenario. Using LoadRunner Analysis, you can generate a graph for any of the monitors. For more information, see the *LoadRunner Analysis User's Guide*.

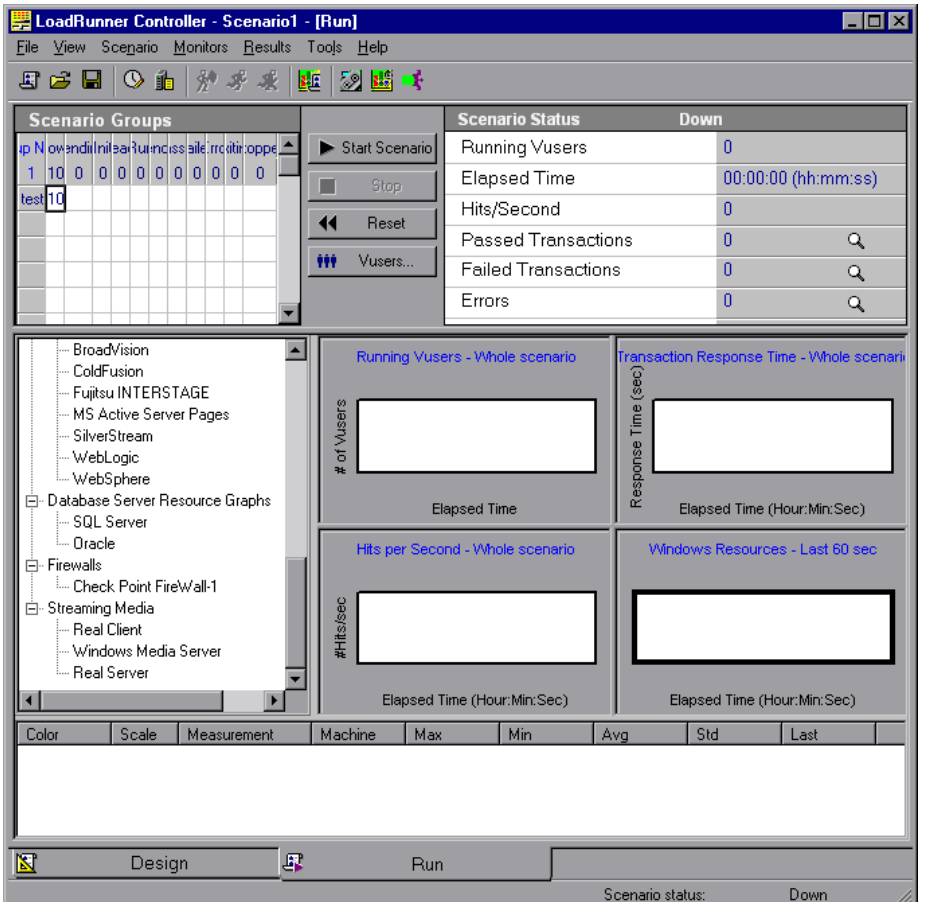
Starting the Monitors

You use the online monitors to monitor Vuser status, errors, transactions, system resources, database server resources, Web resources, Web server resources, Web application server resources, streaming media resources, firewall server resources, and network delay.

To start the online monitors:

1 Start the scenario. Select the Vuser groups you want to run and click the **Start Scenario** button, or choose **Scenario** > **Start**.

2 Click the **Run** tab. The default graphs are displayed below the Scenario Groups window.



3 Double-click a graph to maximize it. Repeat the operation to restore the tiled view.

4 Click the "+" in the left pane to expand the graph tree. To hide the graph tree view, select **View > Hide Available Graphs**.

5 Select a graph from the tree and drag it into the right pane. You can also drag graphs between panes.

Note: The Transaction Monitor graphs will not contain any data unless transactions are being executed. In addition, the System Resource Monitor, Web Monitor, Network Monitor, Streaming Media, and FireWall graphs will not contain any data unless you set up a list of resources to monitor before running your scenario. To set up a list of resources to monitor, see Chapter 16, "System and Database Resource Monitoring," Chapter 17, "Web Resource and Web Server Resource Monitors," Chapter 18, "Web Application Server Resource Monitors," Chapter 19, "Network Monitoring," Chapter 20, "Streaming Media Monitoring," Chapter 22, "ERP Server Resource Monitoring," and Chapter 21, "FireWall Server Performance Monitoring."

Configuring Online Monitors

LoadRunner lets you configure the settings for your online monitors. You can set graph measurements and properties, such as the sampling time, the colors of the lines, and the scale of the graph.

Monitor options: global sampling rate, error handling, debugging, and the frequency settings. For more information, see "Setting Monitor Options" on page 161.

Graph properties: refresh rate, display type, graph time for the x-axis, and the y-axis scale. For more information, see "Configuring Online Graphs" on page 163.

Measurement settings: line color, scale of the y-axis, and whether to show or hide the line. For more information, see "Configuring Online Measurements" on page 170.

When you save a scenario, the online monitor configuration settings are saved as well.

Setting Monitor Options

Before running your scenario, you can set monitor options in the following areas:

- Sampling Rate
- Error Handling
- Debug
- Frequency

Sampling Rate: The sampling rate is the period of time (in seconds) between consecutive samples. By default, the online monitor samples the data at intervals of three seconds. If you increase the sampling rate, the data is monitored less frequently. This setting applies to all graphs. To set a sampling rate for a specific graph, see "Configuring Online Graphs" on page 163.

Error Handling: You indicate how LoadRunner should behave when a monitor error occurs—issue a popup message box (default) or send error messages to the Output window.

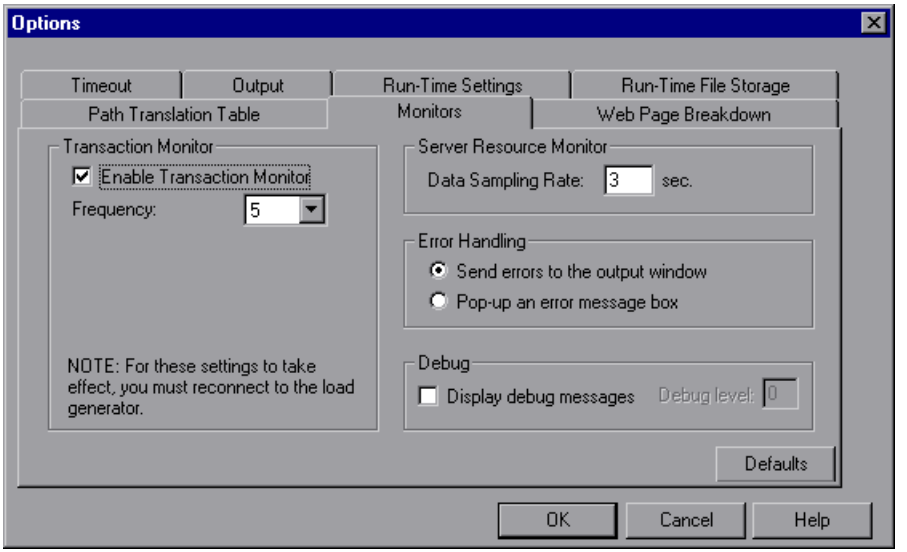
Debug: The online monitor provides debugging capabilities. You can display the debug messages in the output log. For the Network monitor, you can indicate the debug (detail) level of messages sent to the log, ranging from 1-9.

Frequency: You set the frequency at which the monitor sends updates to the Controller for the Transaction, Throughput, Data Point, Hits per Second, HTTP Responses per Second, and Downloaded Pages per Second graphs. The data is averaged for the frequency period defined, and only one value is sent to the Controller.

For information on enabling and disabling the Transaction monitor and Web page breakdown, see Chapter 15, "Run-Time and Transaction Online Monitors."

To set monitor options:

1 Select **Tools** > **Options** and select the **Monitors** tab.



2 Specify the frequency at which the monitor should send updates to the Controller for the Transaction, Throughput, Data Point, Hits per Second, HTTP Responses per Second, and Downloaded Pages per Second graphs. The default value is 5 seconds. For a small scenario, it is recommended that you use a frequency of 1. For a large scenario, it is recommended that you use a frequency of 3-5. The higher the frequency, the less network traffic there will be.

Note: You cannot modify these settings during scenario execution—you must stop the scenario before disabling the monitor or changing its frequency.

An additional setting may be configured while working in Expert mode. For information on working in Expert mode, see Appendix C, "Working in Expert Mode."

3 Enter a sampling rate.

4 Set the desired **Error Handling** option.

5 To display debug messages in the Output window, select the **Display Debug Messages** check box. For the Network monitor, specify a **Debug level** from 1-9.

6 Click **OK** to save your settings and close the Options dialog box.

Configuring Online Graphs

You can customize your graph in the following areas:

- Refresh Rate
- X-axis Style
- Graph Time
- Display Type
- Y-axis Style

Note that these settings can be set globally—to apply to all graphs—or per graph.

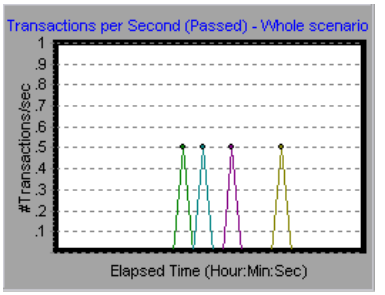
Refresh Rate

The refresh rate is the interval in which the graph is refreshed with new data. By default, the graph is refreshed every five seconds. If you increase the refresh rate, the data is refreshed less frequently.

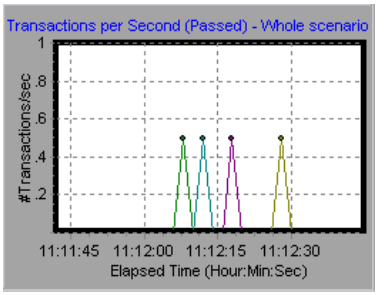
Note: In a large load test, it is recommended to use a refresh rate of three to five seconds. This enables you to avoid problems with CPU resource usage.

X-axis Style

You can specify how the graph displays the x-axis time: *Don't show*, *Clock time*, or *Relative to Scenario Start*. The *Don't Show* setting instructs LoadRunner not to display values for the x-axis. The *Clock Time* setting displays the absolute time, based on the system clock. The *Relative to Scenario Start* setting displays the time relative to the beginning of the scenario. In the following example, the graph is shown with the *Don't show* and *Clock time* options:



Don't Show



Clock time

Graph Time

The Graph Time settings indicate the scale for a graph's x-axis when it is time-based. A graph can show 60 or 3600 seconds of activity. To see the graph in greater detail, decrease the graph time. To view the performance over a longer period of time, increase the graph time. The available graph times are: *Whole scenario*, *60*, *180*, *600*, and *3600* seconds.

Display Type

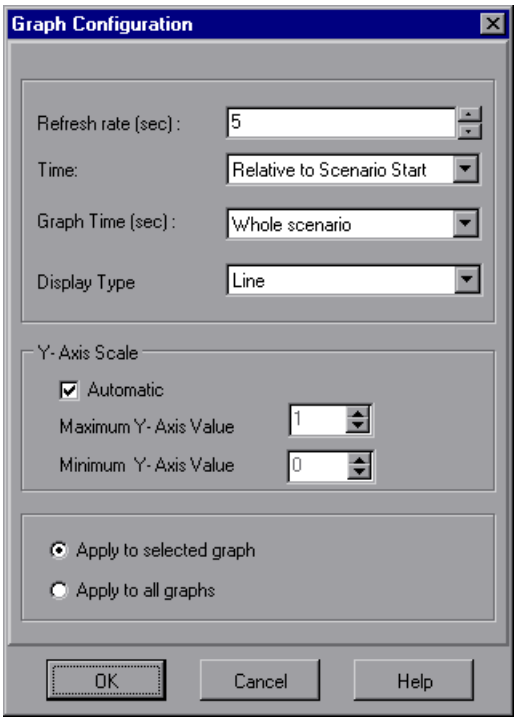
You can specify whether LoadRunner displays the Network Delay Time graph as a line, pie, or area graph. By default, the graph is displayed as a line graph. Note that all other graphs can only be displayed as line graphs.

Y-axis Style

You can instruct LoadRunner to display graphs using the default y-axis scale, or you can specify a different y-axis scale. Click **Automatic** if you want LoadRunner to use the default y-axis values. Specify a maximum or minimum value for the y-axis if you want to modify the y-axis scale.

To customize your graphs:

1 Select the online graph you want to configure (in either the right or left pane) and choose **Monitors** > **Online Graphs** > **Configure**. Alternatively, right-click a graph and select **Configure**. The Graph Configuration dialog box opens.



2 To apply the dialog box settings to all graphs, select **Apply to all graphs**.

3 Enter the desired refresh rate—the time between graph updates—in the Refresh Rate box.

4 Select a style for the x-axis from the Time box.

5 Select a value from the Graph Time box. The graph time is the time in seconds displayed by the x-axis.

6 For the Network Delay Time graph, select a graph style—Line, Pie, or Area—from the Display Type box.

7 Select a maximum or minimum value for the y-axis, or choose **Automatic** to view graphs using the default y-axis scale.

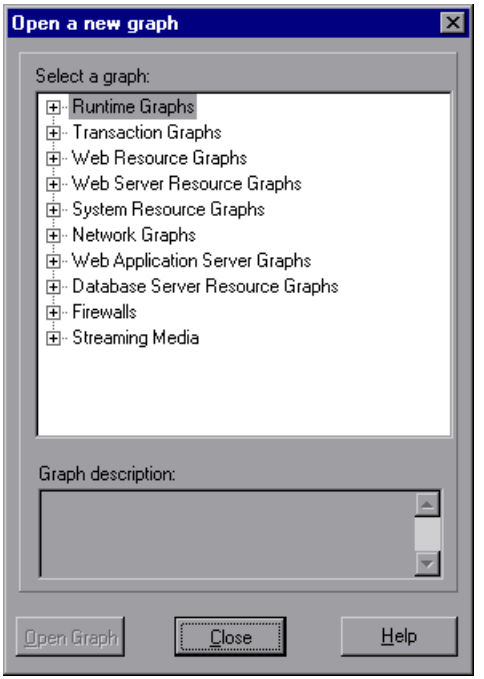
8 Click **OK** to save your settings and close the Graph Configuration dialog box.

Opening Online Monitor Graphs

By default, LoadRunner displays four graphs in the Run view: Running Vusers, Transaction Response Time, Hits per Second, and Windows Resources. You can display the other graphs by clicking and dragging them from the graph tree view to the graph view area. Alternatively, you can open a new graph using the Open a New Graph dialog box.

To open a new graph using the Open a New Graph dialog box:

1 Select **Monitors** > **Online Graphs** > **Add New Graph**, or right-click a graph and select **Open New Graph**. The Open a New Graph dialog box opens.



2 Click the "+" in the left pane to expand the graph tree, and select a graph. You can view a description of the graph in the **Graph Description** box.

3 Click **Open Graph**. The graph appears in the graph view area.

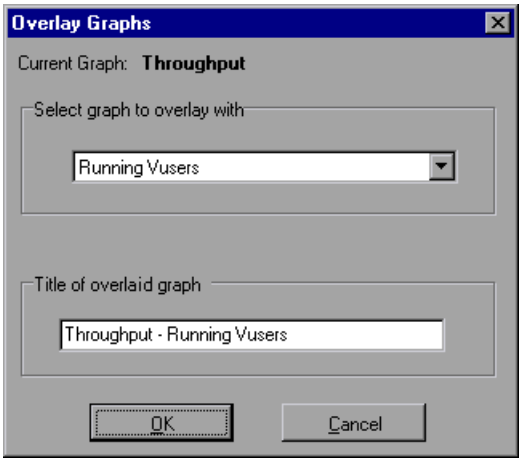
Merging Graphs

LoadRunner lets you merge the results of two graphs from the same scenario into a single graph. The merging allows you to compare several different measurements at once. For example, you can make a merged graph to display the Web Throughput and Hits per Second, as a function of the elapsed time. Note that in order to merge graphs, their x-axis must be the same measurement.

When you overlay the contents of two graphs that share a common x-axis, the left y-axis on the merged graph shows the current graph's values. The right y-axis shows the values of the graph that was merged.

To overlay two graphs:

1 Right-click one of the graphs you want to overlay, and select **Overlay Graphs**. The Overlay Graphs dialog box opens.



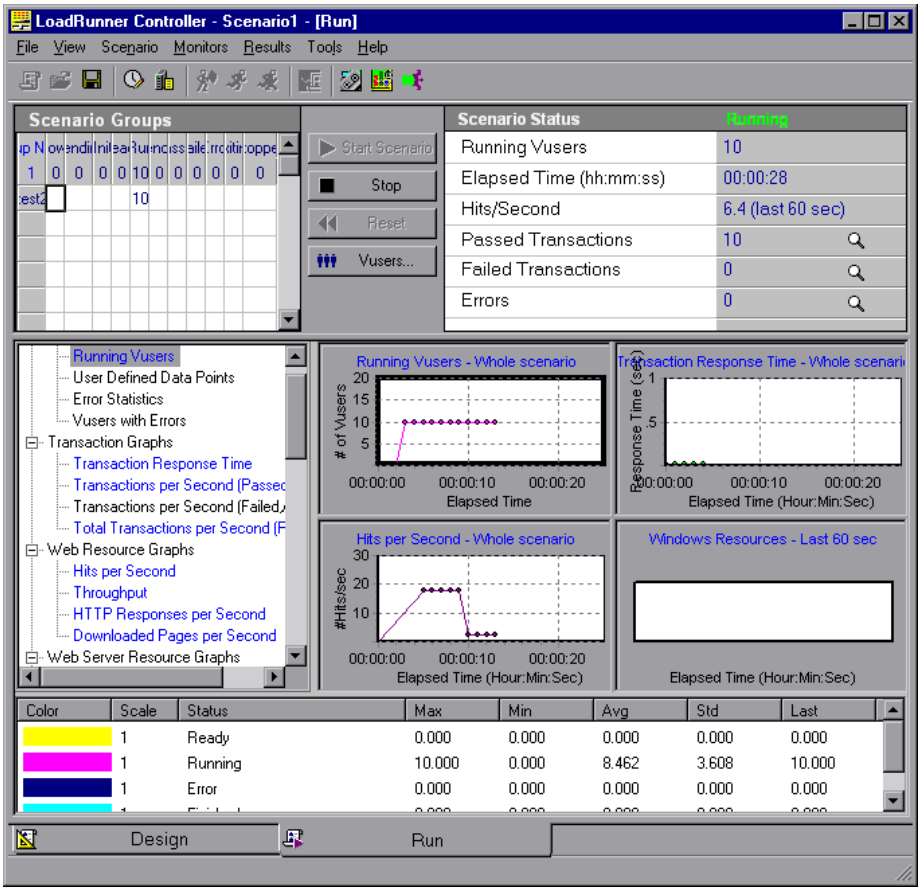
2 Select a graph with which you want to overlay the current graph. The drop-down list only shows the active graphs that have a common x-axis with the current graph.

3 Enter a title for the overlaid graph.

4 Click **OK**. The merged graph appears in the graph view area.

Understanding Online Monitor Graphs

Online monitor graphs display information about the measurements listed below the graph. Each value is represented by a colored line. A legend beneath the graph indicates the color and measurement.



By default, the online monitor displays each measurement in a scenario in the legend below the graphs. The legend displays the measurements for the selected graph.

To get additional information about a measurement, right-click the measurement and choose **Description**.

To focus on a particular line, you can:

➤ **Highlight a measurement:** To highlight a specific measurement, select it in the legend. The corresponding line in the graph is displayed in blue.

➤ **Hide a measurement:** To hide a measurement, right-click the measurement and choose **Hide**.

To show a hidden measurement, right-click the measurement and choose **Show**.

➤ **Pause the monitor:** To pause a specific graph during scenario execution, select the graph and choose **Monitors** > **Online Graph** > **Freeze,** or right-click the graph and select **Freeze**. To resume, repeat one of the above actions. When you resume, the graph displays the data for the paused period.

Configuring Online Measurements

You can configure the following online measurement settings:

➤ Changing Line Colors

➤ Setting the Scale of the Measurement

➤ Showing and Hiding Transactions

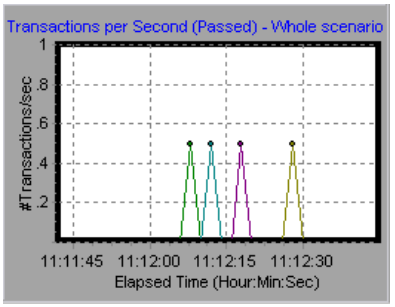
Changing Line Colors

LoadRunner assigns a unique color to each measurement. You can modify the color using the configuration interface.

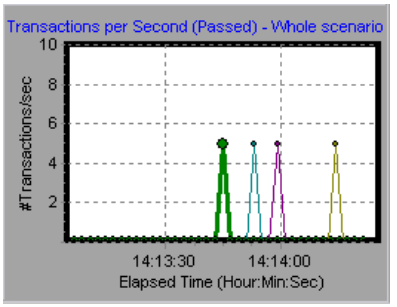
Setting the Scale of the Measurement

You can modify the scale of a measurement—the relationship between the y-axis and the graph's actual value. For example, a scale set at 1 indicates that the measurement's value is the value of the y-axis. If you choose a scale of 10, you must divide the y-axis value by 10 to obtain the true value of the measurement.

In the following example, the same graph is displayed with a scale of 1 and 10.



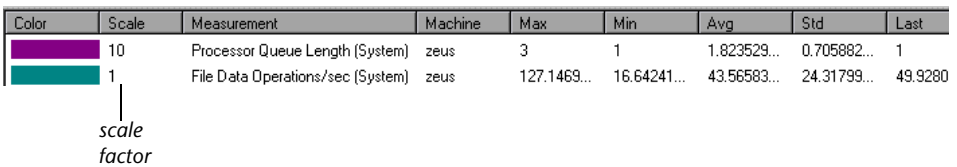
scale = 1



scale = 10

The actual graph values range from 0-1, as shown in the left graph. You can view the information more accurately using a larger scale for the display, as shown in the right graph. However, to obtain the actual values, you need to divide the displayed value by the scale. In the example above, the highest value shown in the graph is 5. Since the scale is 10, the actual value is 0.5.

The legend below the graph indicates the scale factor.

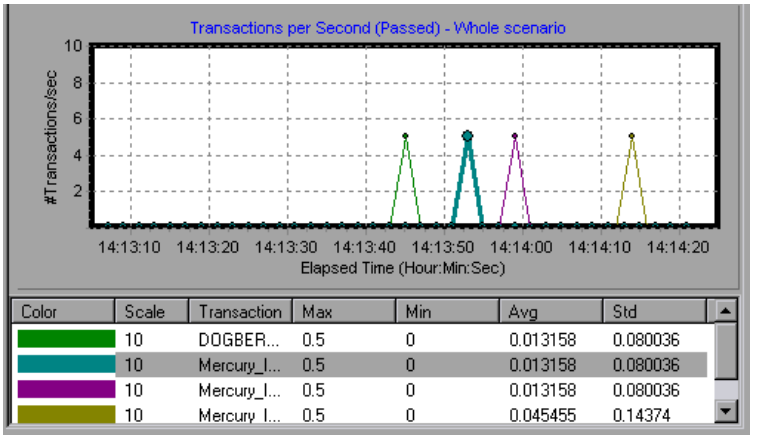


By default, LoadRunner uses the *autoscale* option, which automatically scales the measurements by calculating the best ratio for displaying the graph.

Showing and Hiding Transactions

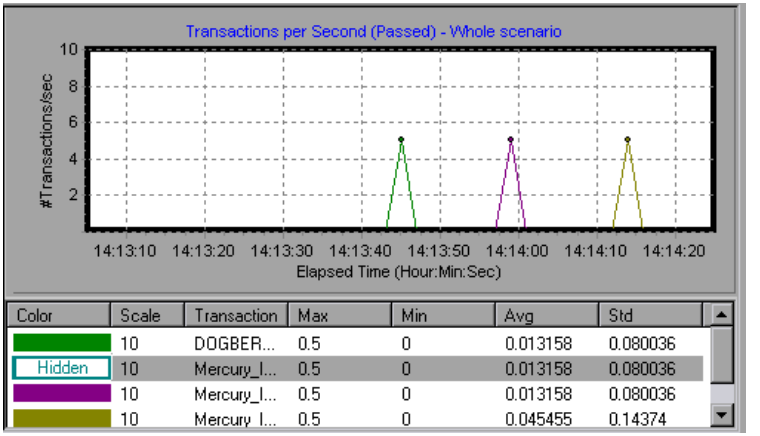
By default, the Transaction Monitor displays a line for each item in the transaction list. You can hide the line for any of the monitored transactions in order to focus on a specific measurement.

In the following example, a line is shown for each measurement.



Color	Scale	Transaction	Max	Min	Avg	Std
	10	DOGBER...	0.5	0	0.013158	0.080036
	10	Mercury_l...	0.5	0	0.013158	0.080036
	10	Mercury_l...	0.5	0	0.013158	0.080036
	10	Mercury_l...	0.5	0	0.045455	0.14374

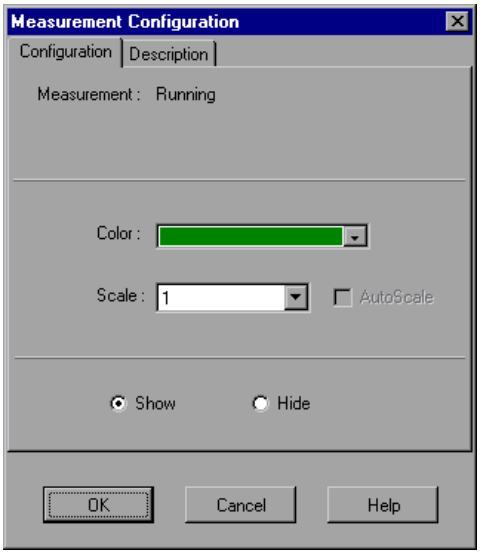
In this example, the second item in the legend is hidden.



Color	Scale	Transaction	Max	Min	Avg	Std
	10	DOGBER...	0.5	0	0.013158	0.080036
Hidden	10	Mercury_l...	0.5	0	0.013158	0.080036
	10	Mercury_l...	0.5	0	0.013158	0.080036
	10	Mercury_l...	0.5	0	0.045455	0.14374

To configure a measurement:

1 In the legend below the graphs, select the measurement you want to configure. Right-click and choose **Configure**. The Measurement Configuration dialog box opens.



2 To change the color of the line, select a color from the Color list.

3 To change the scale, clear the **Autoscale** check box and select the desired ratio from the Scale list.

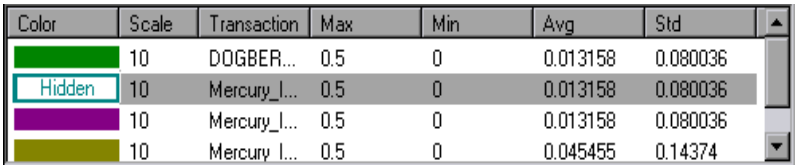
4 To hide a measurement, click **Hide**. To show a hidden resource, click **Show**.

Note that you can also show and hide measurements without opening the Measurement Configuration dialog box, by right-clicking a measurement in the legend and selecting **Show**/**Hide**.

5 Click **OK** to accept the settings and close the dialog box.

The specified changes are reflected in the graph and in the legend beneath the graph. The color is displayed in the first column of the legend. Hidden

transactions are displayed as unfilled boxes. The scale is displayed in the legend's second column.



Color	Scale	Transaction	Max	Min	Avg	Std
	10	DOGBER...	0.5	0	0.013158	0.080036
Hidden	10	Mercury_l...	0.5	0	0.013158	0.080036
	10	Mercury_l...	0.5	0	0.013158	0.080036
	10	Mercury_l...	0.5	0	0.045455	0.14374

Exporting Online Monitor Graphs

LoadRunner allows you to export the online graph to HTML for viewing at a later stage. When you export to HTML, the legend is also displayed with the graph. You can export all graphs or only the selected one.

To export online graphs to HTML:

1. To export a specific graph, select the graph you want to export and choose **Monitors** > **Online Graphs** > **Export to HTML**. The Select Filename and Path dialog box opens.
2. To export all graphs in the Online Monitor view, choose **Monitors** > **Export Online Graphs to HTML.** The Select Filename and Path dialog box opens.
3. Specify a filename and path and click **Save**.

Viewing Data Offline

After monitoring resources during a scenario run, you can view a graph of the data that was gathered using the LoadRunner Analysis. When you run the Analysis utility, it processes the data and generates a graph for each measurement that was monitored.

To view a graph, choose **Graph** > **Add Graph** in the Analysis window. For more information about working with the LoadRunner Analysis at the conclusion of the scenario, see the *LoadRunner Analysis User's Guide*.

15

Run-Time and Transaction Online Monitors

While running a scenario, you can use LoadRunner's online monitors to view graphs of run-time status and transaction performance.

This chapter describes:

- Run-Time Graphs
- User-Defined Data Point Graph
- Transaction Monitor Graphs
- Enabling the Transaction Monitor
- Adding Transactions to a Script
- Enabling Web Page Breakdown

About Online Graphs

LoadRunner's online monitor lets you view Vuser performance during scenario execution. You can view Vuser status, errors, transaction rate and response time, system and database resource information, network delay information, streaming media resource information, firewall server resource information, ERP server resource information, and Web resource information. This chapter describes the Run-Time and Transaction monitors. For information about the other online monitors, see the following chapterss.

The *Run-Time* monitor provides information about the status of the Vusers participating in the scenario, as well as the number and types of errors that

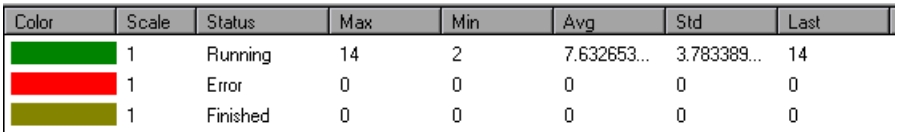
the Vusers generate. It also provides the Data Point graph that displays the real time values for user-defined points.

The *Transaction* monitor displays the transaction rate and response time during scenario execution. For more information about transactions, see "Adding Transactions to a Script" on page 179.

Once the scenario finishes execution, you can use the LoadRunner Analysis to view summary graphs of all the data collected during execution. For more information, see the *LoadRunner Analysis User's Guide*.

Run-Time Graphs

The monitor's **Running Vusers** graph provides information about the status of the Vusers running in the current scenario on all load generator machines. The graph shows the number of running Vusers, while the information in the legend indicates the number of Vusers in each state.



Color	Scale	Status	Max	Min	Avg	Std	Last
	1	Running	14	2	7.632653...	3.783389...	14
	1	Error	0	0	0	0	0
	1	Finished	0	0	0	0	0

The Status field of each Vuser displays the current status of the Vuser. The following table describes each Vuser status.

Status	Description
RUNNING	The total number of Vusers currently running on all load generators.
READY	The number of Vusers that completed the initialization section of the script and are ready to run.
FINISHED	The number of Vusers that have finished running. This includes both Vusers that passed and failed.
ERROR	The number of Vusers whose execution generated an error. Check the Status field in the Vuser view or the Output window for a complete explanation of the error.

The monitor's **Error Statistics** graph provides details about the number of errors that accrue during each second of the scenario run. The errors are grouped by error source—for example, the location in the script or the host name.

The **Vusers with Error Statistics** graph provides details about the number of Vusers that generate errors during scenario execution. The errors are grouped by error source—for example, the location in the script or the load generator name.

User-Defined Data Point Graph

The **User-Defined Data Point** graph displays the real-time values of user-defined data points. You define a data point in your Vuser script by inserting an **lr_user_data_point** function at the appropriate place (**user_data_point** for GUI Vusers and **lr.user_data_point** for Java Vusers).

```
Action1()
{
	lr_think_time(1);
	lr_user_data_point ("data_point_1",1);
	lr_user_data_point ("data_point_2",2);
	return 0;
}
```

For Vuser protocols that support the graphical script representations such as Web and Oracle NCA, you insert a data point as a User Defined step. Data point information is gathered each time the script executes the function or step. For more information about data points, see the *LoadRunner Online Function Reference*.

By default, LoadRunner displays all of the data points in a single graph. The legend provides information about each data point. If desired, you can hide specific data points using the legend below the graphs.

You can also view data points offline, after the completion of the scenario. For more information, see the *LoadRunner Analysis User's Guide*.

Transaction Monitor Graphs

The *Transaction* monitor provides the following graphs:

- Transaction Response Time
- Transactions per Second (Passed)
- Transactions per Second (Failed, Aborted)
- Total Transactions per Second (Passed)

The **Transaction Response Time** graph shows the average response time of transactions in seconds (y-axis) as a function of the elapsed time in the scenario (x-axis).

The **Transactions per Second (Passed)** graph shows the number of successful transactions performed per second (y-axis) as a function of the elapsed time in the scenario (x-axis).

The **Transactions per Second (Failed, Aborted)** graph shows the number of failed and aborted transactions per second (y-axis) as a function of the elapsed time in the scenario (x-axis).

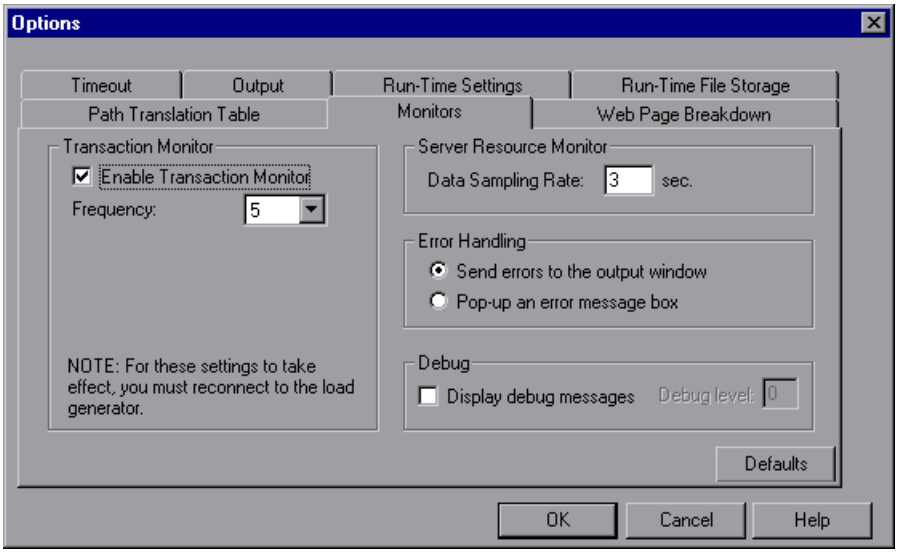
The **Total Transactions per Second (Passed)** graph shows the total number of completed, successful transactions per second (y-axis) as a function of the elapsed time in the scenario (x-axis).

Enabling the Transaction Monitor

The Transaction monitor is enabled by default—it automatically begins monitoring Vuser transactions at the start of a scenario. You can disable the Transaction monitor in order to conserve resources.

To enable the Transaction monitor:

1 Choose **Tools** > **Options** and select the **Monitors** tab.



2 Enable transaction monitoring by selecting the **Enable Transaction Monitor** check box. To disable transaction monitoring, clear the **Enable Transaction Monitor** check box.

Adding Transactions to a Script

If there are no transactions defined in your Vuser script, no data will be displayed in the online graphs. To add transactions to an existing script, edit it using the appropriate tool. The following table shows the script generation tools for each script type:

Script type	Editing tool
GUI Windows	WinRunner
non-GUI Windows	VuGen (Vuser Generator)
SAP	QuickTest for SAP

To add a transaction to a script:

1. Click the **Design** tab to view the list of Vuser groups and scripts.
2. To edit a script for a Vuser group, select the group and click the **View Script** button on the right of the Scenario Groups window to open the script generation tool.

   To edit a script for an individual Vuser, click **Vusers**. Right-click the Vuser whose script you want to edit, and select **View Script** to open the script generation tool.
3. Insert Start and End Transaction functions or markers throughout your script.

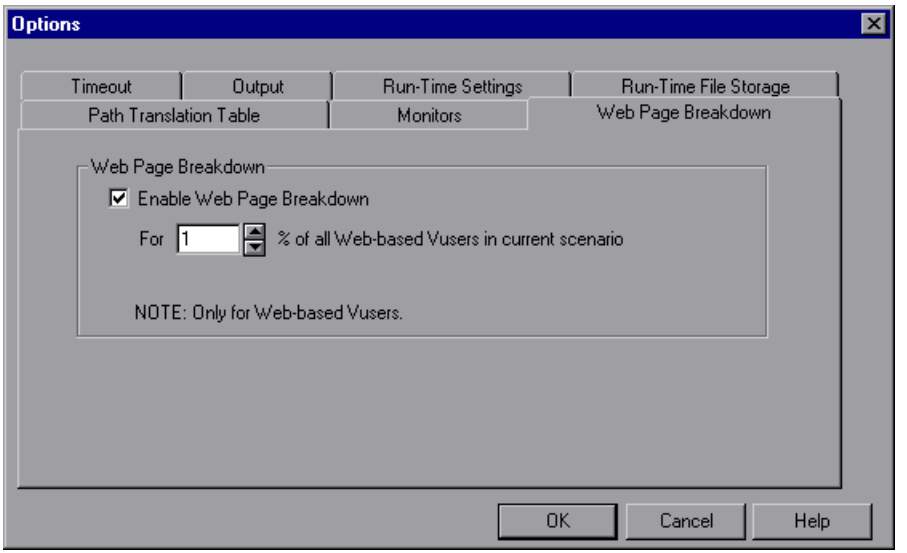
   For more information, see the appropriate user's guide as described in the *Welcome* chapter.

Enabling Web Page Breakdown

In order for the Analysis to generate Web Page Breakdown graphs, which provide you with performance information for each transaction and sub-transaction defined in your script, you must enable the Web page breakdown feature in the Controller before running your scenario.

To enable Web page breakdown:

1 Choose **Tools** > **Options** and select the **Web Page Breakdown** tab.



2 Select **Enable Web Page Breakdown**, and specify the percentage of Web Vusers for which you want Web page breakdown to be performed.

For more information about Web Page Breakdown graphs, see the *LoadRunner Analysis User's Guide*.

16

System and Database Resource Monitoring

You can monitor a machine's system resource usage during a scenario run using LoadRunner's System Resource monitors. You can monitor SQL Server or Oracle database resource usage during a scenario run using LoadRunner's Database Server Resource monitors.

This chapter describes:

- Configuring the Windows, UNIX, and SNMP Resource Monitors
- Starting an rstatd Daemon on UNIX
- Configuring the TUXEDO Monitor
- Working with Database Server Resource Monitors

About Resource Monitoring

A primary factor in a transaction's response time is its resource usage. By monitoring resources during a scenario run, you can determine why a bottleneck occurred on a particular machine. The LoadRunner resource monitors let you keep track of resources used during a scenario. During test execution, LoadRunner displays graphs of the resource usage in real-time.

You can monitor the Windows, TUXEDO, UNIX, or SNMP resources on a machine. The Windows measurements correspond to the built-in counters available from the Windows Performance Monitor.

The UNIX measurements include those available by the *rstatd* daemon: average load, collision rate, context switch rate, CPU utilization, incoming packets error rate, incoming packets rate, interrupt rate, outgoing packets

error rate, outgoing packets rate, page-in rate, page-out rate, paging rate, swap-in rate, swap-out rate, system mode CPU utilization, and user mode CPU utilization.

The TUXEDO monitor can monitor the server, load generator machine, workstation handler, and queue in a TUXEDO system. Note that in order to run the TUXEDO monitor, you must install the TUXEDO client libraries on the machine you want to monitor. For information on configuring the TUXEDO monitor, see "Configuring the TUXEDO Monitor," on page 196.

The SNMP monitor is available for monitoring machines using the Simple Network Management Protocol (SNMP). SNMP monitoring is platform independent.

The SQL server and Oracle database server resource monitors measure statistics for SQL and Oracle database servers. Note that in order to run the Oracle monitor, you must install the Oracle client libraries on the machine you want to monitor.

You can also monitor Web resources, Web server resources, and Web application server resources. See Chapter 17, "Web Resource and Web Server Resource Monitors" and Chapter 18, "Web Application Server Resource Monitors" for details about monitoring Web resources.

A description of each measurement appears in the Resource monitor's Description box when you highlight the measurement.

Note: You must configure an *rstatd* daemon on all UNIX machines being monitored. For information on how to configure an *rstatd* daemon, refer to the UNIX *man* pages, or see "Starting an rstatd Daemon on UNIX", on page 195.

For information on setting monitor options and configuring monitor graphs, see Chapter 14, "Online Monitoring."

Configuring the Windows, UNIX, and SNMP Resource Monitors

The resource monitors are automatically enabled when you execute a scenario. However, you must specify the machine you want to monitor and which resources to monitor for each machine. You can also add or remove machines and resources during the scenario run.

Windows Resources Monitor

Windows NT and Windows 2000 measurements correspond to the built-in counters available from the Windows Performance Monitor.

Note: To monitor a Windows NT or 2000 machine through a firewall, use TCP, port 139.

To configure the Windows Resources monitor:

1. Right-click the graph in the graph view area and choose **Add Measurement(s)**, or choose **Monitors** > **Add Online Measurement**.
2. In the Monitored Server Machines section of the Windows Resources dialog box, click **Add** to enter the server name or IP address of the machine you want to monitor. Select the platform on which the machine runs, and click **OK**.
3. In the Resource Measurements section of the Windows Resources dialog box, select the measurements that you want to monitor.

The following default measurements are available for Windows machines:

Object	Measurement	Description
System	**% Total Processor Time**	The average percentage of time that all the processors on the system are busy executing non-idle threads. On a multi-processor system, if all processors are always busy, this is 100%, if all processors are 50% busy this is 50% and if 1/4th of the processors are 100% busy this is 25%. It can be viewed as the fraction of the time spent doing useful work. Each processor is assigned an Idle thread in the Idle process which consumes those unproductive processor cycles not used by any other threads.
Processor	**% Processor Time (Windows 2000)**	The percentage of time that the processor is executing a non-idle thread. This counter was designed as a primary indicator of processor activity. It is calculated by measuring the time that the processor spends executing the thread of the idle process in each sample interval, and subtracting that value from 100%. (Each processor has an idle thread which consumes cycles when no other threads are ready to run). It can be viewed as the percentage of the sample interval spent doing useful work. This counter displays the average percentage of busy time observed during the sample interval. It is calculated by monitoring the time the service was inactive, and then subtracting that value from 100%.
System	**File Data Operations/sec**	The rate at which the computer issues read and write operations to file system devices. This does not include File Control Operations.

Object	Measurement	Description
System	**Processor Queue Length**	The instantaneous length of the processor queue in units of threads. This counter is always 0 unless you are also monitoring a thread counter. All processors use a single queue in which threads wait for processor cycles. This length does not include the threads that are currently executing. A sustained processor queue length greater than two generally indicates processor congestion. This is an instantaneous count, not an average over the time interval.
Memory	**Page Faults/sec**	This is a count of the page faults in the processor. A page fault occurs when a process refers to a virtual memory page that is not in its Working Set in the main memory. A page fault will not cause the page to be fetched from disk if that page is on the standby list (and hence already in main memory), or if it is in use by another process with which the page is shared.
PhysicalDisk	**% Disk Time**	The percentage of elapsed time that the selected disk drive is busy servicing read or write requests.
Memory	**Pool Nonpaged Bytes**	The number of bytes in the nonpaged pool, a system memory area where space is acquired by operating system components as they accomplish their appointed tasks. Nonpaged pool pages cannot be paged out to the paging file. They remain in main memory as long as they are allocated.

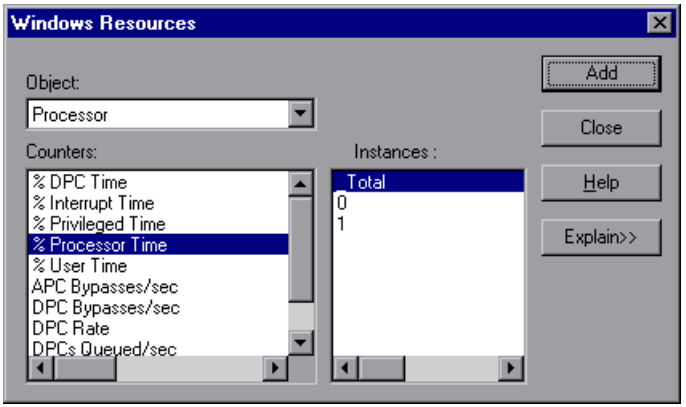
Object	Measurement	Description
Memory	**Pages/sec**	The number of pages read from the disk or written to the disk to resolve memory references to pages that were not in memory at the time of the reference. This is the sum of Pages Input/sec and Pages Output/sec. This counter includes paging traffic on behalf of the system cache to access file data for applications. This value also includes the pages to/from non-cached mapped memory files. This is the primary counter to observe if you are concerned about excessive memory pressure (that is, thrashing), and the excessive paging that may result.
System	**Total Interrupts/sec**	The rate at which the computer is receiving and servicing hardware interrupts. The devices that can generate interrupts are the system timer, the mouse, data communication lines, network interface cards, and other peripheral devices. This counter provides an indication of how busy these devices are on a computer-wide basis. See also Processor:Interrupts/sec.
Objects	**Threads**	The number of threads in the computer at the time of data collection. Notice that this is an instantaneous count, not an average over the time interval. A thread is the basic executable entity that can execute instructions in a processor.
Process	**Private Bytes**	The current number of bytes that the process has allocated that cannot be shared with other processes.

Note: To change the default counters for the Windows machine monitor, see "Changing a Monitor's Default Counters," on page 357.

Note: If you are monitoring a Win2000 machine, some of the NT machine default counters may not be available (such as % Total CPU usage and Interrupts/sec). Proceed to step 4 in order to select counters appropriate for Win2000.

---

4 To select additional measurements, click **Add**. Another Windows Resources dialog box opens, displaying the available measurements and server properties.



5 Select an object, a counter, and an instance. You can select multiple counters using the **Ctrl** key. The instance is relevant only if multiple instances of the highlighted counter are running. For a description of each counter, click **Explain>>** to expand the dialog box.

6 Click **Add** to place the selected counter on the resource list. Add all the desired resources to the list, and click **Close**.

7 Click **OK** in the Windows Resources dialog box to activate the monitor.

Note: If you want to monitor a remote Windows machine that does not use Windows domain security, you must authenticate the Controller machine on the remote Windows machine. To authenticate the Controller machine, create an account, or change the password of the account used to log on to the Controller so that it matches the password and user name used to log on to the remote monitored Windows machine. When the remote Windows machine requests another machine's resources, it sends the logged-in user name and password of the machine requesting the resources.

UNIX Monitor

The UNIX kernel statistics measurements include those available by the *rstatd* daemon: average load, collision rate, context switch rate, CPU utilization, incoming packets error rate, incoming packets rate, interrupt rate, outgoing packets error rate, outgoing packets rate, page-in rate, page-out rate, paging rate, swap-in rate, swap-out rate, system mode CPU utilization, and user mode CPU utilization.

To configure the UNIX monitor:

1. Click the UNIX Resources graph in the graph tree, and drag it into the right pane of the Run view.
2. Right-click the graph in the graph and choose **Add Measurement(s)**, or choose **Monitors** > **Add Online Measurement**.
3. In the Monitored Server Machines section of the UNIX Resources dialog box, click **Add** to enter the server name or IP address of the machine you want to monitor. Select **UNIX** from the list of platforms, and click **OK**.
4. In the Resource Measurements section of the UNIX Resources dialog box, select the default measurements you want to monitor.

The following default measurements are available for the UNIX machine:

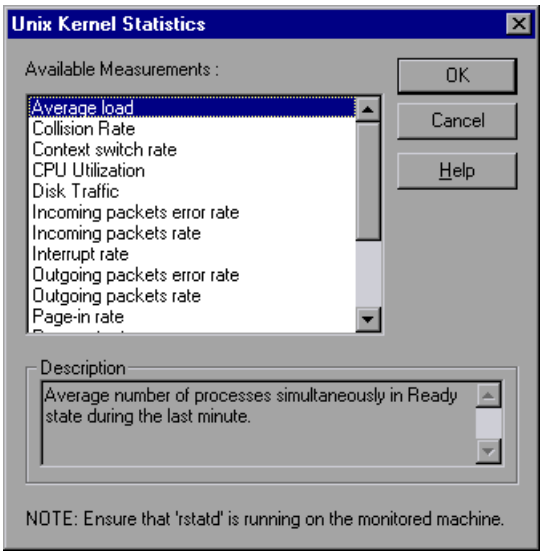
Measurement	Description
Average load	Average number of processes simultaneously in 'Ready' state during the last minute
Collision rate	Collisions per second detected on the Ethernet
Context switches rate	Number of switches between processes or threads, per second
CPU utilization	Percent of time that the CPU is utilized
Disk rate	Rate of disk transfers
Incoming packets error rate	Errors per second while receiving Ethernet packets
Incoming packets rate	Incoming Ethernet packets per second
Interrupt rate	Number of device interrupts per second
Outgoing packets errors rate	Errors per second while sending Ethernet packets
Outgoing packets rate	Outgoing Ethernet packets per second
Page-in rate	Number of pages read to physical memory, per second
Page-out rate	Number of pages written to pagefile(s) and removed from physical memory, per second
Paging rate	Number of pages read to physical memory or written to pagefile(s), per second
Swap-in rate	Number of processes being swapped
Swap-out rate	Number of processes being swapped
System mode CPU utilization	Percent of time that the CPU is utilized in system mode
User mode CPU utilization	Percent of time that the CPU is utilized in user mode

---

Note: To change the default counters for the UNIX monitor, see "Changing a Monitor's Default Counters," on page 357.

---

5 To select additional measurements, click **Add**. The Unix Kernel Statistics dialog box opens, displaying the available measurements and server properties.



6 To add UNIX measurements to the monitor list, select the desired measurements, and click **OK**.

7 Click **OK** in the UNIX Resources dialog box to activate the UNIX monitor.

Note: Ensure that *rstatd* is running on the UNIX machine. For more information, see "Starting an rstatd Daemon on UNIX", on page 195.

SNMP Resource Monitor

The SNMP monitor is available for monitoring any machine that runs an SNMP agent, using the Simple Network Management Protocol (SNMP).

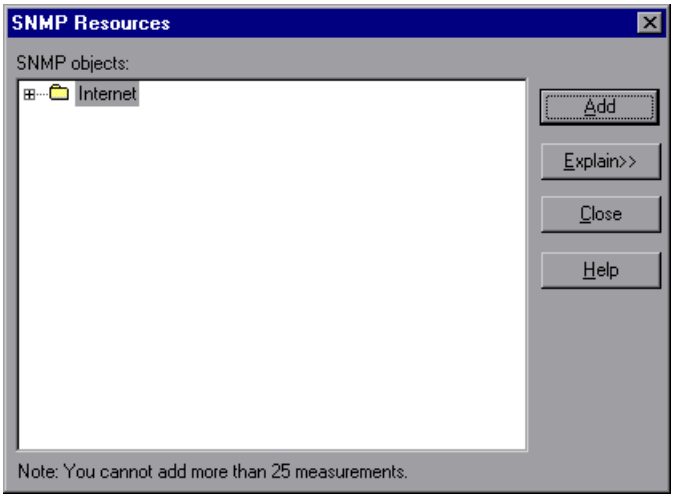
Note: You can specify a port number in the snmp.cfg file. If you do not specify a port, LoadRunner connects to default SNMP port 161. You can also specify a machine name in the following format:
<machine name>:<port number>

To monitor SNMP resources through a firewall, use ports 161 or 162.

To configure the SNMP monitor:

1. Click the SNMP graph in the graph tree, and drag it into the right pane of the Run view.
2. Right-click the graph in the graph and choose **Add Measurement(s)**, or choose **Monitors** > **Add Online Measurement**.
3. In the Monitored Server Machines section of the SNMP dialog box, click **Add** to enter the server name or IP address of the machine you want to monitor. Select the platform on which the machine runs, and click **OK**.
4. In the Resource Measurements section of the SNMP dialog box, click **Add** to select the measurements that you want to monitor.

The SNMP Resources dialog box opens.



5 Browse the SNMP Object tree.

6 To measure an object, select it, and click **Add**. For a description of each resource, click **Explain>>** to expand the dialog box. Add all the desired resources to the list, and click **Close**.

Note: The SNMP monitor can only monitor up to 25 measurements.

---

7 Click **OK** in the SNMP dialog box to activate the monitor.

You can modify the list of resources that you want to monitor at any point during the scenario. Note that a scenario does not have to be active in order for you to monitor the resources on a remote machine.

Starting an rstatd Daemon on UNIX

To monitor UNIX resources, you need to configure the *rstatd* daemon. Before configuring an rstatd daemon, make sure that it is not already activated. It may already be activated, because when a machine receives an rstatd request, the *inetd* on that machine activates the rstatd automatically.

To make sure that the rstatd daemon is not already configured:

The *rup* command reports various machine statistics, including rstatd activation. Run the following command to view the machine statistics:

```
>rup host
```

You can also use **lr_host_monitor** and see if it returns any relevant statistics.

If it returns meaningful statistics, this indicates that the rstatd is activated. If not, or if you receive an error message, rstatd is not configured properly.

To configure rstatd:

1. Run the command: *su root*
2. Go to /etc/inetd.conf and look for the rstatd row (it begins with the word rstatd). If it is commented out (with a #) then remove the comment directive. Save the file.
3. From the command line, run:

   ```
   kill -1 inet_pid
   ```

   where *inet_pid* is the pid of the inetd process. This instructs the inetd to rescan the /etc/inetd.conf file and register all daemons which are uncommented, including the rstatd daemon.
4. Try *rup* again.

   If it still does not indicate that rstatd is activated, contact your system administrator.

Note: To monitor a UNIX machine through a firewall, you must run a UNIX utility called rpcinfo and identify the rstatd's port number. By running rpcinfo -p <hostname>, you will receive a list of all RPC servers registered in the host's portmapper, along with the port number. This list will not change until rstatd is stopped and rerun.

Some firewalls allow you to open an RPC program number instead of a port. In such cases, open program 100001. If are prompted to include a version number, specify versions 3 and 4.

Configuring the TUXEDO Monitor

The TUXEDO monitor allows you to measure and view your TUXEDO client's performance.

Note: The TUXEDO monitor can monitor only one TUXEDO application server at a time.

Before you set up the monitor:

1. Verify that a TUXEDO client is installed on each machine you want to monitor. Install the TUXEDO workstation client libraries (not the native client libraries) on the Controller machine.
2. Define the TUXEDO environment on the monitored machines—set the TUXDIR variable to the TUXEDO installation directory, and add the TUXEDO bin directory to the PATH variable.
3. Configure the TUXEDO application server to accept requests from workstation clients.

To configure the TUXEDO monitor:

1. Click the TUXEDO graph in the graph tree, and drag it into the right pane of the Run view.
2. Right-click the graph in the graph and choose **Add Measurement(s)**, or choose **Monitors** > **Add Online Measurement**.
3. In the Monitored Server Machines section of the TUXEDO dialog box, click **Add** to enter the server name or IP address of the machine you want to monitor. Select the platform on which the machine runs, and click **OK**.
4. Click **Add** in the Resource Measurements section of the TUXEDO dialog box to select the measurements that you want to monitor. You will be prompted to enter information about the TUXEDO server: Login Name, Password, Server Name, Client Name. This information is located in the Logon section of the tpinit.ini file in the recorded script's directory. Note that you can also determine the client name from the **lrt_tpinitialize** statement in the recorded script.

   If you already know the required values, you can manually type them into the dialog box. The format of the server name is //machine name:port number. Alternatively, you can specify the IP address instead of the machine name. The hexadecimal format used by old versions of TUXEDO is also supported. Note that quotation marks should not be used.

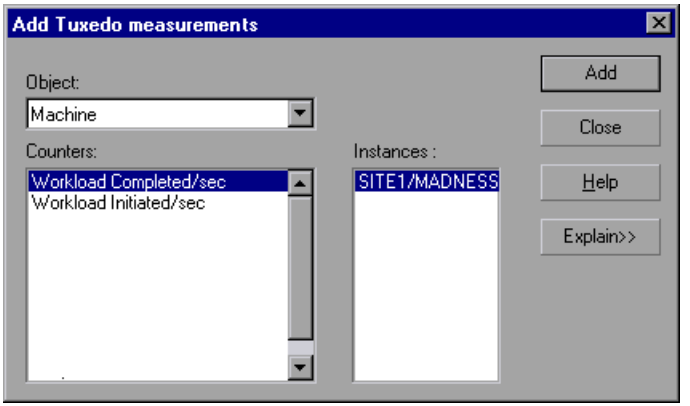
   In the following example of a tpinit.ini file, the TUXEDO monitor was configured for a server named URANUS using port 65535, and a client named bankapp. The logon user name was Smith and the password was mypasswd.

```
[Logon]
LogonServername=//URANUS:65535
LogonUsrName=Smith
LogonCltName=bankapp
LogonGrpName=
LogonPasswd=mypasswd
LogonData=
```

The TUXEDO monitor can only connect to one application server during a Controller session. Once it connects to an application server, that server is

the only one used by the monitor until the Controller is closed. This applies even when all of the counters are deleted from the monitor.

5 Click **OK**. The Add TUXEDO Measurements dialog box opens.



6 Select a TUXEDO object from the Object list. Select the measurements and instances you want to monitor. The following table lists the available TUXEDO monitor measurements:

Monitor	Measurements
Server	**Requests per second** - How many server requests were handled per second
	Workload per second -The workload is a weighted measure of the server requests. Some requests could have a different weight than others. By default, the workload is always 50 times the number of requests.
Machine	**Workload completed per second** - The total workload on all the servers for the machine that was completed, per unit time
	Workload initiated per second - The total workload on all the servers for the machine that was initiated, per unit time
Queue	**Bytes on queue** - The total number of bytes for all the messages waiting in the queue

Monitor	Measurements
Queue	**Messages on queue** - The total number of requests that are waiting on queue. By default this is 0.
Workstation Handler (WSH)	**Bytes received per second** - The total number of bytes received by the workstation handler, per unit time
	Bytes sent per second - The total number of bytes sent back to the clients by the workstation handler, per unit time
	Messages received per second - The number of messages received by the workstation handler, per unit time
	Messages sent per second - The number of messages sent back to the clients by the workstation handler, per unit time
	Number of queue blocks per second - The number of times the queue for the workstation handler blocked, per unit time. This gives an idea of how often the workstation handler was overloaded.

7 Click **Add** to place the selected object on the resource list. Add all the desired objects to the list, and click **Close**.

8 Click **OK** in the TUXEDO dialog box to activate the monitor.

Working with Database Server Resource Monitors

The Database Server Resource monitors measure statistics for SQL server and Oracle databases.

Oracle Monitor

The Oracle monitor measures information from the V$SESSTAT and V$SYSSTAT Oracle V$ tables. In order to measure information in these tables, you must first set up the list of Oracle servers available to the client and configure the desired username/password/server combination. You

must also be able to view the V$SESSION, V$INSTANCE, V$STATNAME, and V$PROCESS tables.

Note: The port you use to monitor an Oracle server through a firewall depends on the configuration of the Oracle server (the tnsnames.ora file).

To set up the Oracle monitor environment:

1. Install the Oracle client libraries on the Controller machine and ensure that the machine has database administrator privileges for the Oracle V$ tables.
2. Ensure that the registry on the Controller machine is updated for the version of Oracle that you are using and has the following key: HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE.
3. Check to see that the Oracle servers you want to monitor are up and running.

Note: It is possible to monitor several Oracle database servers concurrently.

---

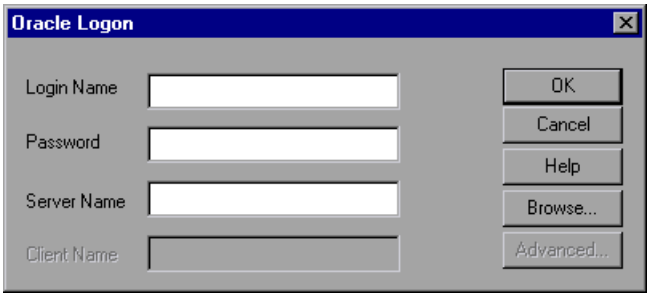
4. Run SQL*Plus from the Controller and attempt to log in to the Oracle server(s) with the desired user name/password/server combination.
5. Type SELECT * FROM V$SYSSTAT to verify that you can view the V$SYSSTAT table on the Oracle server. Use similar queries to verify that you can view the V$SESSTAT, V$SESSION, V$INSTANCE, V$STATNAME, and V$PROCESS tables on the server. Make sure that the Oracle bin directory is in the search path.
6. To change the length of each monitoring sample (in seconds), you need to edit the dat\monitors\vmon.cfg file in the LoadRunner root folder. The default rate is 10 seconds.

Note: If a problem occurs in setting up the Oracle environment, view the error message issued by the Oracle server.

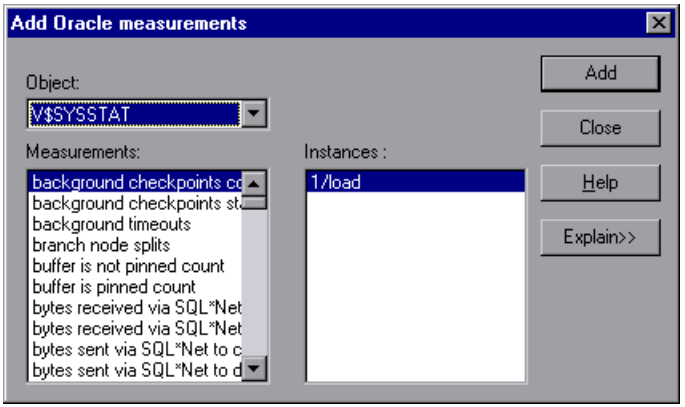
To configure the Oracle monitor:

1. Click the Oracle graph in the graph tree, and drag it into the right pane of the Run view.
2. Right-click the graph and choose **Add Measurement(s)**, or choose **Monitors** > **Add Online Measurement**.
3. In the Monitored Server Machines section of the Oracle dialog box, click **Add** to enter the server name or IP address of the machine you want to monitor. Select the platform on which the machine runs, and click **OK**.
4. In the Resource Measurements section of the Oracle dialog box, click **Add** to select the measurements that you want to monitor.

   The Oracle Logon dialog box opens.



5 Enter your Login Name, Password, and Server Name, and click **OK**. The Add Oracle Measurements dialog box opens.



6 Select an object, a measurement, and an instance. You can select multiple measurements using the **Ctrl** key. The instance is relevant only if multiple instances of the highlighted measurement are running. For a description of each measurement, click **Explain>>** to expand the dialog box. For instructions on creating custom queries, see "Custom Queries," on page 204.

The following measurements are most commonly used when monitoring the Oracle server (from the V$SYSSTAT table):

Measurement	Description
CPU used by this session	This is the amount of CPU time (in 10s of milliseconds) used by a session between the time a user call started and ended. Some user calls can be completed within 10 milliseconds and, as a result, the start and end user-call time can be the same. In this case, 0 milliseconds are added to the statistic. A similar problem can exist in the operating system reporting, especially on systems that suffer from many context switches.
Bytes received via SQL*Net from client	The total number of bytes received from the client over Net8

Measurement	Description
Logons current	The total number of current logons
Opens of replaced files	The total number of files that needed to be reopened because they were no longer in the process file cache
User calls	Oracle allocates resources (Call State Objects) to keep track of relevant user call data structures every time you log in, parse, or execute. When determining activity, the ratio of user calls to RPI calls gives you an indication of how much internal work gets generated as a result of the type of requests the user is sending to Oracle.
SQL*Net roundtrips to/from client	The total number of Net8 messages sent to, and received from, the client
Bytes sent via SQL*Net to client	The total number of bytes sent to the client from the foreground process(es)
Opened cursors current	The total number of current open cursors
DB block changes	Closely related to consistent changes, this statistic counts the total number of changes that were made to all blocks in the SGA that were part of an update or delete operation. These are changes that are generating redo log entries and hence will be permanent changes to the database if the transaction is committed. This statistic is a rough indication of total database work and indicates (possibly on a per-transaction level) the rate at which buffers are being dirtied.
Total file opens	The total number of file opens being performed by the instance. Each process needs a number of files (control file, log file, database file) in order to work against the database.

7 Click **Add** to place the selected measurement on the resource list. Add all the desired resources to the list, and click **Close**.

8 Click **OK** in the Oracle dialog box to activate the monitor.

Note: By default, the database returns the absolute value of a counter. However, by changing the IsRate setting in the dat\monitors\vmon.cfg file to 1, you can instruct the database to report a counter's rate value—the change in the counter per unit time.

Custom Queries

Using the custom query feature, you can define your own query to the Oracle database and view the result of this query—a single numerical value—in the Oracle online monitor graph. By defining your own query, you can monitor not only the V$SYSSTAT and V$SESSTAT table counters that are currently provided by the Oracle monitor, but other tables that contain useful performance information as well.

To create a custom query:

1. In the third line of the vmon.cfg file, CustomCounters=, indicate the number of custom counters you want to create.
2. Create a new section in the vmon.cfg file for the new counter. Each section has the following format:

```
[Custom2]
Name=Number of sessions
Description=This counter returns the number of sessions active.
Query=SELECT COUNT(*) FROM V$SESSION;
IsRate=1
```

3. In the [Custom#] line, assign the next number in the sequence of counters to the new custom counter. Note that the custom counters must be in consecutive order, beginning with the number 0.

4 In the Name line, enter the name of the new counter.

5 In the Description line, enter the description of the counter that you want the help message to contain.

6 In the Query line, enter the text of the SQL query (on one line of the vmon.cfg file) that returns exactly one row from the database. This row must contain one column, a numerical value.

7 In the IsRate line, enter 0 if you want the database to report the counter as an absolute number. If you want the database to report the change in the counter per unit time, enter 1.

Note: Custom queries cannot return negative values.

SQL Server

The SQL Server monitor measures the standard Windows resources on the SQL server machine.

Note: To monitor an SQL server through a firewall, use TCP, port 139.

To configure the SQL server monitor:

1 Click the SQL Server graph in the graph tree, and drag it into the right pane of the Run view.

2 Right-click the graph in the graph and choose **Add Measurement(s)**, or choose **Monitors** > **Add Online Measurement**.

3 In the Monitored Server Machines section of the SQL Server dialog box, click **Add** to enter the server name or IP address of the machine you want to monitor. Select the platform on which the machine runs, and click **OK**.

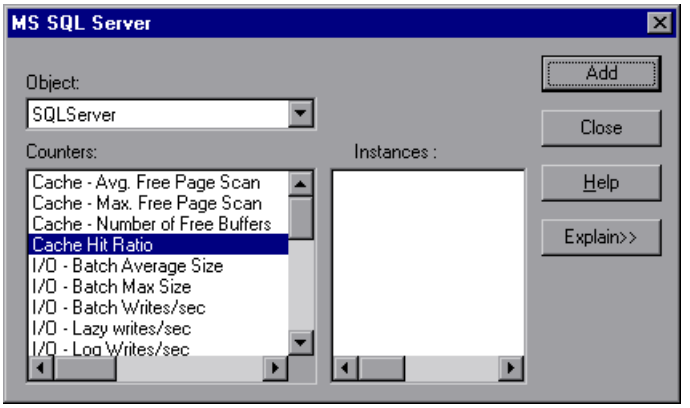
4 In the Resource Measurements section of the SQL Server dialog box, select the measurements you want to monitor. The following table describes the default counters that can be monitored on version 6.5 of the SQL Server:

Measurement	Description
% Total Processor Time (NT)	The average percentage of time that all the processors on the system are busy executing non-idle threads. On a multi-processor system, if all processors are always busy, this is 100%, if all processors are 50% busy this is 50% and if 1/4th of the processors are 100% busy this is 25%. It can be viewed as the fraction of the time spent doing useful work. Each processor is assigned an Idle thread in the Idle process which consumes those unproductive processor cycles not used by any other threads.
% Processor Time (Win 2000)	The percentage of time that the processor is executing a non-idle thread. This counter was designed as a primary indicator of processor activity. It is calculated by measuring the time that the processor spends executing the thread of the idle process in each sample interval, and subtracting that value from 100%. (Each processor has an idle thread which consumes cycles when no other threads are ready to run). It can be viewed as the percentage of the sample interval spent doing useful work. This counter displays the average percentage of busy time observed during the sample interval. It is calculated by monitoring the time the service was inactive, and then subtracting that value from 100%.
Cache Hit Ratio	The percentage of time that a requested data page was found in the data cache (instead of being read from disk)
I/O - Batch Writes/sec	The number of 2K pages written to disk per second, using Batch I/O. The checkpoint thread is the primary user of Batch I/O.
I/O - Lazy Writes/sec	The number of 2K pages flushed to disk per second by the Lazy Writer
I/O - Outstanding Reads	The number of physical reads pending

Measurement	Description
I/O - Outstanding Writes	The number of physical writes pending
I/O - Page Reads/sec	The number of physical page reads per second
I/O - Transactions/sec	The number of Transact-SQL command batches executed per second
User Connections	The number of open user connections

Note: To change the default counters for the SQL Server monitor, see "Changing a Monitor's Default Counters," on page 357.

5 To select additional measurements, click **Add.** The MS SQL Server dialog box opens, displaying the SQL Server object, its counters, and instances.



6 Select a counter and an instance. You can select multiple counters using the **Ctrl** key. The instance is relevant only if multiple instances of the highlighted counter are running. For a description of each counter, click **Explain>>** to expand the dialog box.

7 Click **Add** to place the selected counter on the resource list. Add all the desired resources to the list, and click **Close**.

8 Click **OK** in the SQL Server dialog box to activate the monitor.

Note: Certain measurements or counters are especially useful for determining server performance and isolating the cause of a bottleneck during an initial stress test on the SQL Server. For more information about these counters, see "Useful Counters for Stress Testing," on page 358.

17

Web Resource and Web Server Resource Monitors

You can monitor the hits per second, throughput, HTTP responses per second, and downloaded pages per second using LoadRunner's Web Resource monitor. Using LoadRunner's Web Server Resource monitors, you can monitor the Apache, Netscape, and Microsoft IIS servers during a scenario run and isolate server performance bottlenecks.

This chapter describes:

- Web Resource Monitors
- Activating Web Server Resource Monitors
- Configuring Web Server Resource Monitors
- Monitoring Using a Proxy Server

Web Resource Monitors

The following Web Resource graphs are available:

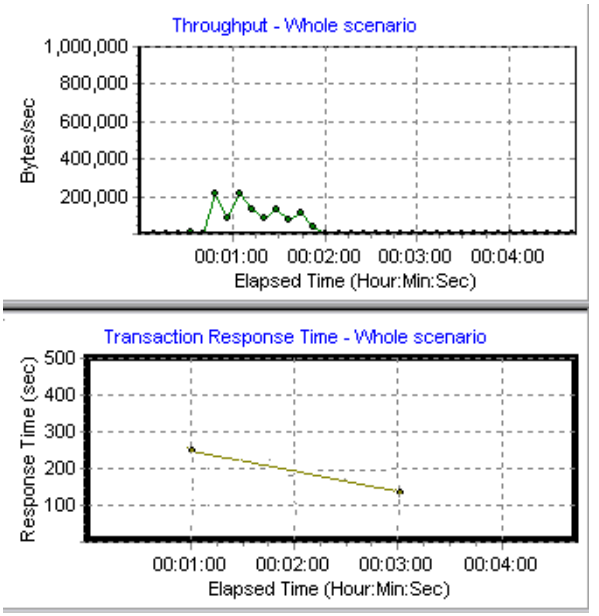
- Hits Per Second
- Throughput
- HTTP Responses per Second
- Pages Downloaded per Second

The **Hits Per Second** graph shows the number of hits (HTTP requests) to the Web server (y-axis) as a function of the elapsed time in the scenario (x-axis). This graph can display the whole scenario, or the last 60, 180, 600, or 3600

seconds. You can compare this graph to the Transaction Response Time graph to see how the number of hits affects transaction performance.

The **Throughput** graph shows the amount of throughput on the Web server (y-axis) during each second of the scenario run (x-axis). Throughput is measured in bytes and represents the amount of data that the Vusers received from the server at any given second. You can compare this graph to the Transaction Response Time graph to see how the throughput affects transaction performance.

In the following example, the Transaction Response time graph is compared with the Throughput graph. It is apparent from the graph that as the throughput increases, the transaction response time also increases. The peak throughput occurred at approximately 1 minute into the scenario. The highest response time also occurred at this time.



The **HTTP Responses per Second** graph shows the number of HTTP status codes—which indicate the status of HTTP requests, for example, "the request was successful," "the page was not found"—(y-axis) returned from the Web server during each second of the scenario run (x-axis), grouped by

status code. You can group the results shown in this graph by script (using the "Group By" function) to locate scripts which generated error codes.

The following table displays a list of HTTP status codes:

Code	Description
200	OK
201	Created
202	Accepted
203	Non-Authoritative Information
204	No Content
205	Reset Content
206	Partial Content
300	Multiple Choices
301	Moved Permanently
302	Found
303	See Other
304	Not Modified
305	Use Proxy
307	Temporary Redirect
400	Bad Request
401	Unauthorized
402	Payment Required
403	Forbidden
404	Not Found
405	Method Not Allowed
406	Not Acceptable
407	Proxy Authentication Required

Code	Description
408	Request Timeout
409	Conflict
410	Gone
411	Length Required
412	Precondition Failed
413	Request Entity Too Large
414	Request - URI Too Large
415	Unsupported Media Type
416	Requested range not satisfiable
417	Expectation Failed
500	Internal Server Error
501	Not Implemented
502	Bad Gateway
503	Service Unavailable
504	Gateway Timeout
505	HTTP Version not supported

For more information on the above status codes and their descriptions, see http://www.w3.org/Protocols/rfc2616/rfc2616-sec10.html#sec10.

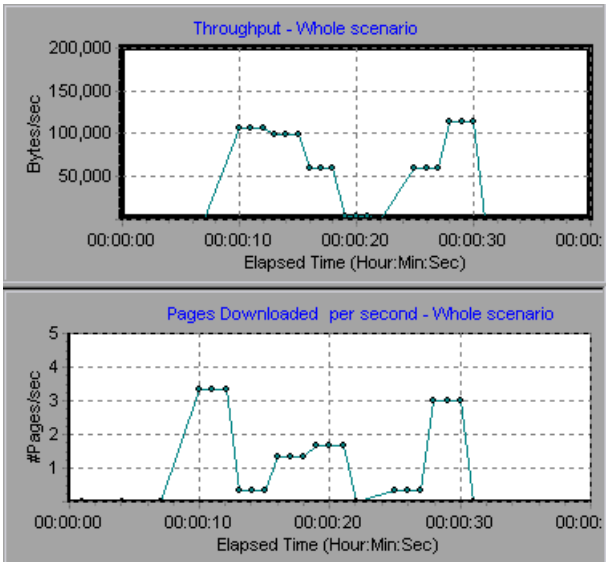
The **Pages Downloaded per Second** graph shows the number of Web pages (y-axis) downloaded from the server during each second of the scenario run (x-axis). This graph helps you evaluate the amount of load Vusers generate, in terms of the number of pages downloaded.

Note: In order to view the Pages Downloaded per Second graph, you must select **Pages per second (HMTL Mode only)** from the run-time settings Preferences tab before running your scenario.

Like throughput, downloaded pages per second is a representation of the amount of data that the Vusers received from the server at any given second.

- The Throughput graph takes into account each resource and its size (for example, the size of each .gif file, the size of each Web page).
- The Pages Downloaded per Second graph takes into account simply the number of pages.

In the following example, the Throughput graph is compared with the Pages Downloaded per Second graph. It is apparent from the graph that throughput is not proportional to the number of pages downloaded per second. For example, between 15 and 16 seconds into the scenario run, the throughput decreased while the number of pages downloaded per second increased.



Activating Web Server Resource Monitors

The Apache, Microsoft IIS, and Netscape Server Resource monitors are not automatically activated when you begin running a scenario. In order to use these monitors, you must first activate them.

To activate the Apache, IIS, and Netscape monitors from the Controller:

1. Click the **Run** tab to open the online monitor view.
2. Click an Apache, IIS, or a Netscape server graph in the tree and drag it into the right pane.
3. Right-click the graph and choose **Add Measurement(s)**, or choose **Monitors** > **Add Online Measurement**.

   The procedures for selecting monitor measurements and configuring the monitors vary according to server type. Refer to the next section,"Configuring Web Server Resource Monitors", for specific instructions.

Note: Certain measurements or counters are especially useful for determining server performance and isolating the cause of a bottleneck during an initial stress test on a Web server. For more information about these counters, see "Useful Counters for Stress Testing" on page 358.

Configuring Web Server Resource Monitors

Each of the sections below describes the configuration settings and limitations for a specific Web server monitor. The following are the monitors described:

- Apache Monitor
- Microsoft IIS Monitor
- Netscape Monitor

Apache Monitor

To monitor an Apache server you need to know the server statistics information URL. A simple way to verify the statistics information URL is to try to view it through the browser.

The URL should be in the following format:

http://<server_name/IP address>:port/server-status?auto

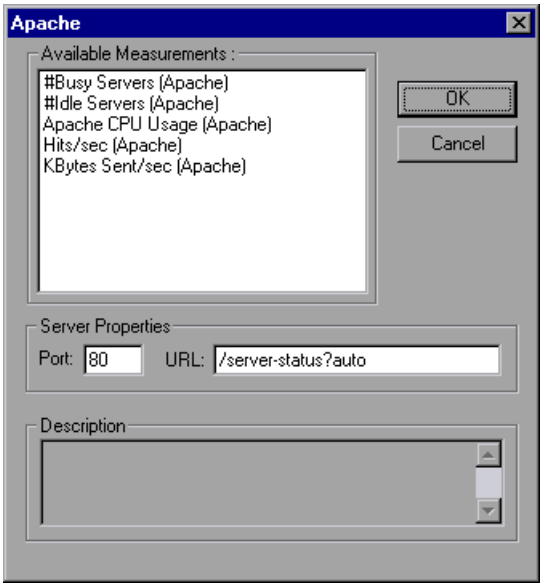
For example:

http://stimpy:80/server-status?auto

To configure the Apache monitor:

1. Right-click the graph in the graph view area and choose **Add Measurement(s)**, or choose **Monitors** > **Add Online Measurement**.
2. In the Monitored Server Machines section of the Apache dialog box, click **Add** to enter the server name or IP address of the machine you want to monitor. Select the platform on which the machine runs, and click **OK**.
3. In the Resource Measurements section of the Apache dialog box, click **Add** to select the measurements that you want to monitor.

Another Apache dialog box opens, displaying the available measurements and server properties.



Select the required measurements. You can select multiple measurements using the **Ctrl** key.

The following table describes the measurements and server properties that can be monitored:

Measurement	Description
# Busy Servers	The number of servers in the Busy state
# Idle Servers	The number of servers in the Idle state
Apache CPU Usage	The percentage of time the CPU is utilized by the Apache server
Hits/sec	The HTTP request rate
KBytes Sent/sec	The rate at which data bytes are sent from the Web server

4 In the Server Properties section, enter the Port number and URL (without the server name), and click **OK**. The default URL is /server-status?auto.

5 Click **OK** in the Apache dialog box to activate the monitor.

Note: The default port number and URL can vary from one server to another. Please consult your Web server administrator.

To change the default server properties:

1 Open the Apache.cfg file in the <LR installation dir>\dat\ monitors\ directory.

2 Edit the following parameters after the Delimiter=: statement:

InfoURL	server statistics information URL
ServerPort	server port number
SamplingRate	rate (milliseconds) at which the LoadRunner monitor will poll the server for the statistics information. If this value is greater than 1000, LoadRunner will use it as its sampling rate. Otherwise, it will use the sampling rate defined in the Monitors tab of the Options dialog box.

Note: To monitor an Apache server through a firewall, use the Web server port (by default, port 80).

Microsoft IIS Monitor

You select measurements for the Microsoft IIS Server monitor using the MS IIS dialog box.

Note: To monitor an IIS server through a firewall, use TCP, port 139.

To configure the IIS server monitor:

1. Right-click the graph in the graph view area and choose **Add Measurement(s)**, or choose **Monitors** > **Add Online Measurement**.
2. In the Monitored Server Machines section of the MS IIS dialog box, click **Add** to enter the server name or IP address of the machine you want to monitor. Select the platform on which the machine runs, and click **OK**.
3. In the Resource Measurements section of the MS IIS dialog box, select the measurements you want to monitor. The following table describes the default measurements that can be monitored:

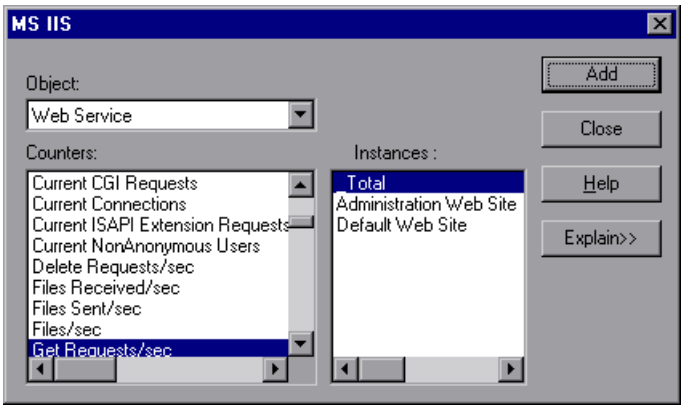
Object	Measurement	Description
Web Service	**Bytes Sent/sec**	The rate at which the data bytes are sent by the Web service
Web Service	**Bytes Received/sec**	The rate at which the data bytes are received by the Web service
Web Service	**Get Requests/sec**	The rate at which HTTP requests using the GET method are made. Get requests are generally used for basic file retrievals or image maps, though they can be used with forms.
Web Service	**Post Requests/sec**	The rate at which HTTP requests using the POST method are made. Post requests are generally used for forms or gateway requests.
Web Service	**Maximum Connections**	The maximum number of simultaneous connections established with the Web service
Web Service	**Current Connections**	The current number of connections established with the Web service

Object	Measurement	Description
Web Service	**Current NonAnonymous Users**	The number of users that currently have a non-anonymous connection using the Web service
Web Service	**Not Found Errors/sec**	The rate of errors due to requests that could not be satisfied by the server because the requested document could not be found. These are generally reported to the client as an HTTP 404 error code.
Process	**Private Bytes**	The current number of bytes that the process has allocated that cannot be shared with other processes.

Note: To change the default counters for the Microsoft IIS Server monitor, see "Changing a Monitor's Default Counters" on page 357.

---

4 To select additional measurements, click **Add.** Another MS IIS dialog box opens, displaying the Web Service object, its counters, and instances.



5 Select a counter and an instance. You can select multiple counters using the **Ctrl** key. The instance is relevant only if multiple instances of the highlighted counter are running. For a description of each counter, click **Explain**>> to expand the dialog box.

6 Click **Add** to place the selected counter on the resource list. Add all the desired resources to the list, and click **Close**.

7 Click **OK** in the MS IIS dialog box to activate the monitor.

Netscape Monitor

To monitor a Netscape server, you need to know the administration server URL. A simple way to verify the administration server URL, is to try to view it through the browser.

The URL should be in the following format:

http://<admin_srv_name/IP addr>:port/https-<admin_srv_name/IP addr>/bin/sitemon?doit

for example:

http://lazarus:12000/https-lazarus.mercury.co.il/bin/sitemon?doit

---

Note: In some server configurations, the URL must contain the administration server name and not the IP address.

In addition, the administration server name may differ from the Netscape server name.

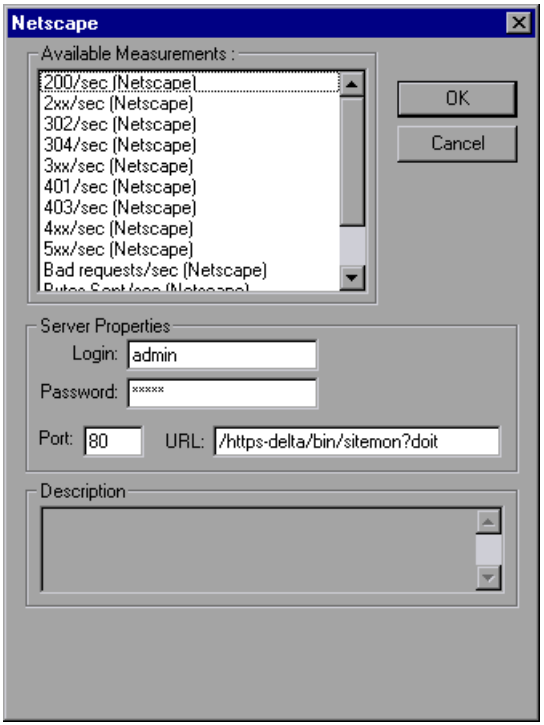
To activate the Netscape monitor from the Controller:

1 Right-click the graph in the graph view area and choose **Add Measurement(s)**, or choose **Monitors** > **Add Online Measurement**.

2 In the Monitored Server Machines section of the Netscape dialog box, click **Add** to enter the server name or IP address of the machine you want to monitor. Select the platform on which the machine runs, and click **OK**.

3 In the Resource Measurements section of the Netscape dialog box, click **Add** to select the measurements that you want to monitor.

Another Netscape dialog box opens, displaying the available measurements and server properties:



Select the required measurements. You can select multiple measurements using the **Ctrl** key.

The following table describes the measurements and server properties that can be monitored:

Measurement	Description
200/sec	The rate of successful transactions being processed by the server
2xx/sec	The rate at which the server handles status codes in the 200 to 299 range

Measurement	Description
302/sec	The rate of relocated URLs being processed by the server
304/sec	The rate of requests for which the server tells the user to use a local copy of a URL instead of retrieving a newer version from the server
3xx/sec	The rate at which the server handles status codes in the 300 to 399 range
401/sec	The rate of unauthorized requests handled by the server
403/sec	The rate of forbidden URL status codes handled by the server
4xx/sec	The rate at which the server handles status codes in the 400 to 499 range
5xx/sec	The rate at which the server handles status codes 500 and higher
Bad requests/sec	The rate at which the server handles bad requests
Bytes sent/sec	The rate at which bytes of data are sent from the Web server
Hits/sec	The HTTP request rate
xxx/sec	The rate of all status codes (2xx-5xx) handled by the server, excluding timeouts and other errors that did return an HTTP status code

4 Fill in the Server Properties:

➤ Enter the user login name and password. The user must have administrator permissions on the server.

➤ Enter the port number and URL (without the server name), and click **OK**. The default URL is /https-<admin_server>/bin/sitemon?doit.

5 Click **OK** in the Netscape dialog box to activate the monitor.

Note: The default port number and URL can vary from one server to another. Please consult the Web server administrator. In some server configurations, the URL must contain the administration server name and not the IP address.

To change the default server properties:

1 Open the netscape.cfg file in the <LR installation dir>\dat\ monitors\ directory.

2 Edit the following parameters in the [Netscape] section:

Counters	number of counters that the LoadRunner Netscape monitor will show you. This value should match the number of counters defined in the file.
InfoURL	server statistics information URL
ServerPort	server port number
ServerLogin	login name to the server
ServerPassword	login password for the login name
SamplingRate	rate (milliseconds) at which the LoadRunner monitor will poll the server for the statistics information. If this value is greater than 1000, LoadRunner will use it as its sampling rate. Otherwise, it will use the sampling rate defined in the Monitors tab of the Options dialog box.

Note: To monitor a Netscape server through a firewall, use the Netscape Administration server port. Configure this port during the server installation process.

Monitoring Using a Proxy Server

LoadRunner allows you to monitor using the Apache and Netscape monitors when there is a proxy server between the Controller and the monitored server. To enable this, you must define settings in your configuration file: in *<LR installation>\dat\monitors\apache.cfg* for the Apache monitor, or in *<LR installation>\dat\monitors\netscape.cfg* for the Netscape monitor.

Before defining settings, you need to determine whether you want LoadRunner to obtain proxy settings from your Internet Explorer connection configuration, or from the proxy settings in the configuration file.

To have LoadRunner read proxy settings from your Internet Explorer connection:

1. In the **Proxy Settings** section of the configuration file, assign **useProxy** a value of 1.
2. If the proxy requires a username, password, or domain, enter these parameters on the lines **proxyUsername**, **proxyPassword**, and **proxyDomain**.

To have LoadRunner read proxy settings from the configuration file:

1. In the **Proxy Settings** section of the configuration file, enter the proxy information on the **httpProxy** line. Use the format: *[<protocol>=][<scheme>://]<proxy>[:<port>][[<protocol>=][<scheme>://] <proxy>[:<port>]]*

   For example: *httpProxy=http=http://my_http_proxy:8080 https=https://my_https_proxy:9000*
2. If the proxy requires a username, password, or domain, enter these parameters on the lines **proxyUsername**, **proxyPassword**, and **proxyDomain**.

To have LoadRunner connect directly to the server (any proxy settings are ignored):

➤ In the **Proxy Settings** section of the configuration file, assign **useProxy** a value of 0.

18

Web Application Server Resource Monitors

You can monitor a Web application server during a scenario run and isolate application server performance bottlenecks using LoadRunner's Web Application Server monitors.

This chapter describes:

- Activating Web Application Server Resource Monitors
- Configuring Web Application Server Resource Monitors

About Web Application Server Resource Monitors

LoadRunner supports the following Web application server versions:

- Ariba Monitor
- Microsoft Active Server Pages (ASP) Monitor
- ATG Dynamo Monitor
- BroadVision Monitor
- ColdFusion Monitor
- Fujitsu INTERSTAGE Monitor
- SilverStream Monitor
- WebLogic Monitor
- WebSphere Monitor

The Web Application Server monitors measure statistics at the Web application servers during scenario execution. You view graphs of Web application server performance during scenario execution in the Run view of the Controller.

Activating Web Application Server Resource Monitors

The Web Application Server monitors are not automatically activated when you begin running a scenario. In order to use these monitors, you must first activate them.

To activate the Web Application Server monitors from the Controller:

1. Click the **Run** tab to open the online monitor view.
2. Click a Web application server graph in the graph tree, and drag it into the right pane.
3. Right-click the graph and choose **Add Measurement(s)**, or choose **Monitors** > **Add Online Measurement**.

   The procedures for selecting monitor measurements and configuring the monitors vary according to server type. Refer to the next section,"Configuring Web Application Server Resource Monitors", for specific instructions.

Configuring Web Application Server Resource Monitors

Each of the sections below describes the configuration settings and limitations for a specific Web Application Server monitor. The following monitors are described:

- Ariba Monitor
- Microsoft Active Server Pages (ASP) Monitor
- ATG Dynamo Monitor
- BroadVision Monitor
- ColdFusion Monitor
- Fujitsu INTERSTAGE Monitor
- SilverStream Monitor
- WebLogic Monitor
- WebSphere Monitor

Ariba Monitor

You select measurements to monitor the Ariba server using the Ariba Monitor Configuration dialog box.

Note: The port you use to monitor an Ariba server through a firewall depends on the configuration of your server.

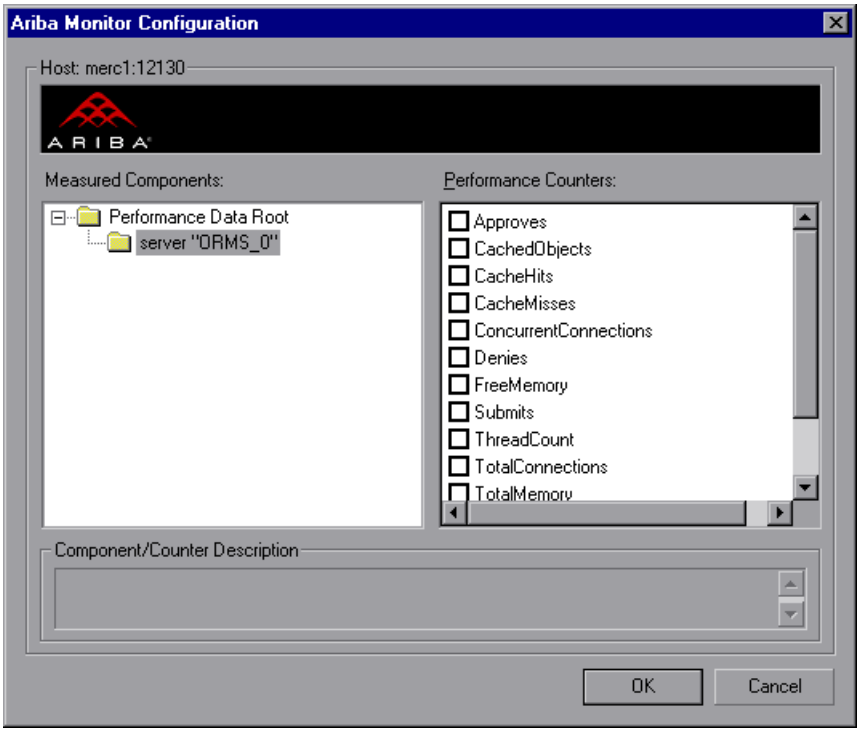
To configure the Ariba monitor:

1. Right-click the graph in the graph view area and choose **Add Measurement(s)**, or choose **Monitors** > **Add Online Measurement**.
2. In the Monitored Server Machines section of the Ariba dialog box, click **Add** to enter the server name or IP address of the machine you want to monitor. Enter the server name or IP address according to the following format: server_name:port_number.

   For example: merc1:12130

3 Click **Add** in the Resource Measurements section of the Ariba dialog box to select the measurements that you want to monitor.

The Ariba Monitor Configuration dialog box opens, displaying the counters that can be monitored:



4 Browse the Measured Components tree.

5 Check the required performance counters in the Ariba Monitor Configuration window's right pane. The following tables describe the counters that can be monitored:

Core Server Performance Counters

Measurement	Description
Requisitions Finished	The instantaneous reading of the length of the worker queue at the moment this metric is obtained. The longer the worker queue, the more user requests are delayed for processing.
Worker Queue Length	The instantaneous reading of the length of the worker queue at the moment this metric is obtained. The longer the worker queue, the more user requests are delayed for processing.
Concurrent Connections	The instantaneous reading of the number of concurrent user connections at the moment this metric is obtained
Total Connections	The cumulative number of concurrent user connections since Ariba Buyer was started.
Total Memory	The instantaneous reading of the memory (in KB) being used by Ariba Buyer at the moment this metric is obtained
Free Memory	The instantaneous reading of the reserved memory (in KB) that is not currently in use at the moment this metric is obtained
Up Time	The amount of time (in hours and minutes) that Ariba Buyer has been running since the previous time it was started
Number of Threads	The instantaneous reading of the number of server threads in existence at the moment this metric is obtained
Number of Cached Objects	The instantaneous reading of the number of Ariba Buyer objects being held in memory at the moment this metric is obtained

Measurement	Description
Average Session Length	The average length of the user sessions (in seconds) of all users who logged out since previous sampling time. This value indicates on average how long a user stays connected to server.
Average Idle Time	The average idle time (in seconds) for all the users who are active since previous sampling time. The idle time is the period of time between two consecutive user requests from the same user.
Approves	The cumulative count of the number of approves that happened during the sampling period. An Approve consists of a user approving one Approvable.
Submits	The cumulative count of the number of Approvables submitted since previous sampling time
Denies	The cumulative count of the number of submitted Approvables denied since previous sampling time
Object Cache Accesses	The cumulative count of accesses (both reads and writes) to the object cache since previous sampling time
Object Cache Hits	The cumulative count of accesses to the object cache that are successful (cache hits) since previous sampling time

System Related Performance Counters

Measurement	Description
Database Response Time	The average response time (in seconds) to the database requests since the previous sampling time
Buyer to DB server Traffic	The cumulative number of bytes that Ariba Buyer sent to DB server since the previous sampling time.
DB to Buyer server Traffic	The cumulative number of bytes that DB server sent to Ariba Buyer since the previous sampling time

Measurement	Description
Database Query Packets	The average number of packets that Ariba Buyer sent to DB server since the previous sampling time
Database Response Packets	The average number of packets that DB server sent to Ariba Buyer since the previous sampling time

6 Click **OK** in the Ariba Monitor Configuration dialog box, and in the Ariba dialog box, to activate the Ariba monitor.

XML Accessibility Verification

Only browsers that are XML-compatible will allow you to view the performance XML file.

To verify whether the XML file is accessible:

Display the XML file through the browser. The URL should be in the following format: http://<server name:server port>/metrics?query=getStats

For example: http://merc1:12130/metrics?query=getStats

Note: In some cases, although the browser is XML-compatible, it may still return the error: The XML page cannot be displayed. In these cases, the XML file can be accessed by the Ariba performance monitor, although it cannot be viewed by the browser.

Microsoft Active Server Pages (ASP) Monitor

You select measurements to monitor the Microsoft ASP application server using the MS Active Server Pages dialog box.

Note: To monitor an ASP server through a firewall, use TCP, port 139.

To configure the ASP monitor:

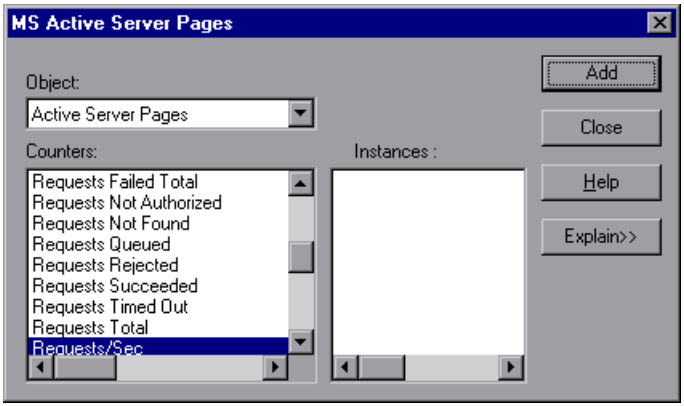
1. Right-click the graph in the graph view area and choose **Add Measurement(s)**, or choose **Monitors** > **Add Online Measurement**.
2. In the Monitored Server Machines section of the MS Active Server Pages dialog box, click **Add** to enter the server name or IP address of the machine you want to monitor. Select the platform on which the machine runs, and click **OK**.
3. In the Resource Measurements section of the MS Active Server Pages dialog box, select the measurements you want to monitor. The following table describes the default counters that can be monitored:

Measurement	Description
Errors per Second	The number of errors per second.
Requests Wait Time	The number of milliseconds the most recent request was waiting in the queue.
Requests Executing	The number of requests currently executing.
Requests Queued	The number of requests waiting in the queue for service.
Requests Rejected	The total number of requests not executed because there were insufficient resources to process them.
Requests Not Found	The number of requests for files that were not found.
Requests/sec	The number of requests executed per second.
Memory Allocated	The total amount of memory, in bytes, currently allocated by Active Server Pages.

Measurement	Description
Errors During Script Run-Time	The number of failed requests due to run-time errors.
Sessions Current	The current number of sessions being serviced.
Transactions/sec	The number of transactions started per second.

Note: To change the default counters for the Microsoft ASP monitor, see "Changing a Monitor's Default Counters" on page 357.

4 To select additional measurements, click **Add.** Another MS Active Server Pages dialog box opens, displaying the Active Server Pages object, its counters, and instances.



5 Select a counter and instance. You can select multiple counters using the **Ctrl** key. The instance is relevant only if multiple instances of the highlighted counter are running. For a description of each counter, click **Explain>>** to expand the dialog box.

6 Click **Add** to place the selected counter on the resource list. Add all the desired resources to the list, and click **Close**.

7 Click **OK** in the MS Active Server Pages dialog box to activate the monitor.

ATG Dynamo Monitor

The ATG Dynamo monitor uses SNMP to retrieve ATG Dynamo server statistics. You define the measurements for the ATG Dynamo monitor using the ATG Dynamo Resources dialog box.

To configure the ATG Dynamo server monitor:

1. Right-click the graph and choose **Add Measurement(s)**, or choose **Monitors** > **Add Online Measurement**.
2. In the Monitored Server Machines section of the ATG Dynamo dialog box, click **Add** to enter the server name or IP address of the machine you want to monitor. Select the platform on which the machine runs, and click **OK**.

Note: You need to define the port number if the ATG SNMP agent is running on a different port than the default ATG SNMP port 8870. You can define the default port for your ATG server in the configuration file, snmp.cfg, located in <LoadRunner root folder>\dat\monitors. For example, if the port used by the SNMP agent on your ATG system is 8888, you should edit the snmp.cfg file as follows:

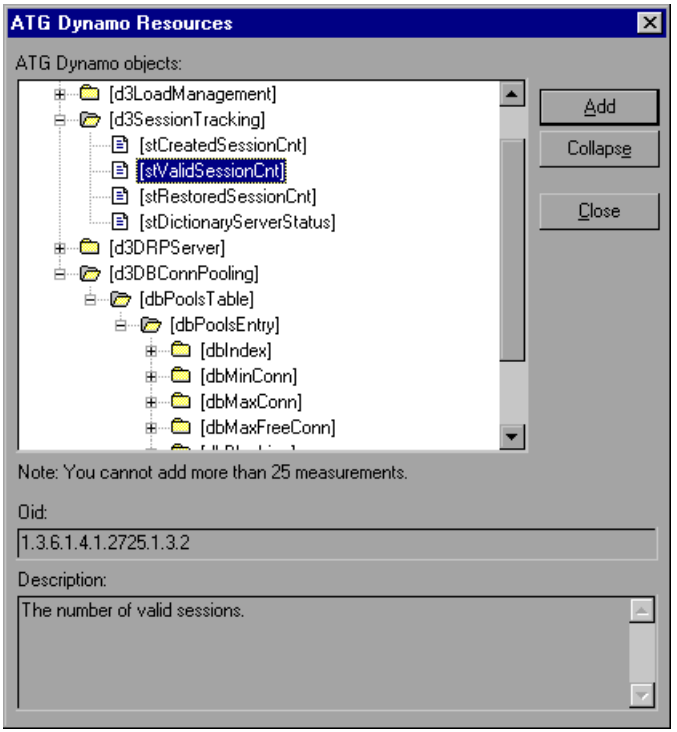
```
; ATG Dynamo
[cm_snmp_mon_atg]
port=8888
```

You can also specify a machine name and port number in the Add Machine dialog box using the following format:
<Server_name:Port_number>
For example: digi:8888

---

3. Click **Add** in the Resource Measurements section of the ATG Dynamo dialog box. The ATG Dynamo Resources dialog box opens.

4 Browse the ATG Dynamo Object tree, and select the measurements you want to monitor.



The following tables describe the measurements that can be monitored:

d3System

Measurement	Description
sysTotalMem	The total amount of memory currently available for allocating objects, measured in bytes
sysFreeMem	An approximation of the total amount of memory currently available for future allocated objects, measured in bytes
sysNumInfoMsgs	The number of system global info messages written

Measurement	Description
sysNumWarningMsgs	The number of system global warning messages written
sysNumErrorMsgs	The number of system global error messages written

d3LoadManagement

Measurement	Description
lmIsManager	True if the Dynamo is running a load manager
lmManagerIndex	Returns the Dynamo's offset into the list of load managing entities
lmIsPrimaryManager	True if the load manager is an acting primary manager
lmServicingCMs	True if the load manager has serviced any connection module requests in the amount of time set as the connection module polling interval
lmCMLDRPPort	The port of the connection module agent
lmIndex	A unique value for each managed entity
lmSNMPPort	The port for the entry's SNMP agent
lmProbability	The probability that the entry will be given a new session
lmNewSessions	Indicates whether or not the entry is accepting new sessions, or if the load manager is allowing new sessions to be sent to the entry. This value is inclusive of any override indicated by lmNewSessionOverride.
lmNewSessionOverride	The override set for whether or not a server is accepting new sessions

d3SessionTracking

Measurement	Description
stCreatedSessionCnt	The number of created sessions
stValidSessionCnt	The number of valid sessions

Measurement	Description
stRestoredSessionCnt	The number of sessions migrated to the server
StDictionaryServerStatus	d3Session Tracking

d3DRPServer

Measurement	Description
drpPort	The port of the DRP server
drpTotalReqsServed	Total number of DRP requests serviced
drpTotalReqTime	Total service time in msecs for all DRP requests
drpAvgReqTime	Average service time in msecs for each DRP request
drpNewessions	True if the Dynamo is accepting new sessions

d3DBConnPooling

Measurement	Description
dbPoolsEntry	A pooling service entry containing information about the pool configuration and current status
dbIndex	A unique value for each pooling service
dbPoolID	The name of the DB connection pool service
dbMinConn	The minimum number of connections pooled
dbMaxConn	The maximum number of connections pooled
dbMaxFreeConn	The maximum number of free pooled connections at a time
dbBlocking	Indicates whether or not the pool is to block out check outs
dbConnOut	Returns the number of connections checked out

Measurement	Description
dbFreeResources	Returns the number of free connections in the pool. This number refers to connections actually created that are not currently checked out. It does not include how many more connections are allowed to be created as set by the maximum number of connections allowed in the pool.
dbTotalResources	Returns the number of total connections in the pool. This number refers to connections actually created and is not an indication of how many more connections may be created and used in the pool.

5 Click **Add** to place the selected counter on the resource list. Add all the desired resources to the list, and click **Close**.

Note: The ATG Dynamo monitor can only monitor up to 25 measurements.

---

6 Click **OK** in the ATG Dynamo dialog box to activate the monitor.

BroadVision Monitor

To monitor a BroadVision server, you must grant the client permission to invoke or launch services on the server.

Note: The port you use to monitor a BroadVision server through a firewall depends on the configuration of your server.

To grant permission for a BroadVision server:

➤ Use the Iona Technologies (Orbix) command for setting user and access permission on a load generator machine:

chmodit [-h <host>] [-v] { <server> | -a <dir> }

{i{+,-}{user,group} | l{+,-}{user,group} }

➤ If you experience problems connecting to the BroadVision monitor, you may need to redefine the permissions to "all."

To invoke permission for all, enter the following command at the BroadVision server command prompt:

chmodit <server> i+all

To launch permission for all, enter the following command at the BroadVision server command prompt:

chmodit <server> l+all

➤ Alternatively, set ORBIX_ACL. Setting ORBIX_ACL=i+all l+all in the BroadVision/Orbix configuration file gives permission to all.

In addition, to monitor a BroadVision server, you need to have JDK 1.2 or higher installed on the Controller machine.

You can install JDK 1.2 by following the download and installation instructions at the following Web site: http://java.sun.com/products/jdk/1.2/

Before activating the monitor, make sure that your Java environment is configured properly.

To configure your Java environment:

1. Open the Windows Registry.
2. The registry should contain the correct path to the Java executable (java.exe) under the JDK 1.2 installation directory. Verify the following registry key:

   HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\App Paths\java.exe

3 The registry should contain the correct path to the Java run-time environment (JRE) under the JRE 1.2 installation directory. Verify the following registry key:

HKEY_LOCAL_MACHINE\SOFTWARE\JavaSoft\Java Runtime Environment\1.2\JavaHome

To configure the BroadVision online monitor:

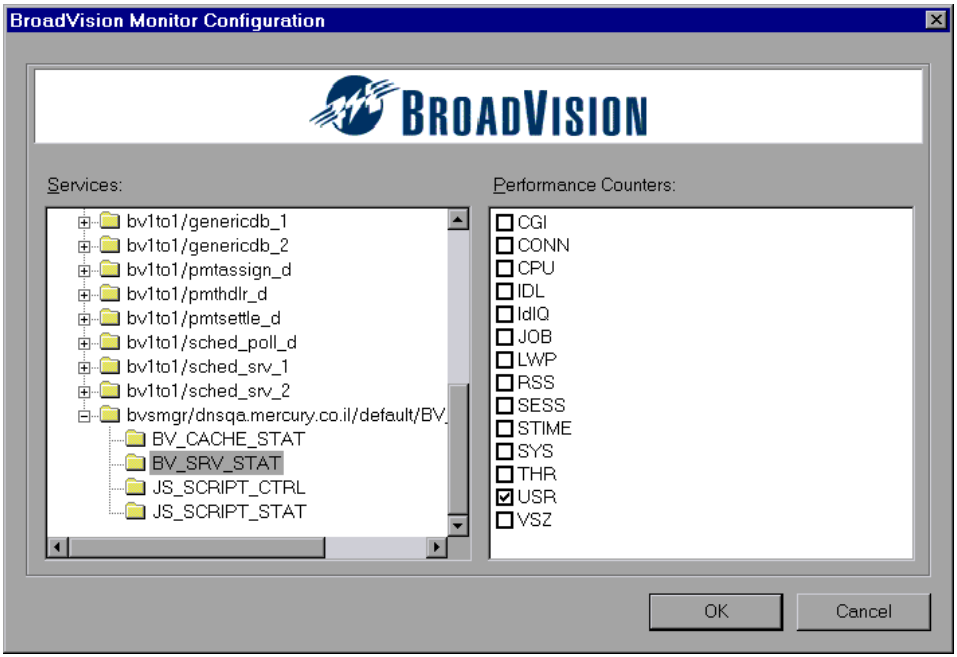
1 Right-click the graph in the graph view area and choose **Add Measurement(s)**, or choose **Monitors** > **Add Online Measurement**.

2 In the Monitored Server Machines section of the BroadVision dialog box, click **Add** to enter the BroadVision server name or IP address with the port number according to the following format: server_name:port_number.

For example: dnsqa:1221. The platform is detected automatically.

3 Click **Add** in the Resource Measurements section of the BroadVision dialog box.

The BroadVision Monitor Configuration dialog box opens, displaying the available measurements:



4 Browse the Services tree and check the required performance counters in the BroadVision Monitor Configuration window's right pane. For a description of the performance counters, see the information below.

5 Click **OK** in the BroadVision Monitor Configuration dialog box, and in the BroadVision dialog box, to activate the BroadVision monitor.

The following table describes the servers/services that can be monitored:

Server	Multiple Instances	Description
adm_srv	No	One-To-One user administration server. There must be one.
alert_srv	No	Alert server handles direct IDL function calls to the Alert system.
bvconf_srv	No	One-To-One configuration management server. There must be one.
cmsdb	Yes	Visitor management database server.
cntdb	Yes	Content database server.
deliv_smtp_d	Yes	Notification delivery server for e-mail type messages. Each instance of this server must have its own ID, numbered sequentially starting with "1".
deliv_comp_d	No	Notification delivery completion processor.
extdbacc	Yes	External database accessor. You need at least one for each external data source.
genericdb	No	Generic database accessor handles content query requests from applications, when specifically called from the application. This is also used by the One-To-One Command Center.
hostmgr	Yes	Defines a host manager process for each machine that participates in One-To-One, but doesn't run any One-To-One servers. For example, you need a hostmgr on a machine that runs only servers. You don't need a separate hostmgr on a machine that already has one of the servers in this list.

Server	Multiple Instances	Description
g1_ofbe_srv	No	Order fulfillment back-end server.
g1_ofdb	Yes	Order fulfillment database server.
g1_om_srv	No	Order management server.
pmtassign_d	No	The payment archiving daemon routes payment records to the archives by periodically checking the invoices table, looking for records with completed payment transactions, and then moving those records into an archive table.
pmthdlr_d	Yes	For each payment processing method, you need one or more authorization daemons to periodically acquire the authorization when a request is made.
pmtsettle_d	Yes	Payment settlement daemon periodically checks the database for orders of the associated payment processing method that need to be settled, and then authorizes the transactions.
sched_poll_d	No	Notification schedule poller scans the database tables to determine when a notification must be run.
sched_srv	Yes	Notification schedule server runs the scripts that generate the visitor notification messages.

Performance Counters

Performance counters for each server/service are divided into logical groups according to the service type.

The following section describes all the available counters under each group. Please note that for some services the number of counters for the same group can be different.

Counter groups:

- BV_DB_STAT
- BV_SRV_CTRL
- BV_SRV_STAT
- NS_STAT
- BV_CACHE_STAT
- JS_SCRIPT_CTRL
- JS_SCRIPT_STAT

BV_DB_STAT

The database accessor processes have additional statistics available from the BV_DB_STAT memory block. These statistics provide information about database accesses, including the count of selects, updates, inserts, deletes, and stored procedure executions.

- DELETE - Count of deletes executions
- INSERT - Count of inserts executions
- SELECT - Count of selects executions
- SPROC - Count of stored procedure executions.
- UPDATE - Count of updates executions

BV_SRV_CTRL

- SHUTDOWN

NS_STAT

The NS process displays the namespace for the current One-To-One environment, and optionally can update objects in a name space.

- Bind
- List
- New
- Rebnd
- Rsolv
- Unbnd

BV_SRV_STAT

The display for Interaction Manager processes includes information about the current count of sessions, connections, idle sessions, threads in use, and count of CGI requests processed.

- **HOST** - Host machine running the process.
- **ID** - Instance of the process (of which multiple can be configured in the bv1to1.conf file), or engine ID of the Interaction Manager.
- **CGI** - Current count of CGI requests processed.
- **CONN** - Current count of connections.
- **CPU** - CPU percentage consumed by this process. If a process is using most of the CPU time, consider moving it to another host, or creating an additional process, possibly running on another machine. Both of these specifications are done in the bv1to1.conf file. The CPU % reported is against a single processor. If a server is taking up a whole CPU on a 4 processor machine, this statistic will report 100%, while the Windows Task Manager will report 25%. The value reported by this statistic is consistent with "% Processor Time" on the Windows Performance Monitor.
- **GROUP** - Process group (which is defined in the **bv1to1.conf** file), or Interaction Manager application name.

- **STIME** - Start time of server. The start times should be relatively close. Later times might be an indication that a server crashed and was automatically restarted.
- **IDL** - Total count of IDL requests received, not including those to the monitor.
- **IdlQ**
- **JOB**
- **LWP** - Number of light-weight processes (threads).
- **RSS** - Resident memory size of server process (in Kilobytes).
- **STIME** - System start time.
- **SESS** - Current count of sessions.
- **SYS** - Accumulated system mode CPU time (seconds).
- **THR** - Current count of threads.
- **USR** - Accumulated user mode CPU time (seconds).
- **VSZ** - Virtual memory size of server process (in kilobytes). If a process is growing in size, it probably has a memory leak. If it is an Interaction Manager process, the culprit is most likely a component or dynamic object (though Interaction Manager servers do grow and shrink from garbage collection during normal use).

BV_CACHE_STAT

Monitors the request cache status.

The available counters for each request are:

- **CNT- Request_Name-HIT** - Count of requests found in the cache.
- **CNT- Request_Name-MAX** - Maximum size of the cache in bytes
- **CNT- Request_Name-SWAP** - Count of items that got swapped out of the cache.
- **CNT- Request_Name-MISS** - Count of requests that were not in the cache.
- **CNT- Request_Name-SIZE** - Count of items currently in the cache.

Cache Metrics

Cache metrics are available for the following items:

- **AD**
- **ALERTSCHED** - Notification schedules are defined in the BV_ALERTSCHED and BV_MSGSCHED tables. They are defined by the One-To-One Command Center user or by an application.
- **CATEGORY_CONTENT**
- **DISCUSSION** - The One-To-One discussion groups provide moderated system of messages and threads of messages aligned to a particular topic. Use the Discussion group interfaces for creating, retrieving and deleting individual messages in a discussion group. To create, delete, or retrieve discussion groups, use the generic content management API. The BV_DiscussionDB object provides access to the threads and messages in the discussion group database.
- **EXT_FIN_PRODUCT**
- **EDITORIAL** - Using the Editorials content module, you can point cast and community cast personalized editorial content, and sell published text on your One-To-One site. You can solicit editorial content, such as investment reports and weekly columns, from outside authors and publishers, and create your own articles, reviews, reports, and other informative media. In addition to text, you can use images, sounds, music, and video presentations as editorial content.
- **INCENTIVE** - Contains sales incentives
- **MSGSCHED** - Contains the specifications of visitor-message jobs. Notification schedules are defined in the BV_ALERTSCHED and BV_MSGSCHED tables. They are defined by the One-To-One Command Center user or by an application.
- **MSGSCRIPT** - Contains the descriptions of the JavaScripts that generate visitor messages and alert messages. Contains the descriptions of the JavaScripts that generate targeted messages and alert messages. Use the Command Center to add message script information to this table by selecting the Visitor Messages module in the Notifications group. For more information, see the Command Center User's Guide.

- **PRODUCT** - BV_PRODUCT contains information about the products that a visitor can purchase.
- **QUERY** - BV_QUERY contains queries.
- **SCRIPT** - BV_SCRIPT contains page scripts.
- **SECURITIES**
- **TEMPLATE** - The Templates content module enables you to store in the content database any BroadVision page templates used on your One-To-One site. Combining BroadVision page templates with BroadVision dynamic objects in the One-To-One Design Center application is one way for site developers to create One-To-One Web sites. If your developers use these page templates, you can use the Command Center to enter and manage them in your content database. If your site doesn't use BroadVision page template, you will not use this content module.

JS_SCRIPT_CTRL

- CACHE
- DUMP
- FLUSH
- METER
- TRACE

JS_SCRIPT_STAT

- ALLOC
- ERROR
- FAIL
- JSPPERR
- RELEASE
- STOP
- SUCC
- SYNTAX

ColdFusion Monitor

You select measurements to monitor the ColdFusion server using the ColdFusion dialog box.

Note: To monitor a ColdFusion server through a firewall, use TCP, port 139.

To configure the ColdFusion monitor:

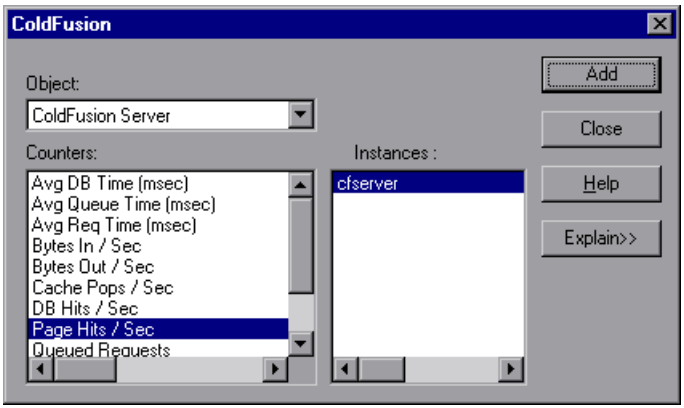
1. Right-click the graph in the graph view area and choose **Add Measurement(s)**, or choose **Monitors** > **Add Online Measurement**.
2. In the Monitored Server Machines section of the ColdFusion dialog box, click **Add** to enter the server name or IP address of the machine you want to monitor. Select the platform on which the machine runs, and click **OK**.
3. In the Resource Measurements section of the ColdFusion dialog box, select the measurements you want to monitor. The following table describes the default counters that can be monitored:

Measurement	Description
Avg. Queue Time (msec)	The running average of the amount of time, in milliseconds, that requests spent waiting in the ColdFusion input queue before ColdFusion began to process the request.
Avg Req Time (msec)	The running average of the total amount of time, in milliseconds, that it takes ColdFusion to process a request. In addition to general page processing time, this value includes both queue time and database processing time.
Bytes Out/sec	The number of bytes per second returned by the ColdFusion server.
Page Hits/sec	This is the number of Web pages processed per second by the ColdFusion server.

Measurement	Description
Queued Requests	The number of requests currently waiting to be processed by the ColdFusion server.
Running Requests	The number of requests currently being actively processed by the ColdFusion server.

Note: To change the default counters for the ColdFusion Server monitor, see "Changing a Monitor's Default Counters" on page 357.

4 To select additional measurements, click **Add.** Another ColdFusion dialog box opens, displaying the ColdFusion object, its counters, and instances.



5 Select a counter and instance. You can select multiple counters using the **Ctrl** key. The instance is relevant only if multiple instances of the highlighted counter are running. For a description of each counter, click **Explain>>** to expand the dialog box.

6 Click **Add** to place the selected counter on the resource list. Add all the desired resources to the list, and click **Close**.

7 Click **OK** in the ColdFusion dialog box to activate the monitor.

Fujitsu INTERSTAGE Monitor

The Fujitsu INTERSTAGE monitor uses SNMP to retrieve Fujitsu INTERSTAGE server statistics. You define the measurements for the Fujitsu INTERSTAGE monitor using the Fujitsu INTERSTAGE SNMP Resources dialog box.

To configure the Fujitsu INTERSTAGE server monitor:

1. Right-click the graph in the graph view area and choose **Add Measurement(s)**, or choose **Monitors** > **Add Online Measurement**.
2. In the Monitored Server Machines section of the Fujitsu INTERSTAGE dialog box, click **Add** to enter the server name or IP address of the machine you want to monitor. Select the platform on which the machine runs, and click **OK**.

Note: You need to define the port number if the Fujitsu INTERSTAGE SNMP agent is running on a different port than the default SNMP port 161. Enter the following information in the Add Machine dialog box:
<Server_name:Port_number>
For example: digi:8888

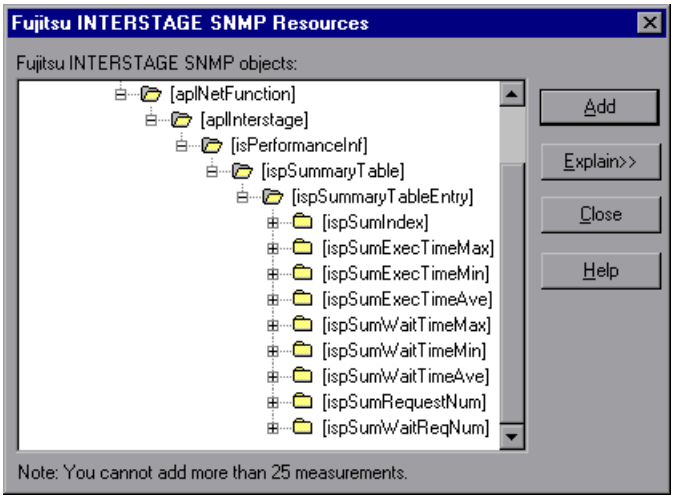
In addition, you can define the default port for your Fujitsu INTERSTAGE server in the configuration file, snmp.cfg, located in <LoadRunner root folder>\dat\monitors. For example, if the port used by the SNMP agent on your Fujitsu INTERSTAGE system is 8888, you should edit the snmp.cfg file as follows:

```
; Fujitsu INTERSTAGE
[cm_snmp_mon_isp]
port=8888
```

---

3. Click **Add** in the Resource Measurements section of the Fujitsu INTERSTAGE dialog box. The Fujitsu INTERSTAGE SNMP Resources dialog box opens.

4 Browse the Fujitsu INTERSTAGE SNMP Object tree, and select the measurements you want to monitor.



The following tables describe the measurements that can be monitored:

Measurement	Description
IspSumObjectName	The object name of the application for which performance information is measured
IspSumExecTimeMax	The maximum processing time of the application within a certain period of time
IspSumExecTimeMin	The minimum processing time of the application within a certain period of time
IspSumExecTimeAve	The average processing time of the application within a certain period of time
IspSumWaitTimeMax	The maximum time required for INTERSTAGE to start an application after a start request is issued
IspSumWaitTimeMin	The minimum time required for INTERSTAGE to start an application after a start request is issued
IspSumWaitTimeAve	The average time required for INTERSTAGE to start an application after a start request is issued

Measurement	Description
IspSumRequestNum	The number of requests to start an application
IspSumWaitReqNum	The number of requests awaiting application activation

5 Click **Add** to place the selected counter on the resource list. Add all the desired resources to the list, and click **Close**.

Note: The Fujitsu INTERSTAGE monitor can only monitor up to 25 measurements.

---

6 Click **OK** in the Fujitsu INTERSTAGE dialog box to activate the monitor.

SilverStream Monitor

To monitor a SilverStream server you need to know the server statistics information URL. A simple way to verify the statistics URL is to access it from a browser.

The URL should be in the following format:

http://<srv_name/IP addr>[:#port]/SilverStream/Statistics

for example:

http://199.203.78.57:80/SilverStream/Statistics

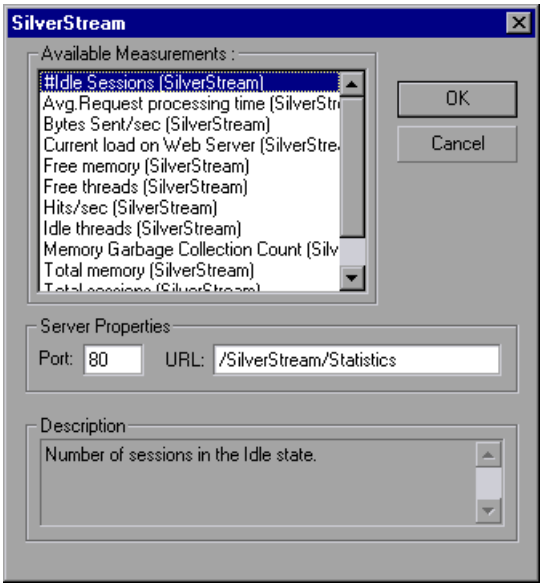
To configure the SilverStream monitor:

1 Right-click the graph in the graph view area and choose **Add Measurement(s)**, or choose **Monitors** > **Add Online Measurement**.

2 In the Monitored Server Machines section of the SilverStream dialog box, click **Add** to enter the server name or IP address of the machine you want to monitor. Select the platform on which the machine runs, and click **OK**.

3 In the Resource Measurements section of the SilverStream dialog box, click **Add** to select the measurements that you want to monitor.

Another SilverStream dialog box opens, displaying the available measurements and server properties.



Select the required measurements. You can select multiple measurements using the **Ctrl** key.

The following table describes the measurements and server properties that can be monitored:

Measurement	Description
#Idle Sessions	The number of sessions in the Idle state.
Avg. Request processing time	The average request processing time.
Bytes Sent/sec	The rate at which data bytes are sent from the Web server.
Current load on Web Server	The percentage of load utilized by the SilverStream server, scaled at a factor of 25.
Hits/sec	The HTTP request rate.
Total sessions	The total number of sessions.

Measurement	Description
Free memory	The total amount of memory in the Java Virtual Machine currently available for future allocated objects.
Total memory	The total amount of memory in the Java Virtual Machine.
Memory Garbage Collection Count	The total number of times the JAVA Garbage Collector has run since the server was started.
Free threads	The current number of threads not associated with a client connection and available for immediate use.
Idle threads	The number of threads associated with a client connection, but not currently handling a user request.
Total threads	The total number of client threads allocated.

4 In the Server Properties section, enter the Port number and URL (without the server name), and click **OK**. The default URL is /SilverStream/Statistics.

5 Click **OK** in the SilverStream dialog box to activate the monitor.

Note: The default port number and URL can vary from one server to another. Please consult the Web server administrator.

To change the default server properties:

1 Open the SilverStream.cfg file in the <LR installation dir>\dat\ monitors\ directory.

2 Edit the following parameters at the end of the file:

InfoURL	server statistics information URL
ServerPort	server port number
SamplingRate	rate (milliseconds) at which the LoadRunner monitor will poll the server for the statistics information. If this value is greater than 1000, LoadRunner will use it as its sampling rate. Otherwise, it will use the sampling rate defined in the Monitors tab of the Options dialog box.

Note: To monitor a SilverStream server through a firewall, use the Web server port (by default, port 80).

WebLogic Monitor

The WebLogic monitor uses SNMP to retrieve server statistics. To use this monitor, you must make sure that the WebLogic SNMP agent is installed and activated on the server. For instructions on installing the SNMP agent, see http://www.weblogic.com/docs51/admindocs/snmpagent.html.

Note: To monitor a WebLogic server, use port 161 or 162, depending on the configuration of the agent.

To configure the WebLogic monitor:

1 Right-click the graph in the graph view area and choose **Add Measurement(s)**, or choose **Monitors** > **Add Online Measurement**.

2 In the Monitored Server Machines section of the WebLogic dialog box, click **Add** to enter the server name or IP address of the machine you want to monitor. Select the platform on which the machine runs, and click **OK**.

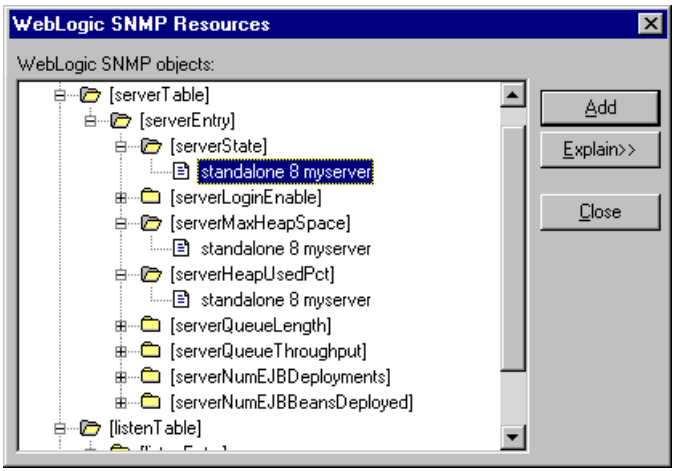
Note: You need to define the port number if the WebLogic SNMP agent is running on a different port than the default SNMP port. Enter the following information in the Add Machine dialog box:
<Server_name:Port_number>
For example: digi:8888

In addition, you can define the default port for your WebLogic server in the configuration file, snmp.cfg, located in <LoadRunner root folder>\dat\monitors. For example, if the port used by the SNMP agent on your WebLogic server is 8888, you should edit the snmp.cfg file as follows:
; WebLogic
[cm_snmp_mon_isp]
port=8888

---

3 In the Resource Measurements section of the WebLogic dialog box, click **Add** to select the measurements that you want to monitor. The WebLogic SNMP Resources dialog box displays the available measurements.

4 Browse the WebLogic SNMP Objects tree.



5 To measure an object, select it, and click **Add**. The following tables describe the measurements and server properties that can be monitored:

Server Table

The Server Table lists all WebLogic servers that are being monitored by the agent. A server must be contacted or be reported as a member of a cluster at least once before it will appear in this table. Servers are only reported as a member of a cluster when they are actively participating in the cluster, or shortly thereafter.

Measurement	Description
ServerState	The state of the WebLogic server, as inferred by the SNMP agent. *Up* implies that the agent can contact the server. *Down* implies that the agent cannot contact the server.
ServerLoginEnable	This value is true if client logins are enabled on the server.
ServerMaxHeapSpace	The maximum heap size for this server, in KB
ServerHeapUsedPct	The percentage of heap space currently in use on the server
ServerQueueLength	The current length of the server execute queue
ServerQueueThroughput	The current throughput of execute queue, expressed as the number of requests processed per second
ServerNumEJBDeployment	The total number of EJB deployment units known to the server
ServerNumEJBBeansDeployed	The total number of EJB beans actively deployed on the server

Listen Table

The Listen Table is the set of protocol, IP address, and port combinations on which servers are listening. There will be multiple entries for each server: one for each protocol, ipAddr, port) combination. If clustering is used, the clustering-related MIB objects will assume a higher priority.

Measurement	Description
ListenPort	Port number.
ListenAdminOK	True if admin requests are allowed on this (protocol, ipAddr, port); otherwise false
ListenState	Listening if the (protocol, ipAddr, port) is enabled on the server; not Listening if it is not. The server may be listening but not accepting new clients if its server Login Enable state is false. In this case, existing clients will continue to function, but new ones will not.

ClassPath Table

The ClassPath Table is the table of classpath elements for Java, WebLogic server, and servlets. There are multiple entries in this table for each server. There may also be multiple entries for each path on a server. If clustering is used, the clustering-related MIB objects will assume a higher priority.

Measurement	Description
CPType	The type of CP element: Java, WebLogic, servlet. A Java CPType means the cpElement is one of the elements in the normal Java classpath. A WebLogic CPType means the cpElement is one of the elements in weblogic.class.path. A servlet CPType means the cpElement is one of the elements in the dynamic servlet classpath.
CPIndex	The position of an element within its path. The index starts at 1.

Note: The WebLogic monitor can only monitor up to 25 measurements.

---

6 After selecting and adding the required objects, click **Close**.

7 Click **OK** in the WebLogic dialog box to activate the monitor.

WebSphere Monitor

To monitor the IBM WebSphere application server, you must first install the appropriate IBM WebSphere servlet patch.

To install the IBM WebSphere servlet patch:

1 Unzip the IBM_WebSphere<version#>_Servlet.zip file.

2 Copy **xm14j.jar**, **performance.dtd**, and **perf.jar** (**perf35.jar** for version 3.5; **perf352.jar** for version 3.5.2) into the default_host\default_app servlets directory on the monitored machine. For example, if the IBM WebSphere folder is installed on drive e, the files should be copied to: e://WebSphere\AppServer\hosts\default_host\default_app\servlets.

3 Verify that the servlet is running properly and that the performance data is being generated. A simple way to verify that the performance data is accessible is to display it in a Web browser. The URL must be in the following format:

http://<server:port_number>/<servlet_folder>/com.ibm.ivb.epm.servlet.PerformanceServlet

For example: http://websphere.mercury.co.il:81/servlet/com.ibm.ivb.epm.servlet.PerformanceServlet

---

Note: Only browsers that are XML-compatible will allow you to view the performance XML file.

To configure the WebSphere server for version 3.5.2:

1. To enable the EPM counters, which are by default set to "none," right-click the application you are monitoring in the WebSphere Administrator's Console browser, and select **Performance**. Expand the Performance Modules tree in the dialog box that opens. In order to manage different levels of performance data, right-click the performance modules and choose a performance level. Click the **Set** button.
2. To enable the "servlet" counters, extract <WAS_HOME>\lib\ibmwebas.jar to your servlet directory. Modify the content of the com/ibm/servlet/appserver.properties file as follows:

   ```
   listeners.application=com.ibm.servlet.engine.EPMApplicationListener
   com.ibm.servlet.debug.OLTServletManager
   #listeners.application=
   ```

3. Repackage the <WAS_HOME>\lib\ibmwebas.jar file.

Note: You can use WSCP to set the EPM data. When an appserver is started, the attribute format for EPM should be:
EpmSpec epm=high:epm.beanMethodData=none
Disabling the method level data is recommended.

To configure the WebSphere monitor:

1. Right-click the graph in the graph view area and choose **Add Measurement(s)**, or choose **Monitors** > **Add Online Measurement**.
2. In the Monitored Server Machines section of the WebSphere dialog box, click **Add** to enter the server name or IP address of the machine you want to monitor. Select the platform on which the machine runs, and click **OK**.

Note: By default, LoadRunner uses servlet as the Web alias of the servlet directory. If the Web alias for the servlet directory is not servlet, you must specify the servlet directory Web alias in the Add Machine dialog box according to the following format:

http://<server:port_number>/<servlet_dir_alias>

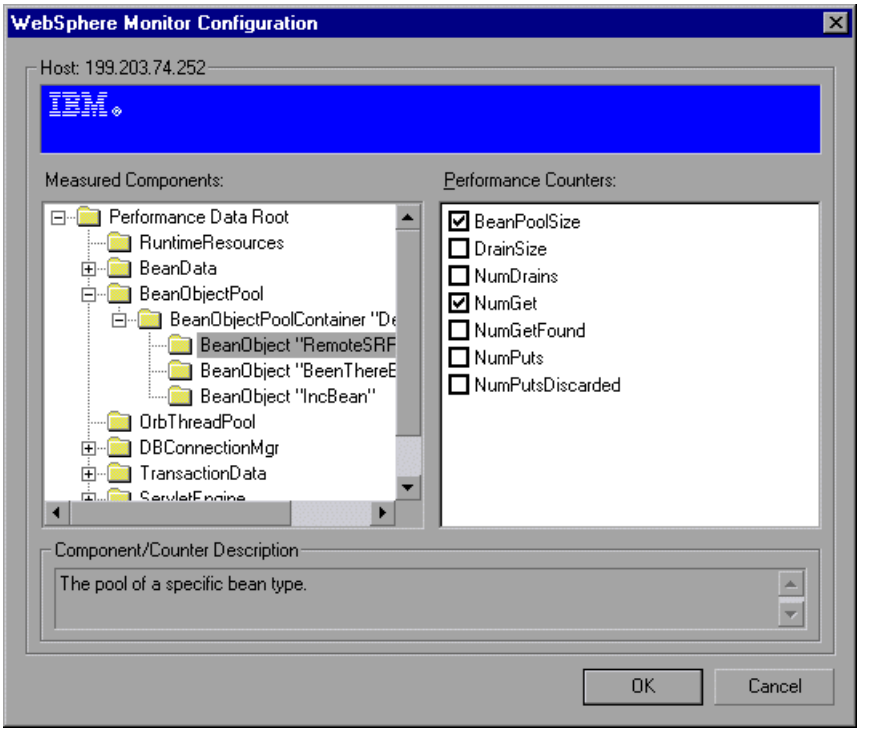
For example: http://mercury/servlet2

Using this method, you can monitor as many application servers as you want—whether they are installed on the same machine, or on different machines.

---

3 In the Resource Measurements section of the WebSphere dialog box, click **Add** to select the measurements that you want to monitor. The WebSphere Monitor Configuration dialog box displays the available measurements:

4 Browse the Measured Components tree.



5 Check the required performance counters in the WebSphere Monitor Configuration window's right pane.

The following tables describe the counters that can be monitored:

➤ **Run-Time Resources**

Contains resources related to the Java Virtual Machine run-time, as well as the ORB.

Measurement	Description
MemoryFree	The amount of free memory remaining in the Java Virtual Machine
MemoryTotal	The total memory allocated for the Java Virtual Machine
MemoryUse	The total memory in use within the Java Virtual Machine

➤ **BeanData**

Every home on the server provides performance data, depending upon the type of bean deployed in the home. The top level bean data holds an aggregate of all the containers.

Measurement	Description
BeanCreates	The number of beans created. Applies to an individual bean that is either 'stateful' or 'entity'
EntityBeanCreates	The number of entity beans created
BeanRemoves	The number of entity beans pertaining to a specific bean that have been removed. Applies to an individual bean that is either 'stateful' or 'entity'
EntityBeanRemoves	The number of entity beans removed
StatefulBeanCreates	The number of stateful beans created
StatefulBeanRemoves	The number of stateful bean removed
BeanPassivates	The number of bean passivates pertaining to a specific bean. Applies to an individual bean that is either 'stateful' or 'entity'

Measurement	Description
EntityBeanPassivates	The number of entity bean passivates
StatefulBeanPassivates	The number of stateful bean passivates
BeanActivates	The number of bean activates pertaining to a specific bean. Applies to an individual bean that is either 'stateful' or 'entity'
EntityBeanActivates	The number of entity bean activates
StatefulBeanActivates	The number of stateful bean activates
BeanLoads	The number of times the bean data was loaded. Applies to entity
BeanStores	The number of times the bean data was stored in the database. Applies to entity
BeanInstantiates	The number of times a bean object was created. This applies to an individual bean, regardless of its type.
StatelessBeanInstantiates	The number of times a stateless session bean object was created
StatefulBeanInstantiates	The number of times a stateful session bean object was created
EntityBeanInstantiates	The number of times an entity bean object was created
BeanDestroys	The number of times an individual bean object was destroyed. This applies to any bean, regardless of its type
StatelessBeanDestroys	The number of times a stateless session bean object was destroyed
StatefulBeanDestroys	The number of times a stateful session bean object was destroyed
EntityBeanDestroys	The number of times an entity bean object was destroyed

Measurement	Description
BeansActive	The average number of instances of active beans pertaining to a specific bean. Applies to an individual bean that is either 'stateful' or 'entity'
EntityBeansActive	The average number of active entity beans
StatefulBeansActive	The average number of active session beans
BeansLive	The average number of bean objects of this specific type that are instantiated but not yet destroyed. This applies to an individual bean, regardless of its type.
StatelessBeansLive	The average number of stateless session bean objects that are instantiated but not yet destroyed
StatefulBeansLive	The average number of stateful session bean objects that are instantiated but not yet destroyed
EntityBeansLive	The average number of entity bean objects that are instantiated but not yet destroyed
BeanMethodRT	The average method response time for all methods defined in the remote interface to this bean. Applies to all beans
BeanMethodActive	The average number of methods being processed concurrently. Applies to all beans
BeanMethodCalls	The total number of method calls against this bean

➤ **BeanObjectPool**

The server holds a cache of bean objects. Each home has a cache and there is therefore one BeanObjectPoolContainer per container. The top level BeanObjectPool holds an aggregate of all the containers data.

Measurement	Description
BeanObjectPoolContainer	The pool of a specific bean type
BeanObject	The pool specific to a home

Measurement	Description
NumGet	The number of calls retrieving an object from the pool
NumGetFound	The number of calls to the pool that resulted in finding an available bean
NumPuts	The number of beans that were released to the pool
NumPutsDiscarded	The number of times releasing a bean to the pool resulted in the bean being discarded because the pool was full
NumDrains	The number of times the daemon found the pool was idle and attempted to clean it
DrainSize	The average number of beans discarded by the daemon during a clean
BeanPoolSize	The average number of beans in the pool

➤ **OrbThreadPool**

These are resources related to the ORB thread pool that is on the server.

Measurement	Description
ActiveThreads	The average number of active threads in the pool
TotalThreads	The average number of threads in the pool
PercentTimeMaxed	The average percent of the time that the number of threads in the pool reached or exceeded the desired maximum number
ThreadCreates	The number of threads created
ThreadDestroys	The number of threads destroyed
ConfiguredMaxSize	The configured maximum number of pooled threads

➤ **DBConnectionMgr**

These are resources related to the database connection manager. The manager consists of a series of data sources, as well as a top-level aggregate of each of the performance metrics.

Measurement	Description
DataSource	Resources related to a specific data source specified by the "name" attribute
ConnectionCreates	The number of connections created
ConnectionDestroys	The number of connections released
ConnectionPoolSize	The average size of the pool, i.e., number of connections
ConnectionAllocates	The number of times a connection was allocated
ConnectionWaiters	The average number of threads waiting for a connection
ConnectionWaitTime	The average time, in seconds, of a connection grant
ConnectionTime	The average time, in seconds, that a connection is in use
ConnectionPercentUsed	The average percentage of the pool that is in use
ConnectionPercentMaxed	The percentage of the time that all connections are in use

➤ **TransactionData**

These are resources that pertain to transactions.

Measurement	Description
NumTransactions	The number of transactions processed
ActiveTransactions	The average number of active transactions
TransactionRT	The average duration of each transaction
BeanObjectCount	The average number of bean object pools involved in a transaction
RolledBack	The number of transactions rolled back
Commited	The number of transactions committed
LocalTransactions	The number of transactions that were local
TransactionMethodCount	The average number of methods invoked as part of each transaction
Timeouts	The number of transactions that timed out due to inactivity timeouts
TransactionSuspended	The average number of times that a transaction was suspended

➤ **ServletEngine**

These are resources that are related to servlets and JSPs.

Measurement	Description
ServletsLoaded	The number of servlets currently loaded
ServletRequests	The number of requests serviced
CurrentRequests	The number of requests currently being serviced
ServletRT	The average response time for each request
ServletsActive	The average number of servlets actively processing requests

Measurement	Description
ServletIdle	The amount of time that the server has been idle (i.e., time since last request)
ServletErrors	The number of requests that resulted in an error or an exception
ServletBeanCalls	The number of bean method invocations that were made by the servlet
ServletBeanCreates	The number of bean references that were made by the servlet
ServletDBCalls	The number of database calls made by the servlet
ServletDBConAlloc	The number of database connections allocated by the servlet
SessionLoads	The number of times the servlet session data was read from the database
SessionStores	The number of times the servlet session data was stored in the database
SessionSize	The average size, in bytes, of a session data
LoadedSince	The time that has passed since the server was loaded (UNC time)

➤ **Sessions**

These are general metrics regarding the HTTP session pool.

Measurement	Description
SessionsCreated	The number of sessions created on the server
SessionsActive	The number of currently active sessions
SessionsInvalidated	The number of invalidated sessions. May not be valid when using sessions in the database mode
SessionLifetime	Contains statistical data of sessions that have been invalidated. Does not include sessions that are still alive

6 Click **OK** in the WebSphere Monitor Configuration dialog box, and in the WebSphere dialog box, to activate the WebSphere monitor.

Note: The port you use to monitor a WebSphere server through a firewall depends on the configuration of your server.

19

Network Monitoring

You can use Network monitoring to determine whether your network is causing a delay in the scenario. You can also determine the problematic network segment.

This chapter describes:

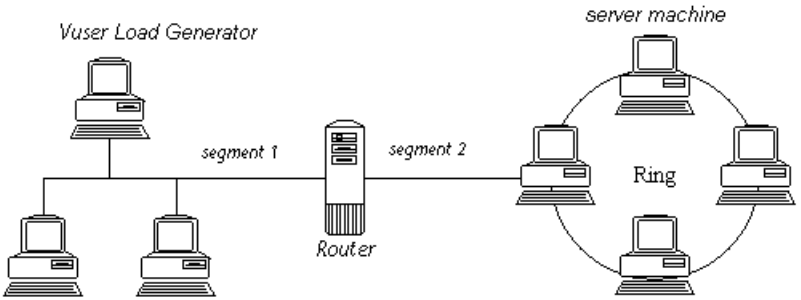
- Configuring the Network Monitor
- Viewing the Network Delay Time Graph

About Network Monitoring

Network configuration is a primary factor in the performance of applications. A poorly designed network can slow client activity to unacceptable levels.

In a true Web or client/server system, there are many network segments. A single network segment with poor performance can affect the entire system.

The following diagram shows a typical network. In order to go from the server machine to the Vuser machine, data must travel over several segments.



To measure network performance, the Network monitor sends packets of data across the network. When a packet returns, the monitor calculates the time it takes for the packet to go to the requested node and return. This time is the delay which appears in the Network Delay Time graph.

Using the online Network Delay Time graph, you can locate the network-related problem so that it can be fixed.

Note: The delays from the source machine to each of the nodes are measured concurrently, yet independently. It is therefore possible that the delay from the source machine to one of the nodes could be greater than the delay for the complete path between the source and destination machines.

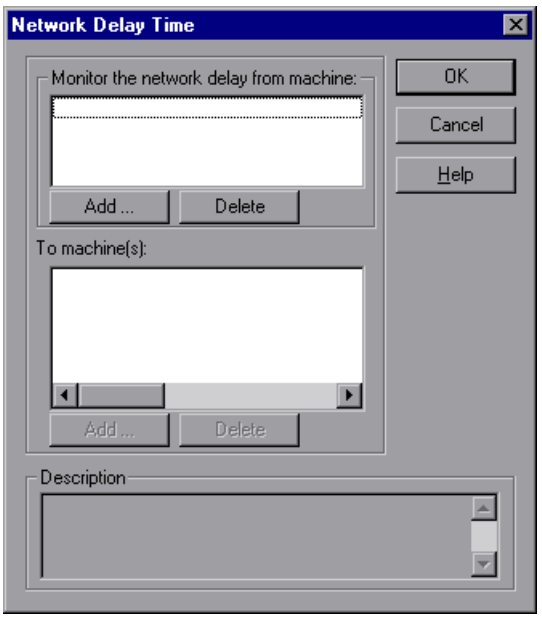
Configuring the Network Monitor

You configure the Network monitor from the Run view of the Controller before you begin running a scenario. Using the Network Delay Time and Add Destination Machines for Network Delay Monitoring dialog boxes, you select the network path you want to monitor.

Note: To enable network monitoring, you must install the LoadRunner Agent on the source machine. You do not have to install the LoadRunner Agent on the destination machine.

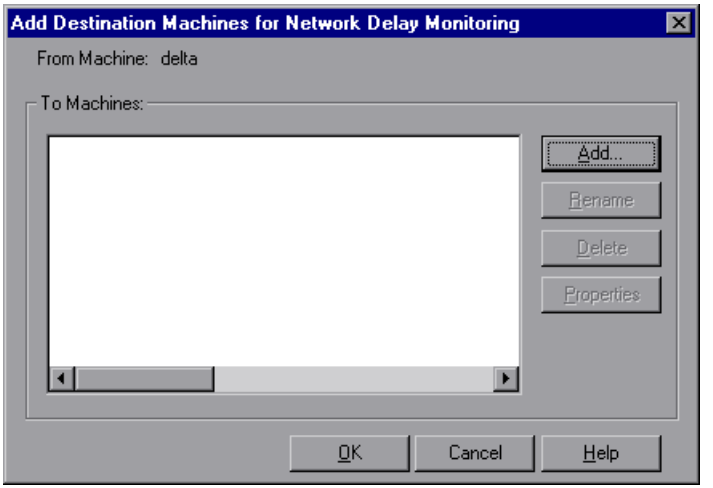
To configure the Network Monitor:

1. In the graph tree view, select the **Network Delay Time** graph and drag it into the right pane.
2. Right-click the graph and choose **Add Measurement(s)**, or choose **Monitors > Add Online Measurement**. The Network Delay Time dialog box opens.



3 In the **Monitor the network delay from machine** section, click **Add** to enter the server name or IP address of the source machine, from which you want the network path monitoring to begin. Select the platform on which the machine runs, and click **OK**.

4 In the **To machine(s)** section of the Network Delay Time dialog box, click **Add** to enter the name of the machine at the final destination of the path you want to monitor. The Add Destination Machines for Network Delay Monitoring dialog box opens.

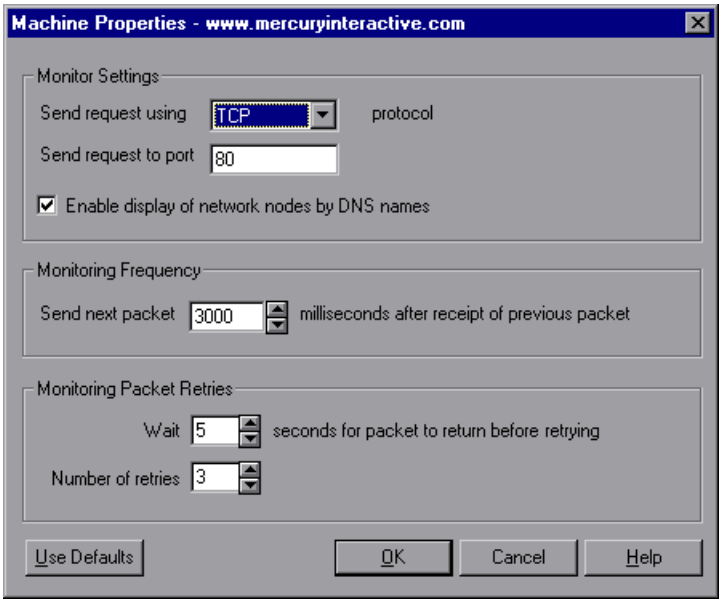


5 Click **Add**, enter the name of the destination machine, and click **OK**. The name of the machine appears in the Add Destination Machines for Network Delay Monitoring dialog box. Repeat this procedure for each path you want to monitor.

To rename a machine, click **Rename**, and enter a new name for the machine.

To delete a machine, select it and click **Delete**.

6 Click **Properties** to configure additional network monitor settings. The Machine Properties - <Machine Name> dialog box opens.



7 In the Monitor Settings box, select the protocol and enter the port number being used by the network path. Select **Enable display of network nodes by DNS names** if you want to view the DNS name of each node along the network path, in addition to its IP address.

8 In the Monitoring Frequency box, select the number of milliseconds the monitor should wait between receiving a packet and sending out the next packet. The default value is 3000 milliseconds. If you have a long, steady scenario, you can increase the interval by several seconds.

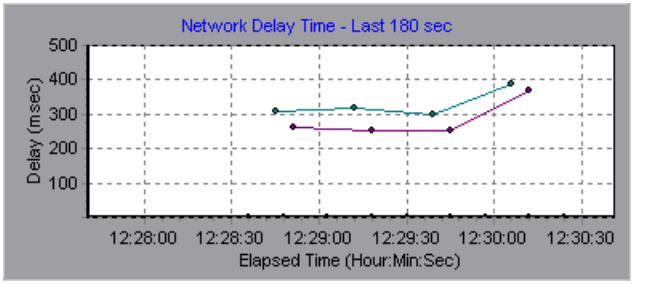
9 In the Monitoring Packet Retries box, select the maximum number of seconds that the monitor should wait for a packet to return before it retries to send the packet. The default value is 5 seconds. If your network is very large and loaded (an internet connection with a low capacity), you should increase the value by several seconds. If you have a small network (such as a LAN), you can decrease the value.

In addition, select the number of times the Network monitor should try resending a packet to a node if the packet is not initially returned. The default value is 3.

Viewing the Network Delay Time Graph

The **Network Delay Time** graph shows the delay for the complete path between the source and destination machines (y-axis) as a function of the elapsed scenario time (x-axis).

Each path defined in the Add Destination Machines for Network Delay Monitoring dialog box is represented by a separate line with a different color in the graph.

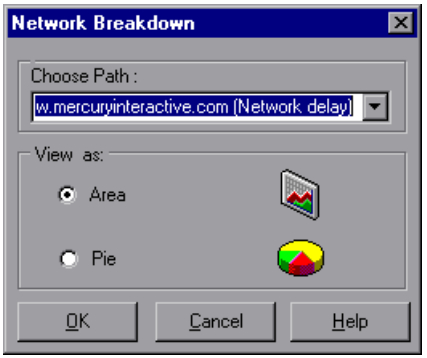


To view the DNS names of the measurements displayed in the legend, right-click the graph and select **View as DNS Name**.

In addition, you can view the delay from the source machine to each of the nodes along the network path.

To view the delay time for the network segments:

1 Right-click the Network Delay Time graph, and select **View Segments**. The Network Breakdown dialog box opens.



2 Select the path that you want to break down.

3 Choose whether you want to view the network segments of the graph of the graph you chose as an area graph or a pie graph.

4 Click **OK** to close the Network Breakdown dialog box. The delay time for the network segments of the path you chose is displayed in the graph view area.

To return to the complete path delay time view, select **Hide Segments** from the right-click menu.

20

Streaming Media Monitoring

During a scenario run, you can monitor the Windows Media Server and RealPlayer audio/video servers, as well as the RealPlayer client, in order to isolate server and client performance bottlenecks.

This chapter describes:

- Configuring the Windows Media Server Monitor
- Configuring the RealPlayer Server Monitor
- Viewing the RealPlayer Client Online Graph
- Viewing the Media Player Client Online Graph

Note: For instructions on recording a script containing streaming media functions, see the *Creating Vuser Scripts* guide.

Configuring the Windows Media Server Monitor

To monitor the Windows Media Server, you must first select the counters you want the Windows Media Server monitor to measure. You select these counters using the Windows Media Server dialog box.

To configure the Windows Media Server monitor:

1. Click the Windows Media Server graph in the graph tree, and drag it into the right pane of the Run view.
2. Right-click the graph and choose **Add Measurement(s)**, or choose **Monitors** > **Add Online Measurement**.

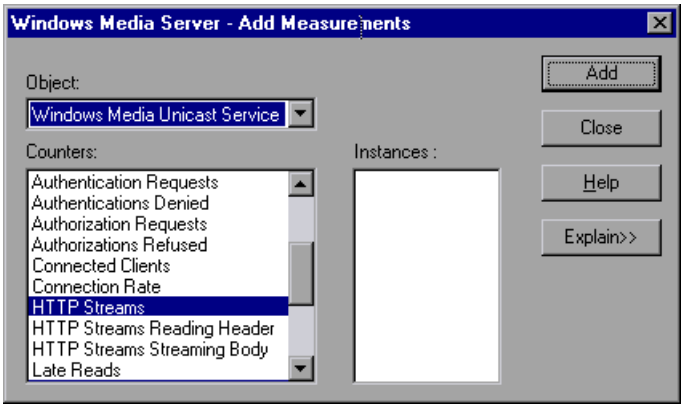
3 In the Monitored Server Machines section of the Windows Media Server dialog box, click **Add** to enter the server name or IP address of the machine you want to monitor. Select the platform on which the machine runs, and click **OK**.

4 In the Resource Measurements section of the Windows Media Server dialog box, select the measurements you want to monitor. The following table describes the default counters that can be monitored:

Measurement	Description
Active Live Unicast Streams (Windows)	The number of live unicast streams that are being streamed
Active Streams	The number of streams that are being streamed
Active TCP Streams	The number of TCP streams that are being streamed
Active UDP Streams	The number of UDP streams that are being streamed
Aggregate Read Rate	The total, aggregate rate (bytes/sec) of file reads
Aggregate Send Rate	The total, aggregate rate (bytes/sec) of stream transmission
Connected Clients	The number of clients connected to the server
Connection Rate	The rate at which clients are connecting to the server
Controllers	The number of controllers currently connected to the server
HTTP Streams	The number of HTTP streams being streamed
Late Reads	The number of late read completions per second
Pending Connections	The number of clients that are attempting to connect to the server, but are not yet connected. This number may be high if the server is running near maximum capacity and cannot process a large number of connection requests in a timely manner.
Stations	The number of station objects that currently exist on the server

Measurement	Description
Streams	The number of stream objects that currently exist on the server
Stream Errors	The cumulative number of errors occurring per second

5 To select additional measurements, click **Add.** The Windows Media Server - Add Measurements dialog box opens, displaying the Windows Media Unicast Service object, its counters, and instances.



6 Select a counter and an instance. You can select multiple counters using the **Ctrl** key. The instance is relevant only if multiple instances of the highlighted counter are running. For a description of each counter, click **Explain>>** to expand the dialog box.

7 Click **Add** to place the selected counter on the resource list. Add all the desired resources to the list, and click **Close**.

8 Click **OK** in the Windows Media Server dialog box to activate the monitor.

Configuring the RealPlayer Server Monitor

To monitor the RealPlayer Server, you must first select the counters you want the RealPlayer Server monitor to measure. You select these counters using the Real Server dialog box.

To configure the RealPlayer Server monitor:

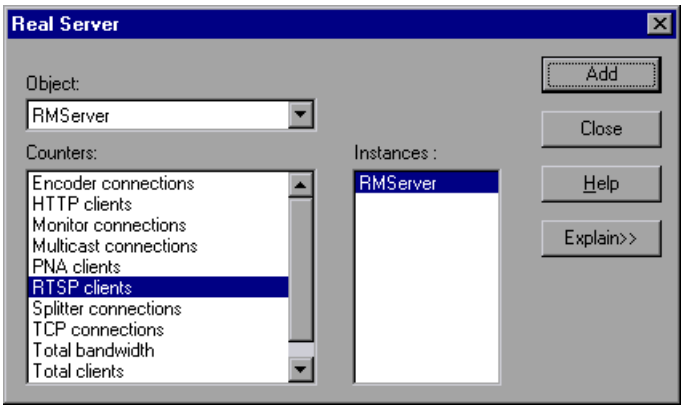
1 Click the Real Server graph in the graph tree, and drag it into the right pane of the Run view.

2 Right-click the graph and choose **Add Measurement(s)**, or choose **Monitors** > **Add Online Measurement**.

3 In the Monitored Server Machines section of the Real Server dialog box, click **Add** to enter the server name or IP address of the machine you want to monitor. Select the platform on which the machine runs, and click **OK**.

4 Click **Add** in the Resource Measurements section of the Real Server dialog box to select the measurements that you want to monitor.

Another Real Server dialog box opens, displaying the counters that can be monitored.



5 Select a counter and an instance. You can select multiple counters using the **Ctrl** key. The instance is relevant only if multiple instances of the highlighted counter are running. For a description of each counter, click **Explain**>> to expand the dialog box.

The following table describes the default counters that can be monitored:

Measurement	Description
Encoder Connections	The number of active encoder connections
HTTP Clients	The number of active clients using HTTP
Monitor Connections	The number of active server monitor connections
Multicast Connections	The number of active multicast connections
PNA Clients	The number of active clients using PNA
RTSP Clients	The number of active clients using RTSP
Splitter Connections	The number of active splitter connections
TCP Connections	The number of active TCP connections
Total Bandwidth	The number of bits per second being consumed
Total Clients	The total number of active clients
UDP Clients	The number of active UDP connections

6 Click **Add** to place the selected counter on the resource list. Add all the desired resources to the list, and click **Close**.

7 Click **OK** in the Real Server dialog box to activate the monitor.

Viewing the RealPlayer Client Online Graph

You can view the RealPlayer Client online monitor graph by dragging it from the graph tree into the right pane of the Run view.

The following table describes the RealPlayer Client measurements that are monitored:

Measurement	Description
Current Bandwidth (Kbits/sec)	The number of kilobytes in the last second
Buffering Event Time (sec)	The average time spent on buffering
Network Performance	The ratio (percentage) between the current bandwidth and the actual bandwidth of the clip
Percentage of Recovered Packets	The percentage of error packets that were recovered
Percentage of Lost Packets	The percentage of packets that were lost
Percentage of Late Packets	The percentage of late packets
Time to First Frame Appearance (sec)	The time for first frame appearance (measured from the start of the replay)
Number of Buffering Events	The average number of all buffering events
Number of Buffering Seek Events	The average number of buffering events resulting from a seek operation
Buffering Seek Time	The average time spent on buffering events resulting from a seek operation
Number of Buffering Congestion Events	The average number of buffering events resulting from network congestion
Buffering Congestion Time	The average time spent on buffering events resulting from network congestion

Measurement	Description
Number of Buffering Live Pause Events	The average number of buffering events resulting from live pause
Buffering Live Pause Time	The average time spent on buffering events resulting from live pause

Viewing the Media Player Client Online Graph

You can view the Windows Media Player Client online monitor graph by dragging it from the graph tree into the right pane of the Run view.

The following table describes the Media Player Client measurements that are monitored:

Measurement	Description
Stream Quality (Packet-level)	The percentage ratio of packets received to total packets
Current bandwidth (Kbits/sec)	The number of kbits per second received
Stream Packet Rate	The number of packets received
Total number of recovered packets	The number of lost packets that were recovered. This value is only relevant during network playback.
Total number of lost packets	The number of lost packets that were not recovered. This value is only relevant during network playback.
Stream Quality (Sampling-level)	The percentage of stream samples received on time (no delays in reception)

21

FireWall Server Performance Monitoring

During a scenario run, you can monitor the FireWall server in order to isolate server performance bottlenecks.

This chapter describes:

➤ Configuring the Check Point FireWall-1 Server Monitor

Configuring the Check Point FireWall-1 Server Monitor

To monitor the Check Point FireWall-1 server, you must select the counters you want the Check Point FireWall-1 server monitor to measure. You select these counters using the Check Point FireWall-1 SNMP Resources dialog box.

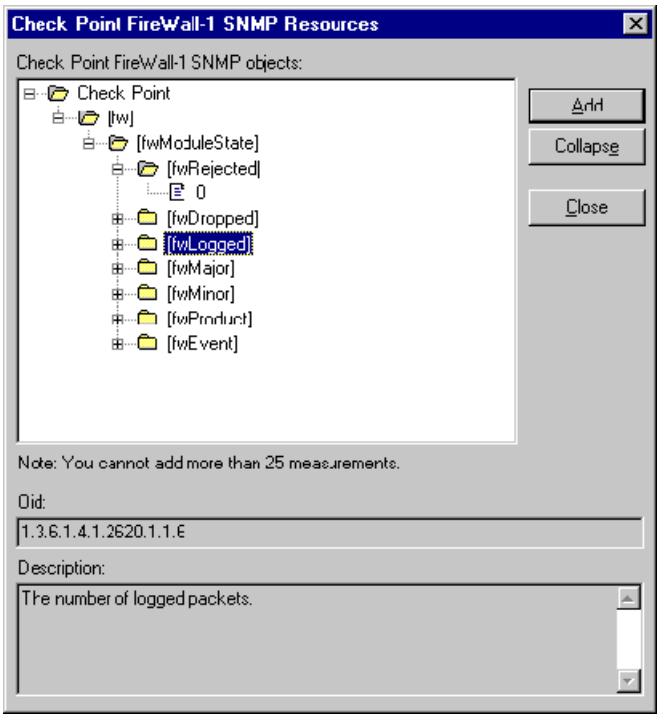
To configure the Check Point FireWall-1 server monitor:

1. Click the Check Point FireWall-1 graph in the graph tree, and drag it into the right pane of the Run view.
2. Right-click the graph in the graph and choose **Add Measurement(s)**, or choose **Monitors > Add Online Measurement**.
3. In the Monitored Server Machines section of the Check Point FireWall-1 dialog box, click **Add** to enter the server name or IP address of the machine you want to monitor. Select the platform on which the machine runs, and click **OK**.

Note: You can specify a port number in the snmp.cfg file. If you do not specify a port number, LoadRunner connects to port 260, the default port for the Check Point FireWall-1 SNMP agent. You can also specify a machine name and port number in the Add Machine dialog box using the following format:
<machine name>:<port number>

---

4 Click **Add** in the Resource Measurements section of the Check Point FireWall-1 dialog box. The Check Point FireWall-1 SNMP Resources dialog box opens.



5 Select the measurements you want to monitor. The following default counters can be monitored:

Measurement	Description
fwRejected	The number of rejected packets.
fwDropped	The number of dropped packets.
fwLogged	The number of logged packets.

6 Click **Add** to place the selected counter on the resource list. Add all the desired resources to the list, and click **Close**.

Note: The Check Point FireWall-1 monitor can only monitor up to 25 measurements.

---

7 Click **OK** in the Check Point FireWall-1 dialog box to activate the monitor.

22

ERP Server Resource Monitoring

During a scenario run, you can monitor ERP server resources in order to isolate server performance bottlenecks.

This chapter describes:

➤ Configuring the SAP Monitor

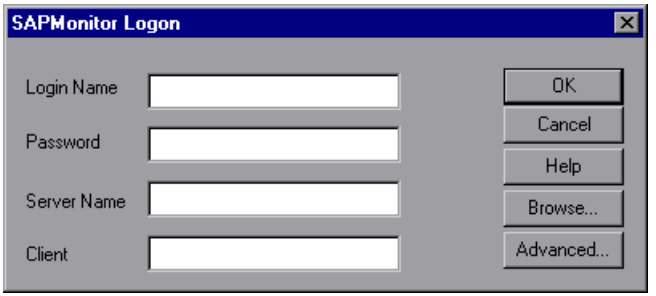
Configuring the SAP Monitor

To monitor a SAP R/3 system server, you must first select the counters you want the SAP monitor to measure. You select these counters using the Add SAP Monitor Measurements dialog box.

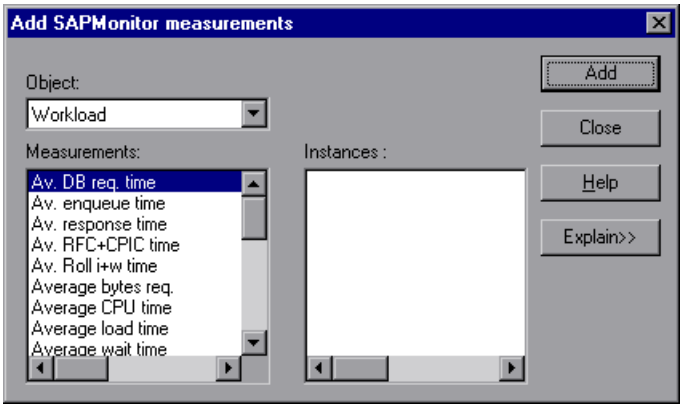
To configure the SAP monitor:

1. Click the SAP graph in the graph tree, and drag it into the right pane of the Run view.
2. Right-click the graph in the graph and choose **Add Measurement(s)**, or choose **Monitors > Add Online Measurement**.
3. In the Monitored Server Machines section of the SAP dialog box, click **Add** to enter the server name or IP address of the machine you want to monitor. Select the platform on which the machine runs, and click **OK**.

4 Click **Add** in the Resource Measurements section of the SAP dialog box. The SAP Monitor Logon dialog box opens.



5 Enter your Login Name, Password, Server Name, and Client, and click **OK**. The Add SAP Monitor Measurements dialog box opens.



6 Select an object, a measurement, and an instance. You can select multiple measurements using the **Ctrl** key. The instance is relevant only if multiple instances of the highlighted measurement are running. For a description of each measurement, click **Explain>>** to expand the dialog box.

The following are the most commonly monitored counters:

Measurement	Description
Average CPU time	The average CPU time used in the work process.
Average response time	The average response time, measured from the time a dialog sends a request to the dispatcher work process, through the processing of the dialog, until the dialog is completed and the data is passed to the presentation layer. The response time between the SAP GUI and the dispatcher is not included in this value.
Average wait time	The average amount of time that an unprocessed dialog step waits in the dispatcher queue for a free work process. Under normal conditions, the dispatcher work process should pass a dialog step to the application process immediately after receiving the request from the dialog step. Under these conditions, the average wait time would be a few milliseconds. A heavy load on the application server or on the entire system causes queues at the dispatcher queue.
Average load time	The time needed to load and generate objects, such as ABAP source code and screen information, from the database.
Database calls	The number of parsed requests sent to the database.
Database requests	The number of logical ABAP requests for data in the database. These requests are passed through the R/3 database interface and parsed into individual database calls. The proportion of database calls to database requests is important. If access to information in a table is buffered in the SAP buffers, database calls to the database server are not required. Therefore, the ratio of calls/requests gives an overall indication of the efficiency of table buffering. A good ratio would be 1:10.

Measurement	Description
GUI time	The GUI time is measured in the work process and is the response time between the dispatcher and the GUI.
Roll ins	The number of rolled-in user contexts.
Roll outs	The number of rolled-out user contexts.
Roll in time	The processing time for roll ins.
Roll out time	The processing time for roll outs.
Roll wait time	The queue time in the roll area. When synchronous RFCs are called, the work process executes a roll out and may have to wait for the end of the RFC in the roll area, even if the dialog step is not yet completed. In the roll area, RFC server programs can also wait for other RFCs sent to them.
Average time per logical DB call	The average response time for all commands sent to the database system (in milliseconds). The time depends on the CPU capacity of the database server, the network, the buffering, and on the input/output capabilities of the database server. Access times for buffered tables are many magnitudes faster and are not considered in the measurement.

7 Click **Add** to place the selected measurement on the resource list. Add all the desired resources to the list, and click **Close**.

8 Click **OK** in the SAP dialog box to activate the monitor.

23

Preparing for Monitoring over the Firewall

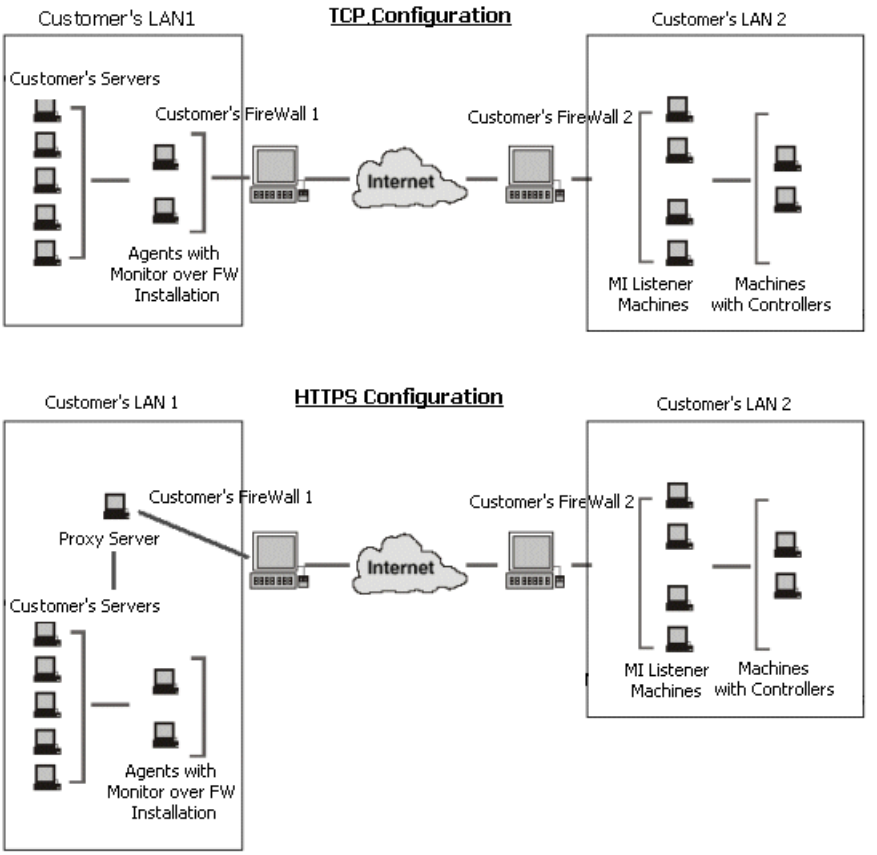
To enable monitoring of your servers from outside the firewall, *Monitors over Firewall* is installed on designated machines inside the firewall. The installation sets up the Server Monitor agent (referred to as the "agent" in this chapter) as well as the Server Monitor configuration tool. You then configure the servers to monitor, and define the specific measurements that LoadRunner collects for each monitored server.

This chapter describes:

- Overview of Monitoring over the Firewall
- Installing Monitors over Firewall
- Installing MI_Listener
- Configuring the Agent to Operate over the Firewall
- Modifying Firewall Settings
- Preparing for Data Collection
- Configuring Server Monitor Properties
- Adding and Removing Measurements
- Configuring Measurement Frequency

About Monitoring over the Firewall

Monitors over Firewall is installed on one or more designated machines inside Firewall 1, using one of the following configurations:



During installation, the LoadRunner Agent is added either as a Windows service or as an executable run from the Startup folder.

Note: *Monitors over Firewall* may have been installed during LoadRunner installation. See "Installing Monitors over Firewall" on page 298 for more information.

After installation, you configure the LoadRunner agent to operate over the firewall. You also modify firewall settings to enable communication between the Server Monitor agent machine(s) inside the firewall and machines outside the firewall. In addition, you prepare the Controller for data collection.

You then need to configure the server monitors. You configure server monitor properties (select the server whose resources you want to monitor, the host machine whose agent will collect the data, and the server monitors you want to run), add the measurements to monitor for each server, and specify the frequency with which you want the monitored measurements to be reported.

Overview of Monitoring over the Firewall

To prepare for monitoring over the firewall, perform the following steps:

1 **Make sure *Monitors over Firewall* is installed.**

Refer to "Installing Monitors over Firewall," on page 298.

2 **Install MI_Listener on one or more machines inside your LAN2.**

For instructions, refer to the *LoadRunner Installation Guide*.

3 **Configure the agent to operate over the firewall.**

Refer to "Configuring the Agent to Operate over the Firewall," on page 306.

4 **Modify firewall settings.**

Modify your firewall settings to enable communication between the Server Monitor agent machine(s) inside the firewall and machines outside the firewall. Refer to "Modifying Firewall Settings," on page 308.

5 **Prepare for data collection.**

Check that you can obtain information for the monitors configured inside the firewall. Refer to "Preparing for Data Collection," on page 309.

6 **Configure server monitor properties.**

Refer to "Configuring Server Monitor Properties," on page 310.

7 **Add and remove measurements.**

Add measurements to monitor for each server. If LoadRunner added default measurements, you can edit them as required. Refer to "Adding and Removing Measurements," on page 312.

8 **Configure measurement frequencies.**

Set a measurement schedule for each measurement to be reported. Refer to "Configuring Measurement Frequency," on page 313.

Installing Monitors over Firewall

Monitors over Firewall may have been installed during LoadRunner installation. To check whether it was installed, click **Start** > **Programs** > **LoadRunner**. If the **Monitor Configuration** option appears on the list of LoadRunner options, then *Monitors over Firewall* was already installed, and you can proceed to "Installing MI_Listener" on page 305.

If Monitors over Firewall was not yet installed, you need to install it using one of the following:

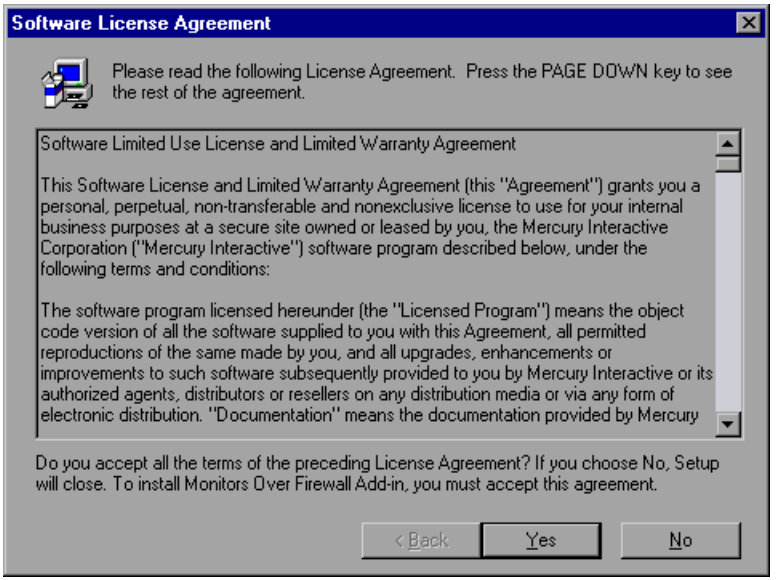
➤ Perform a custom installation of LoadRunner from the LoadRunner CD, choosing only the Monitors over Firewall option. For instructions on performing a custom installation of LoadRunner, refer to the *LoadRunner Installation Guide*.

➤ Obtain the *Monitors over Firewall* file from the Mercury Interactive Customer Support Web site (support.mercuryinteractive.com). *Monitors over Firewall* is a standalone downloadable installation. It comes as a self-extracting installer file.

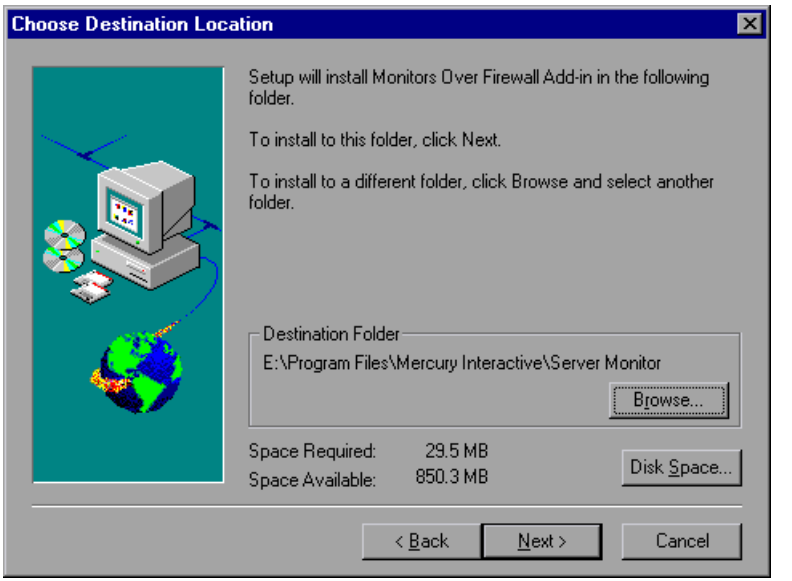
To install Monitors over Firewall from the Mercury Interactive Customer Support Web site:

1 Copy the self-extracting installer file to each agent machine.

2 On each agent machine, double-click the installer file to begin installation. The software license agreement appears. Read the Agreement, and click **Yes** to accept it. If you click **No**, Setup closes.

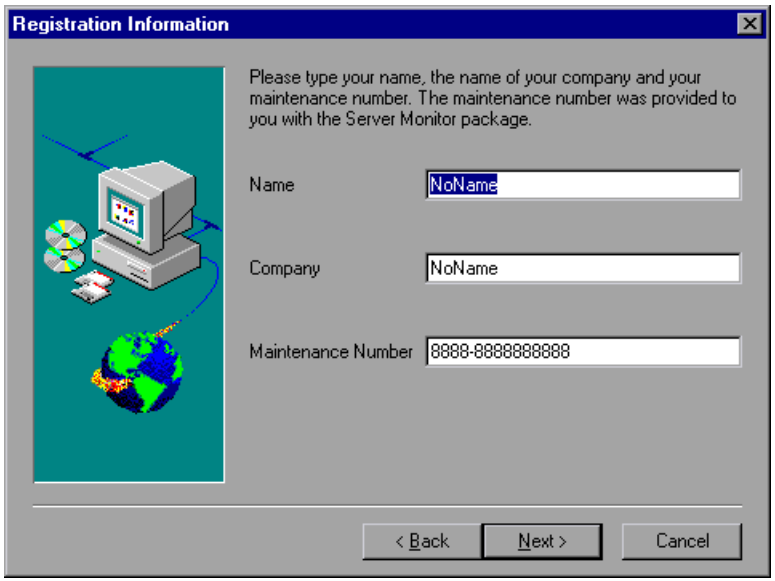


3 In the Choose Destination Location screen, specify the folder in which to install the add-in. To select a different location, click **Browse**, choose a folder, and click **OK.**



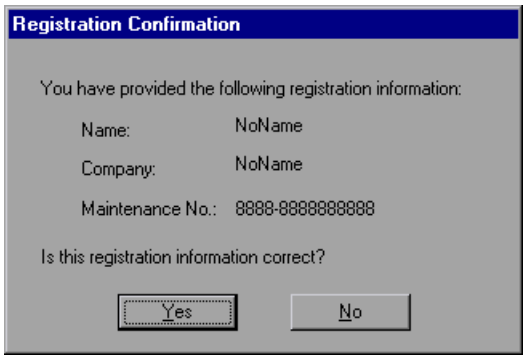
Click **Next**.

4 In the Registration Information screen, type your name, the name of your company, and your maintenance number, which you received with your LoadRunner Server Monitors package.

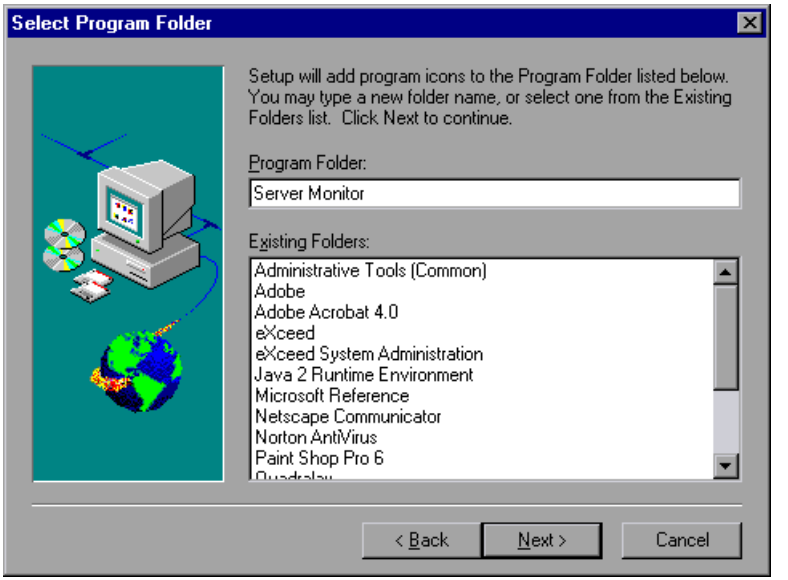


Click **Next**.

5 Confirm that the registration information is correct, and click **Yes**. If the registration information is incorrect, click **No** to go back to the Registration Information screen, and make your corrections.

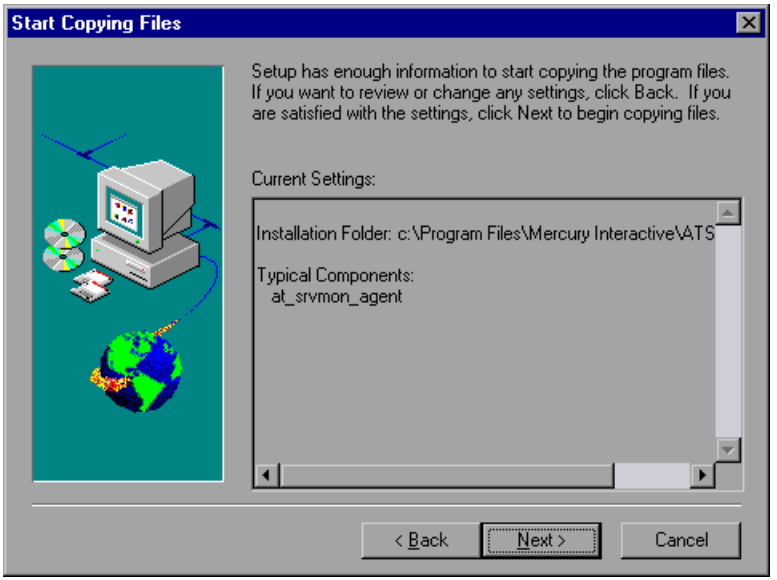


6 In the Select Program Folder screen, specify a program folder, or accept the default folder, *Server Monitor*.



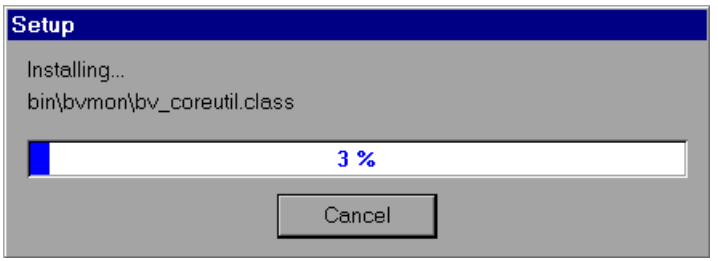
Click **Next**.

7 In the Start Copying Files screen, review your settings. To make changes, click **Back**.



Click **Next**.

8 The installation process begins. To pause or quit the installation, click **Cancel**.



9 In the User Login Settings screen, select the login method. Choose from:

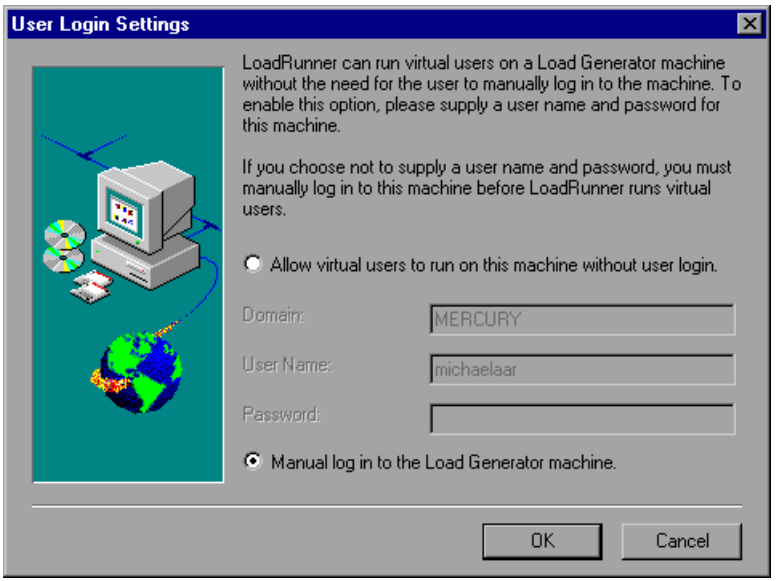
➤ **Allow virtual users to run on this machine without user login.**

This setting instructs Setup to install the Server Monitor agent as a Windows service. When the service starts (for example, on machine startup), the agent uses the specified login parameters.

Enter the network domain where the Server Monitor machine resides, a user name, and password. Note that you must enter the name and password of a user who has administrator privileges on the server machines you plan to monitor.

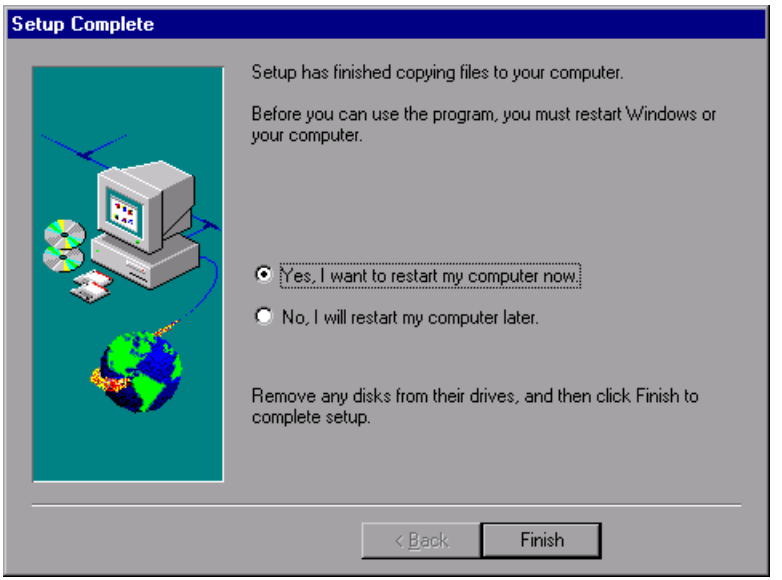
➤ **Manual log in to the Load Generator machine. (recommended)**

This setting instructs Setup to install the Server Monitor agent as an executable run from the Startup folder. You must manually log into the Server Monitor machine before a LoadRunner load test run. Note that you must log in using the name and password of a user who has administrator privileges on the server machines you plan to monitor.



Click **OK**.

10 Setup completes the installation process. The Setup Complete screen prompts you to restart your computer. You can delay restarting your computer until a later point, however, you must restart your computer before you use LoadRunner Server Monitors.



Click **Finish** to complete the setup process.

Installing MI_Listener

To enable monitoring over a firewall, you need to install MI Listener on one or more machines in your LAN2. For instructions, refer to the *LoadRunner Installation Guide*.

Configuring the Agent to Operate over the Firewall

After installation, you configure the Server Monitor agent to operate over your firewall.

To configure the agent to operate over the firewall:

1. Stop the LoadRunner Agent by right-clicking its icon in the system tray and selecting **Close**.
2. Open **<Server Monitor root folder>\launch_service\ dat\br_lnch_server.cfg** in a text editor.
3. In the Firewall section, set each option as follows:

Option	Setting	Description
FireWallServiceActive	1	Enables firewall service.
HostSymbol	<host_alias_name>	A unique host identifier used to establish a unique connection between the Controller host and the agent machine, via the Mercury Interactive listener machine, MI Listener. Note: On the Controller, you will add a host named <host_alias_name>. See "Preparing for Data Collection," on page 309.
RouterMachine	<MI Listener>	The IP address of the Mercury Interactive listener machine, MI Listener.
ConnectMode	1	Enables connect mode.

Option	Setting	Description
TIMEOUT	20000	The length of time you want the agent to wait before retrying to connect to the RouterMachine. TIMEOUT is measured in milliseconds. Use 20000 as a default.
ChannelType	<config_type>	Enter either tcp (default) or http, depending on the configuration you are using.

4 Save your changes.

5 If you set ChannelType to http, edit the **<Server Monitor root folder>\launch_service\ dat\channel_configure.dat** file, which enables you to connect to the proxy server, as follows:

Option	Description
proxy_name OR proxy_full_name OR proxy_ip	The network address of the proxy server. Enter at least one of these settings.
proxy_port	Enter the proxy server connection port.
proxy_username	Enter the username of a user with connection rights to the proxy server.
proxy_password	Enter the user's password.
proxy_domain	Enter the user's domain if defined in proxy server configuration.

6 Save your changes.

7 Restart the LoadRunner Agent:

➤ If the LoadRunner Agent is installed as a service, restart the service.

➤ If the LoadRunner Agent is installed as an executable in the Startup folder, select it from the Startup folder.

Modifying Firewall Settings

You modify your firewall settings to enable communication between the Server Monitor agent machine(s) inside the firewall and machines outside the firewall.

To enable this connection, you allow an outgoing connection for HTTPS service on the firewall for port 443. On the MI Listener machine, you open incoming HTTPS service for port 443.

TCP Configuration

In this configuration, the LoadRunner Agent tries to establish a connection with MI Listener using port 443 every 20000 milliseconds (or whatever amount of time you specified as the TIMEOUT value in the br_lnch_server.cfg file).

As a result, the LoadRunner Agent connects to MI Listener and MI Listener connects back to the LoadRunner Agent. From this point on, the LoadRunner Agent listens to commands from MI Listener.

HTTPS Configuration

In this configuration, the LoadRunner Agent tries to establish a connection with the proxy server using the port defined in channel_configure.dat every 20000 milliseconds (or whatever amount of time you specified as the TIMEOUT value in the br_lnch_server.cfg file). On successful connection, the proxy server connects to MI Listener.

As a result, the proxy server connects to MI Listener and MI Listener connects back to the LoadRunner Agent through the proxy server. From this point on, the LoadRunner Agent listens to commands from MI Listener.

Preparing for Data Collection

You need to make sure that you can obtain information for the monitors configured inside the firewall.

To configure the Controller for data collection:

1 On the Controller, add a host named <**host_alias_name**>.

Note: The name of the host is used to establish a unique connection between the Controller host and the agent machine, via the Mercury Interactive listener machine, MI Listener. See the definition of<host_alias_name> in "Configuring the Agent to Operate over the Firewall," on page 306.

---

2 In the FireWall tab of the Load Generator Information dialog box, enter the IP address of the MI Listener machine, and check **Enable Monitoring Over FireWall**.

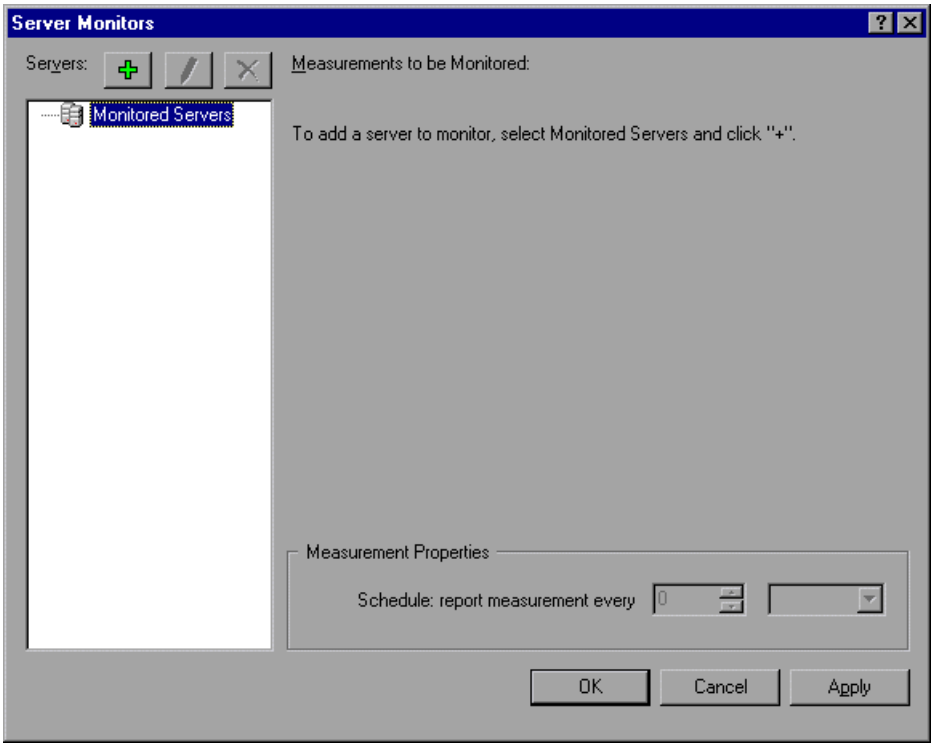
3 Connect to the host. Make sure that you obtain information for the monitors configured inside the firewall.

Configuring Server Monitor Properties

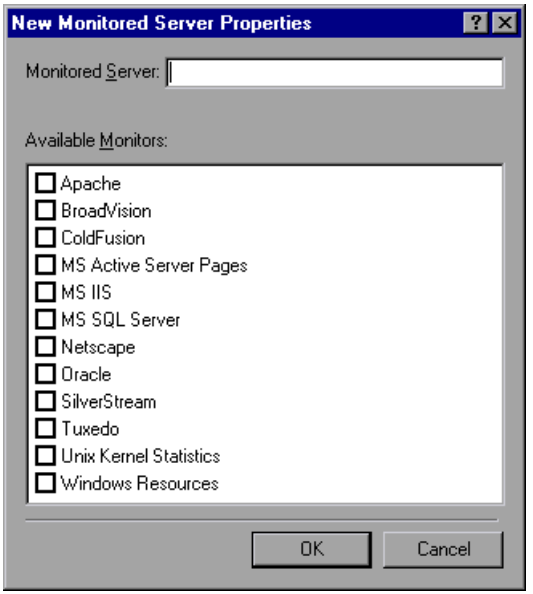
To enable monitoring over the firewall, you need to configure server monitor properties.

To configure server monitor properties:

1 Select **Start > Programs > LoadRunner > Monitor Configuration**. The Server Monitors dialog box opens.



2 Click the **Add Server** button. The New Monitored Server Properties dialog box opens.



3 In the Monitored Server box, type the name or IP address of the server whose resources you want to monitor.

Note: To add several servers simultaneously, separate the server names or IP ranges with commas. For example: 255.255.255.0-255.255.255.5, server1, server2.

---

4 From the Available Monitors list, select the monitors appropriate for the server being monitored.

Note: Data can only be viewed for the monitors that are enabled with your LoadRunner license key. To preview your license key information, in the LoadRunner Controller, select **Help > About LoadRunner**.

---

5 Click **OK** to close the New Monitored Server Properties dialog box. LoadRunner displays the settings you selected in the Monitored Servers list.

Note that, for certain monitors, LoadRunner displays default measurements in the right pane. For details on selecting measurements, see "Adding and Removing Measurements" on page 312.

6 To add additional monitored servers to the list, repeat steps 1-5.

7 Click **Apply** to save your settings.

Adding and Removing Measurements

After you configure one or more server machines to monitor, you add measurements to monitor for each server. If LoadRunner added default measurements, you can edit them as required.

To add a measurement to monitor:

1 Select a server from the Monitored Servers list.

2 Click the **Add Measurement** button. Select the appropriate monitor. A dialog box opens, enabling you to choose measurements for the monitor you selected.

3 Select the measurements that you want to monitor, and click **OK**.

4 Click **Apply** to save your settings.

For information on configuring measurements for each server monitor, see the relevant chapter. For the Windows Resources, UNIX, SNMP, TUXEDO, SQL, and Oracle server monitors, see Chapter 16, "System and Database Resource Monitoring." For the Apache, MS IIS, and Netscape server monitors, see Chapter 17, "Web Resource and Web Server Resource Monitors." For the BroadVision, ColdFusion, ATG Dynamo, Fujitsu

INTERSTAGE, MS ASP, SilverStream, WebLogic, and WebSphere server monitors, see Chapter 18, “Web Application Server Resource Monitors.” For the Windows Media and RealPlayer server monitors, see Chapter 20, “Streaming Media Monitoring.” For the CheckPoint FireWall -1 server monitor, see Chapter 21, “FireWall Server Performance Monitoring.”

To remove a measurement from the measurements list:

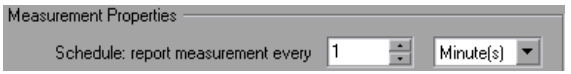
1 Select the measurement, and click the **Delete** button.

2 Click **Apply** to save your settings.

Configuring Measurement Frequency

Once you have configured monitor measurements, you configure measurement frequency.

In the Measurement Properties section, you set a measurement schedule for each measurement to be reported.



To set a measurement schedule for a measurement:

1 Select the configured server measurement you want to schedule.

2 Specify the frequency at which you want LoadRunner to report the measurement.

3 Click **Apply** to save your settings.

Part V

Appendixes

A

Interpreting LoadRunner Online Graphs

LoadRunner online monitor graphs present important information about the performance of your scenario. This chapter describes some of the key online graphs in greater depth and shows how they can be used to identify and pinpoint performance bottlenecks as your scenario is running.

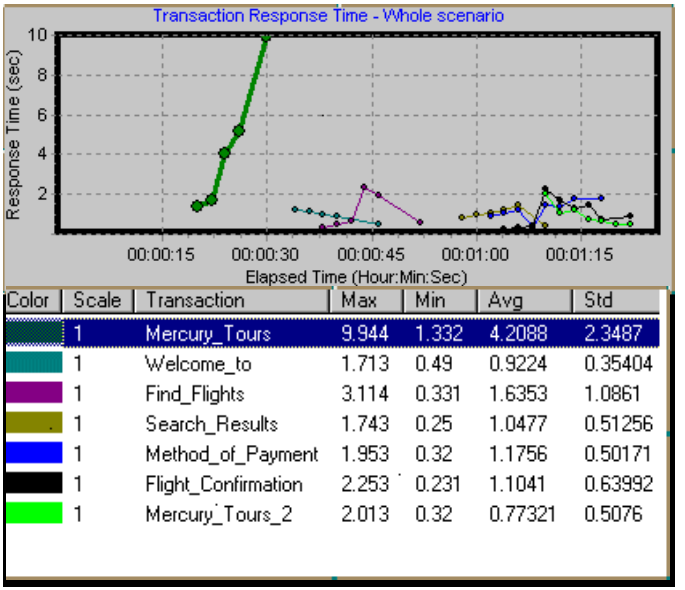
Online Monitoring Graphs

Using the online monitor graphs, you can determine whether transactions transpire within an acceptable amount of time, whether your bandwidth is sufficient to keep download times to a minimum, and whether your hardware and operating system can handle peak load.

Question 1: Do all transactions in my scenario transpire within an acceptable amount of time? Which particular transactions take too long?

Answer: The **Transaction Response Time** graph shows the amount of time it takes for each transaction to be completed. Note that in the graph below, the transaction response time is quick, except for the login transaction. The initial login did not take much time, but subsequent logins were quite slow.

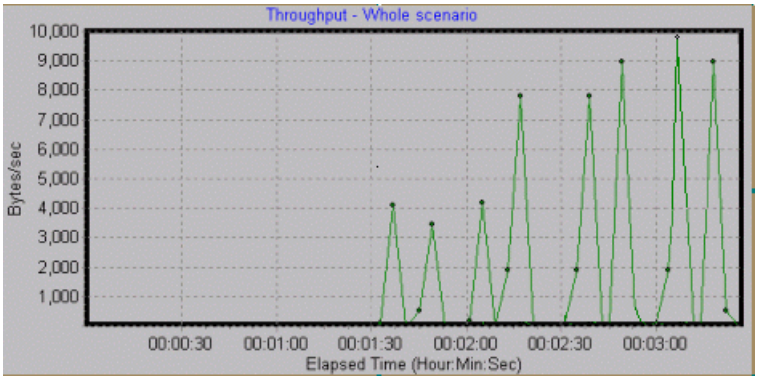
This indicates that the database is unable to process more than one login at a time, which may be due to inefficient database querying.



Color	Scale	Transaction	Max	Min	Avg	Std
	1	Mercury_Tours	9.944	1.332	4.2088	2.3487
	1	Welcome_to	1.713	0.49	0.9224	0.35404
	1	Find_Flights	3.114	0.331	1.6353	1.0861
	1	Search_Results	1.743	0.25	1.0477	0.51256
	1	Method_of_Payment	1.953	0.32	1.1756	0.50171
	1	Flight_Confirmation	2.253	0.231	1.1041	0.63992
	1	Mercury_Tours_2	2.013	0.32	0.77321	0.5076

Question 2: Is bandwidth sufficient to keep download times to a minimum?

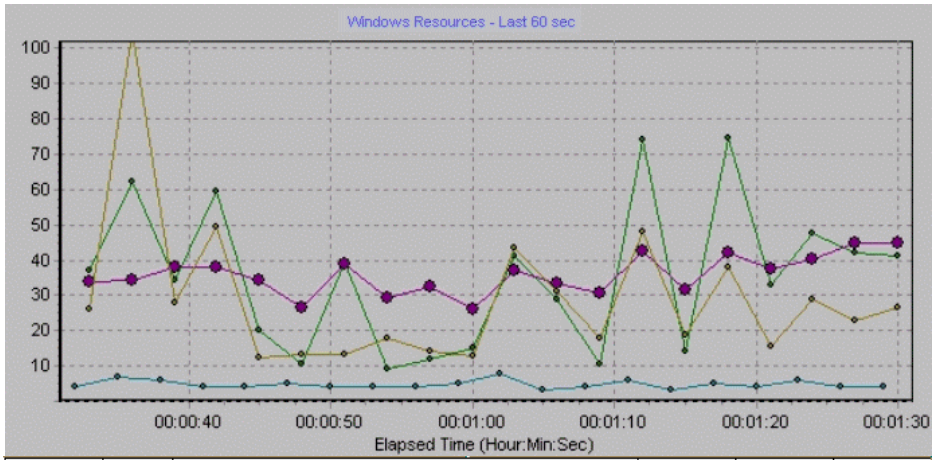
Answer: The **Throughput** graph shows the amount of throughput on the Web server during each second of the scenario run. Throughput represents the amount of data received from the server at any given second.



Note that in the above graph, the throughput scales upward as time progresses and the number of users increases, indicating that the bandwidth is sufficient. If the graph were to remain relatively flat as the number of users increased, it would be reasonable to conclude that the bandwidth is constraining the volume of data requested.

Question 3: Can the hardware and operating system handle peak load?

Answer: The **Windows Resources** graph displays Windows resource usage in real-time. You use this graph to monitor the resources used during a scenario and locate a bottleneck on a particular machine.



Color	Scale	Measurement	Max	Min	Avg	Std
	1	% Total Processor Time (System)	45.003	26.164	35.835	5.4374
	1	File Data Operations/sec (System)	104.94	12.313	29.104	20.834
	1	Page Faults/sec (Memory)	74.61	9.3203	35.301	20.316
	1	Processor Queue Length (System)	8	3	4.7	1.2689

The *% Total Processor Time* in the above graph shows the amount of data processed by the server. *File Data Operations/sec* shows the rate at which the server is issuing Read and Write operations to file system devices. *Page Faults/sec* counts the number of page faults in the processor, representing virtual memory and caching algorithm opportunities.

It is commonly thought that newer and faster servers can resolve slow download times. However the above graph demonstrates that only a small amount of data is processed by the server. The graph indicates that there is adequate processor capacity, and additional server hardware will not result in increased performance. There are cases, however, in which increased performance can be achieved by optimizing the data file system.

B

Performing Path Translation

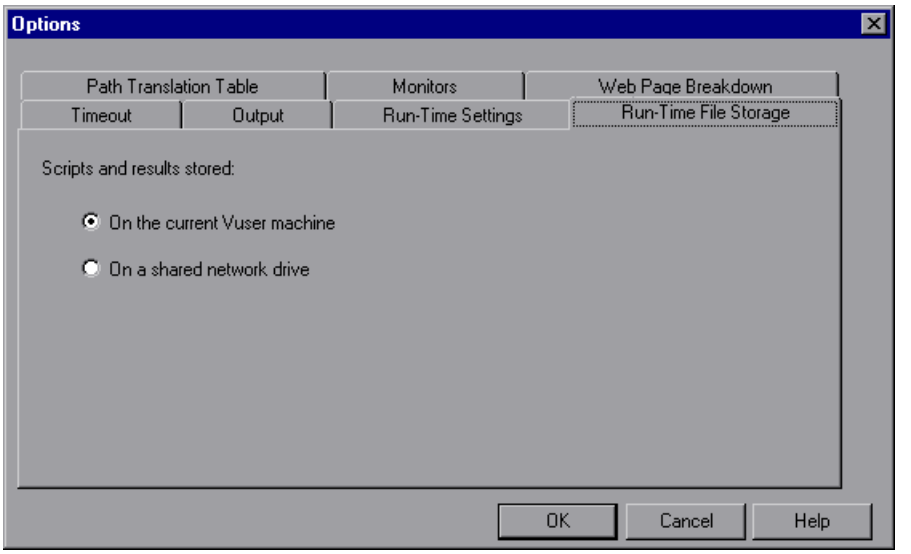
When you run a scenario, LoadRunner gathers run-time data from the participating Vusers. By default, LoadRunner stores the data in temporary files on each Vuser machine. After the scenario, the data is collated in the general results directory.

Alternatively, you can instruct LoadRunner to write the run-time data directly to a shared network drive. (See Chapter 9, "Configuring a Scenario.") This method is not recommended, since it increases network traffic and necessitates path translation.

Understanding Path Translation

Path Translation is a mechanism used by LoadRunner to convert a remote path name for the Controller. A typical scenario might have the LoadRunner Controller running on a Windows-based machine and include multiple Vusers running on both Windows-based and UNIX load generators. One remote load generator may map the network drive as *F*, while another load generator maps the same drive as *H*. In a complex scenario such as this, you need to ensure that all participating machines recognize the same network drive.

You instruct LoadRunner to store scripts and run-time data results on a shared network drive from the Run-time File Storage tab of the Options dialog box.



Result and script files stored on a shared network drive require you to perform path translation.

The Script view contains a list of all the Vuser scripts associated with a scenario—and their locations. A script's location (path) is always based on the Controller machine's mapping of that location. If a Vuser load generator maps to the script's path using a different name, path translation is required.

For example, assume that the Controller is running on a Windows-based machine named *pc2*, and that a Vuser script is located on a network drive. The Controller machine maps the network drive as *m*:\lr_tests. If the remote Vuser machine (load generator) hosting the Vusers also maps the path as *m*:\lr_tests, no translation is necessary. However, if the remote machine maps the path as another drive or path, for example *r*:\lr_tests, you must translate the path to enable the load generator to recognize the script location.

Similarly, when saving run-time result files to a shared drive that is mapped differently by the Controller and remote load generator, you must perform path translation.

Path translation is also effective across platforms—between Windows and UNIX. You use path translation to translate Windows-based paths (as seen by the Controller) into paths recognized by the UNIX Vuser load generator.

Adding Entries to the Path Translation Table

To translate a path from one Windows-based computer to another, or between Windows-based and UNIX machines, you create an entry in the Path Translation table. This table contains a list of paths translated into formats that can be recognized by different machines.

Each line of the Path Translation table has the following format:

<controller_host><controller_path><remote_path>[<remote_host>]

controller_host — The name or type of the machine that is running the Controller. For example, if the Controller is running on a Windows-based computer, you could type win in the host field. Alternatively, you could enter the name of the machine running the Controller (for example, LOADPC1).

The value of *controller_host* can be:

hostname	the name of the machine running the Controller
win	the Controller is running on a Windows-based computer
unix	the Controller is running on a UNIX machine
all	the Controller is running on a Windows-based or a UNIX machine

controller_path	The path of a specific directory—as recognized by the Controller. For example, if the directory *scripts* is located on the network drive *r*—as mapped by the Controller—type the path r:\scripts in the *controller_path* field.
remote_path	The path of a specific directory—as recognized by the remote machine. For example, if the directory *scripts* is located on the network drive *n—as mapped by the remote load generator*—type the path n:\scripts in the *remote_path* field. If a Vuser on the remote UNIX load generator recognizes the above path as /m/tests, you would type this path in the *remote_path* field.
remote_host	The name or type of the remote load generator. For example, if all the remote machines are UNIX workstations, you could type unix in the *remote_host* field. The options for the *remote_host* field are the same as the options for the *controller_host* field, listed above. The *remote_host* parameter is optional.

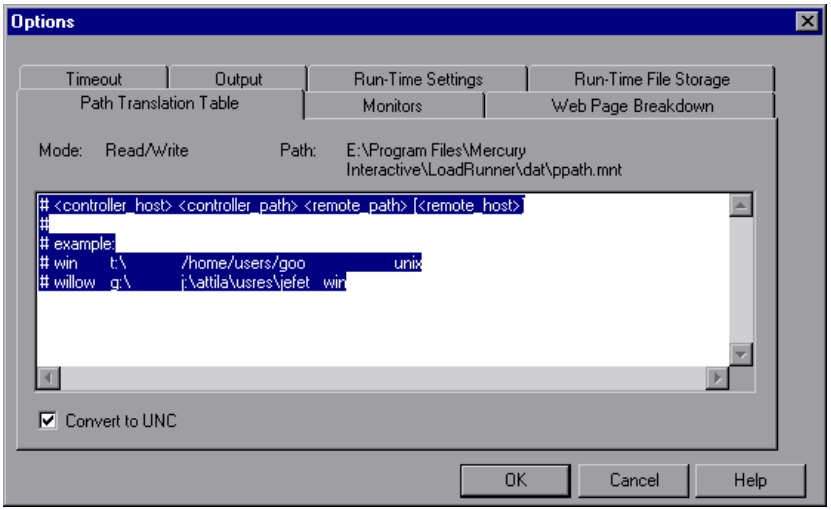
Editing the Path Translation Table

You maintain the Path Translation table using the LoadRunner Controller. LoadRunner saves the Path Translation table as an ASCII file, *ppath.mnt*. This file, stored in loadrunner_directory/dat, has a one–line entry for each network path to translate.

To edit the Path Translation table:

1 Start the LoadRunner Controller.

2 Choose **Tools** > **Options** and select the **Path Translation Table** tab. The Path Translation Table view opens.



3 Before you enter path translation information, consider using the Universal Naming Convention method. If your machines are Windows machines, you can tell the Controller to convert all paths to UNC, and all machines will be able to recognize the path without requiring path translation. An example of UNC format is \\machine_a\results.

Select the Convert to UNC check box to tell LoadRunner to ignore the path translation table and to convert all paths to the Universal Naming Convention.

4 If your machines are not Windows machines and you require path translation, type the path information into the table. You can insert comments by typing the "#" symbol at the start of a line in the table.

5 Click **OK** to close the table and save the information.

Path Translation Examples

The following section illustrates sample Path Translation Table entries.

Note that when you translate a Windows-based path to a UNIX path, you must enter the appropriate slashes—forward slashes for UNIX and back slashes for Windows-based paths.

The examples below show the use of the Path Translation table for a Windows-based Controller called Merlin.

In the first example, Vusers are running on a Windows 95 machine, Oasis. Merlin maps the network drive as f:, while Oasis maps it as g:\loadtest.

merlin	f:\	g:\loadtest\	Oasis

In the second example, Vusers are running on a UNIX machine, Ultra. Ultra maps the networks drive as /u/tests/load.

merlin	f:\	/u/tests/load/	Ultra

In the third example, the mapping of the network drive by the remote load generator Jaguar, is identical to the Controller's mapping, so no translation is required. This line can be excluded from the Path Translation table.

merlin	n:\	n:\	Jaguar

In the fourth example, all Windows-based Vuser load generators map the network drive as m:\loadtest.

merlin	l:\mnt\	m:\loadtest\	win

C

Working in Expert Mode

Advanced users can fine-tune the LoadRunner configuration settings while working in *Expert Mode*. In Expert mode, additional options are displayed in the Options dialog box and in the Load Generator Information dialog box. This chapter describes the additional settings that are available in the Expert mode:

- Entering Expert Mode
- Options - Agent Settings
- Options - General Settings
- Options - Debug Information Settings
- Options - Output Settings
- Options - Monitor Settings
- Load Generator Information - UNIX Environment Settings
- Load Generator Information - Connection Log Settings

Entering Expert Mode

The LoadRunner Controller Expert mode is intended for support personnel to provide access to system information. When you work in the Expert mode, the Controller dialog boxes contain additional options for fine tuning the Controller operation.

To activate the Expert mode, choose **Tools** > **Expert Mode.** An active Expert mode is indicated by a check mark.

To exit the Expert mode, repeat the above process.

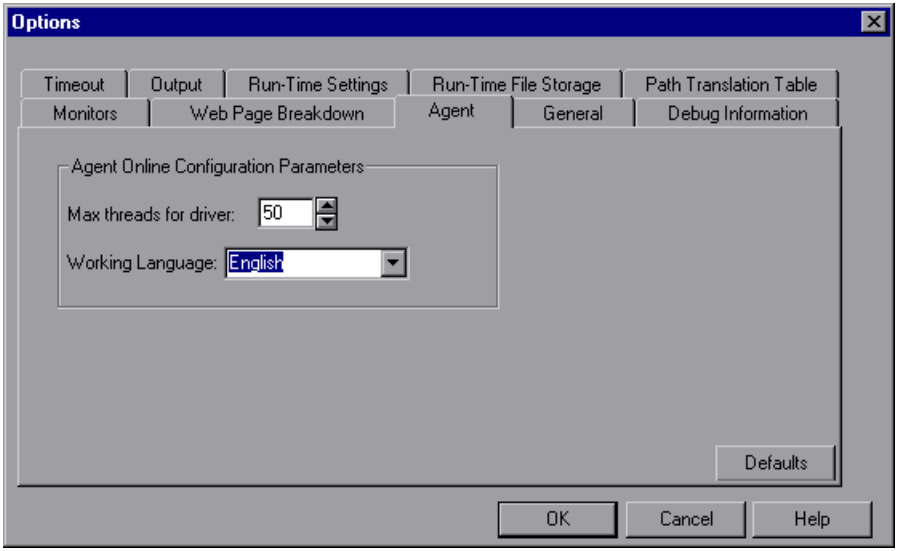
Options - Agent Settings

The Agent settings allow you to customize the behavior of the agent on a remote load generator machine. Using the Options dialog box, you set the online configuration parameters for the agent.

To set the Agent settings:

1 Enter Expert mode (see above).

2 Choose **Tools** > **Options**. The Options dialog box appears. Select the **Agent** tab.



3 Select the maximum number of threads to be executed for the current Vuser's driver.

4 Select the agent's working language (English or Japanese).

5 Click **OK** to accept the settings and close the dialog box.

Options - General Settings

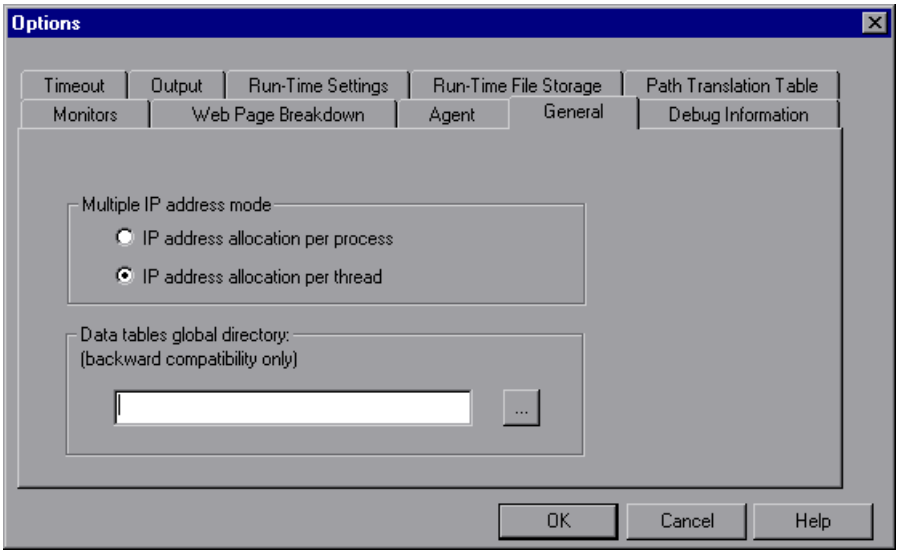
The General tab in the Options dialog box allows you to specify global settings for data table storage and multiple IP address allocation.

Multiple IP address mode: The mode used to allocate IP addresses when the multiple IP address option is enabled (**Scenario** > **Enable IP Spoofer**). The Controller can allocate an IP address per process or per thread. Web Vusers require IP address allocation per process. WinSock Vuser IP addresses can be allocated per thread or per process. Allocation per thread results in a more varied range of IP addresses in a scenario.

Data tables global directory: The network location for data tables used as a source for parameter values. This setting is only required for scripts created with earlier versions of LoadRunner.

To set the General Expert mode settings:

1 Choose **Tools** > **Options**. The Options dialog box appears. Select the **General** tab.



2 Select the Multiple IP address mode.

3 Enter the global directory for data tables.

4 Click **OK** to accept the settings and close the dialog box.

Options - Debug Information Settings

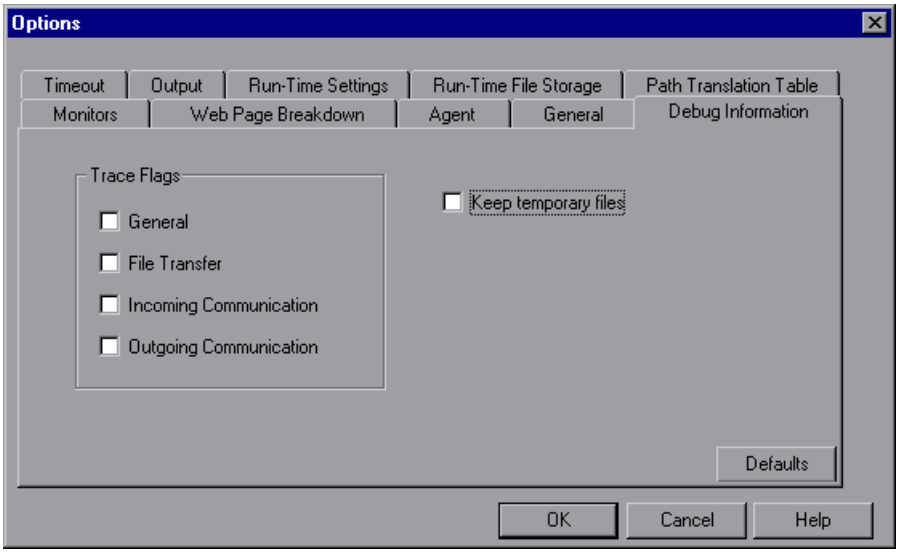
The Debug settings in the Options dialog box allow you to determine the extent of the trace to be performed during scenario execution. The debug information is written to the Output window.

The following trace flags are available: General, File Transfer, Incoming Communication, and Outgoing Communication. You only need to select the flags relating to your problem. For example, if you encounter specific problems with the transfer of files, select the File Transfer flag.

The Agent creates some temporary files in the TMP or TEMP directory, which are normally deleted at the end of the scenario. The temporary files contain information such as the parameter file sent to the Vuser, the output compilation file, and the configuration file. The files are saved in the TMP/lrr#/UsrTmp directory, where # indicates the session number. Using the Debug Information Expert mode settings, you can instruct LoadRunner to keep these temporary files.

To set the Debug Information settings:

1 Choose **Tools** > **Options**. The Options dialog box appears. Select the **Debug Information** tab.



2 Select the check boxes for the desired trace flags.

3 To save the temporary run-time files, select the **Keep temporary files** check box.

4 Click **OK** to accept the settings and close the dialog box.

Options - Output Settings

Basic Output settings are available when Expert mode is disabled. However, Expert mode provides additional settings for handling output messages:

Save Every *xxx* messages: Defines how often to auto-save the output message file. The default is every 100 messages.

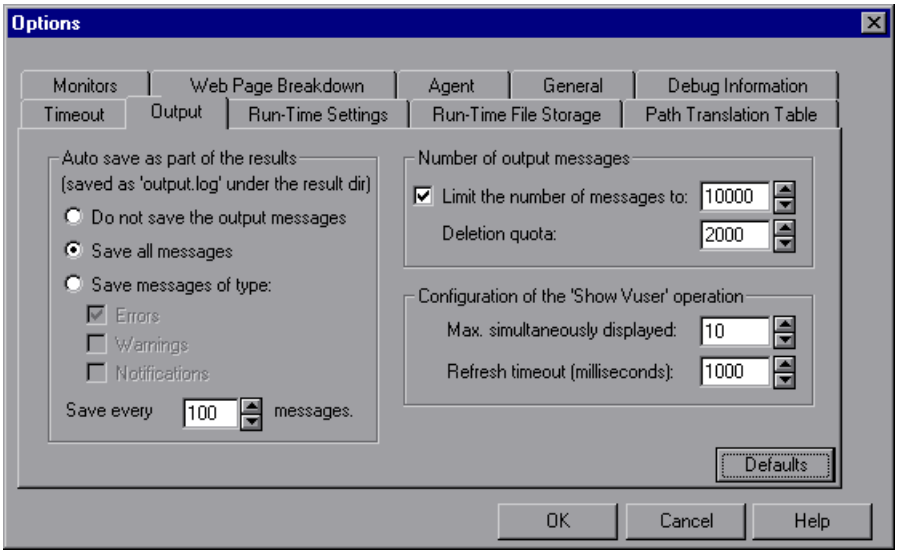
The following two options are used to configure how remote UNIX, GUI, or RTE Vusers are displayed on the Controller machine:

Max Simultaneously Displayed: Specifies the maximum number of active UNIX, GUI, or RTE Vusers that the Controller should display by opening up browsers on your machine. The default is 10 Vusers.

Refresh Time-out: Defines how often to refresh the UNIX Vuser display (in milliseconds). The default is every 1000 milliseconds.

To set the Output settings:

1 Choose **Tools** > **Options**. The Options dialog box appears. Select the **Output** tab.



2 Enter the number of messages after which LoadRunner will perform an auto-save operation.

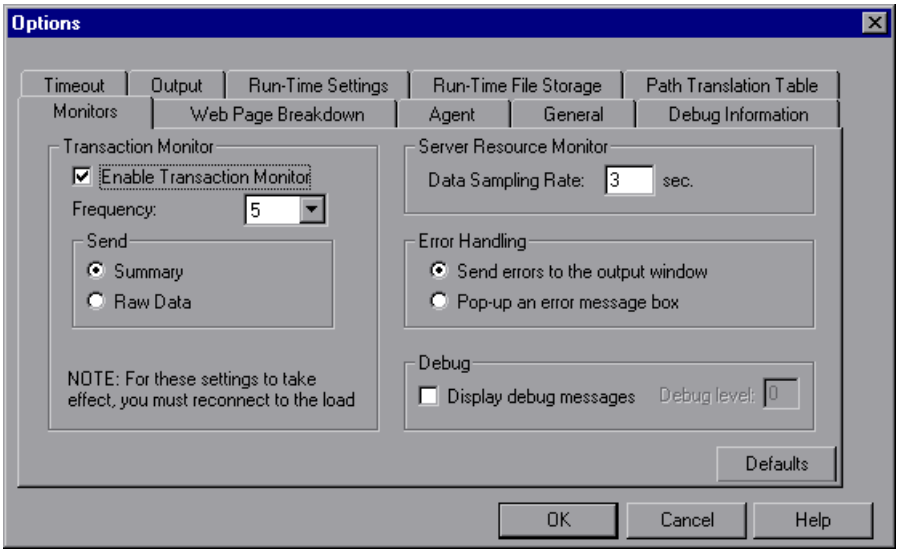
3 Enter values for the relevant Show Vuser options.

4 Click **OK** to accept the settings and close the dialog box.

Options - Monitor Settings

Expert mode provides the following additional monitor setting:

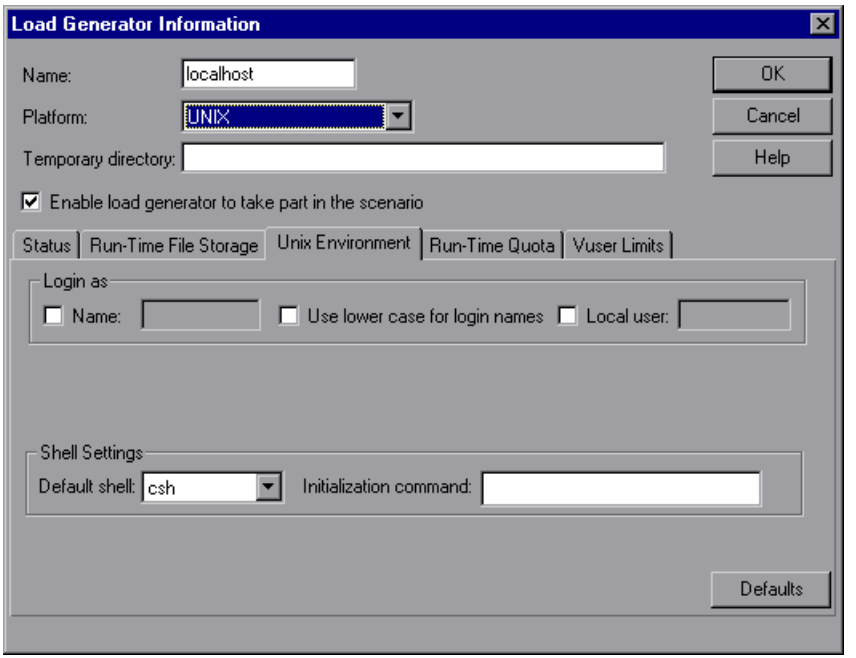
Send Summary or Raw Data: sends a summary of the data collected back to the Controller, or sends all of the data in raw form. Sending the data in raw form saves time because the data does not need to be processed. However, since all of the data is being transferred to the Controller, it may cause more network traffic. If the transfer speed is significant to you, it is recommended that you choose **Summary**.



Load Generator Information - UNIX Environment Settings

Expert mode provides the following additional UNIX Environment setting:

Local User: UNIX load generators that use the *rsh* shell establish a connection as the current Windows user (due to security considerations). To "mislead" rsh and log in as a user other than the current Windows login, select the **Local user** check box and specify the desired UNIX login name. Since modifying the local user name is a security breach for *rsh*, this option should only be used when you encounter a problem connecting to the remote machine.

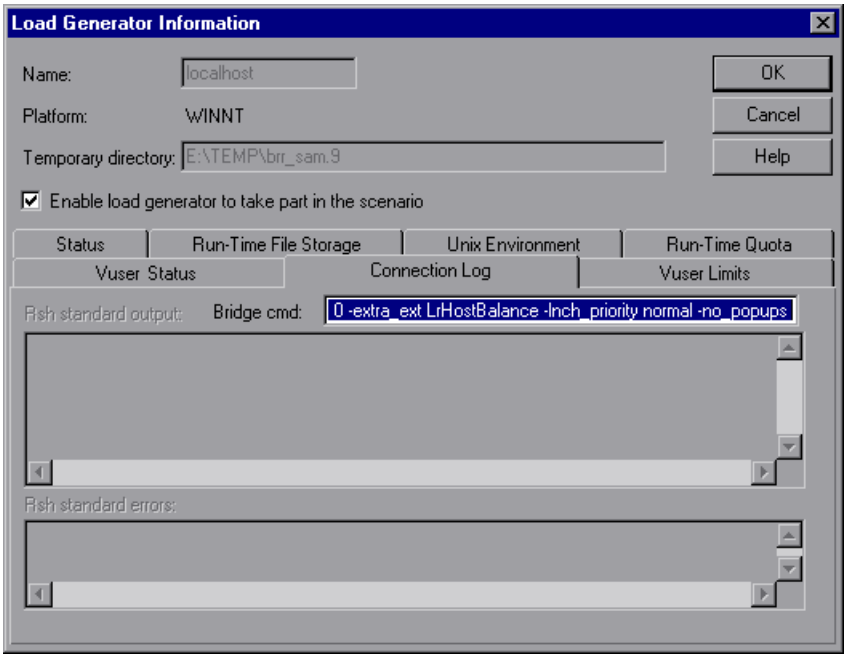


Load Generator Information - Connection Log Settings

The Connection Log tab in the Load Generator dialog box allows you to view the standard output and standard errors generated as the Controller connects to the selected UNIX load generator. You can also change the command that the Controller sends to the remote bridge in order to connect to the load generator.

To set the Connection Log settings:

1. Click the **Generators** button, or select **Scenario** > **Load Generators**. The Load Generators dialog box opens.
2. Click **Connect** to change the Status of a load generator from Down to Ready.
3. Click the **Details** button. The Load Generator Information dialog box opens. Select the **Connection Log** tab.



You can view the Rsh standard output and Rsh standard errors generated as the Controller sends the connection command to the selected UNIX load generator.

In the Bridge cmd box, enter a new command if you want to change the default bridge command being sent by the Controller to the remote bridge in order to connect the UNIX load generator.

D

Troubleshooting the Controller

LoadRunner enables you to test entire applications. If one of the components of the application is not configured properly, LoadRunner scenarios will not run.

This appendix discusses the most common LoadRunner problems:

- LoadRunner Communications
- Failure to Communicate with a Load Generator
- Failure to Connect to the AUT Database
- Failure to Access Files
- Failed Vusers or Transactions
- Increasing the Number of Vusers on a Windows Machine
- Working with Firewall Limitations

About Troubleshooting

LoadRunner relies heavily upon communication between machines on a network. If communication is not established properly, the Controller will be unable to send commands to remote load generators and the scenario will fail. By understanding the reason for the failure and determining when the failure occurred, you can solve most of the communication-related problems.

In order to ensure that the problem lies with your scenario and not your Vuser script, you should verify that your script runs properly on all remote load generators as a stand-alone:

➤ Test your GUI Vuser scripts on Windows platforms using WinRunner.

➤ Test your Vuser scripts on UNIX platforms by running them from the command line.

➤ Test all other types of Vuser scripts on Windows platforms by running them from VuGen, or by running a single user from the Controller.

Note: When a test runs in VuGen, the full browser is used. This differs from a test run in the Controller, where only the browser basics are used. There may be occasions when a test passes its run in VuGen, but fails when it is run in the Controller. Before running a scenario in the Controller with multiple Vusers, run a single Vuser to ensure the test is bug free.

For more information on running Vuser scripts in stand-alone mode, refer to the appropriate guide for creating Vuser scripts.

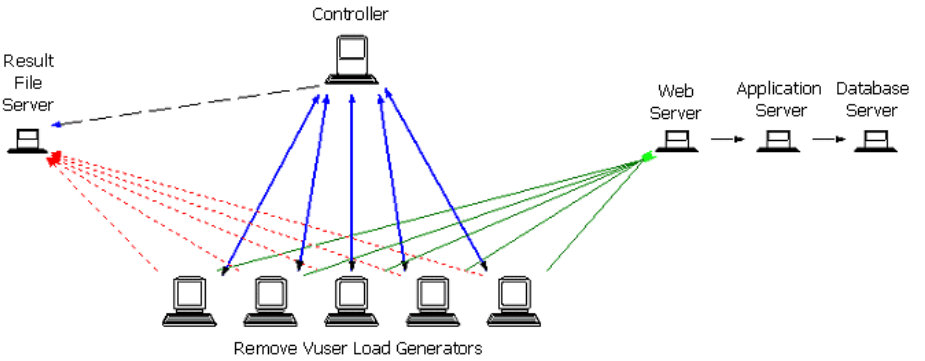
LoadRunner Communications

Most communication problems can be solved if you understand your LoadRunner configuration. This knowledge helps you to determine the source of the problem and perform the necessary actions to correct it.

The following diagram illustrates a sample network running LoadRunner. There are five servers: The LoadRunner Controller, the Web server, the application server, the database server, and the file server which stores the scenario results (note that result files can also be saved on a non-dedicated server). There are five remote load generators, each one running multiple Vusers.

The arrows indicate the type of communication necessary between the elements of the network. The Vusers communicate with the Controller in both directions (send/receive), but with the file server in one direction

(send). The Controller must have access to the file server. All Vusers participating in the scenario must be able to communicate with the Web server in both directions (send/receive). In order for a client machine to connect to the server machine, it must be able to resolve the server machine name.



If any of the connections are broken, the scenario will fail.

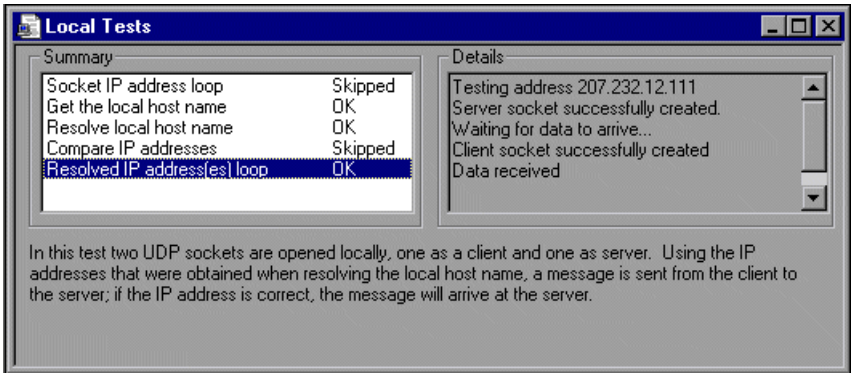
Failure to Communicate with a Load Generator

The most common communication error is the failure of the Controller machine to connect with a remote load generator. Check the following items:

- TCP/IP setup
- TCP/IP connectivity
- Load Generator connections
- UNIX shell

Checking TCP/IP Setup

The first step in checking your configuration is to verify your machine's TCP/IP setup. LoadRunner includes a utility called Hostinfo (hostinfo.exe), located under LoadRunner's bin directory. This utility provides information about the current machine—local name and local address. It also insures that TCP/IP is properly installed on the current machine.



When you invoke Hostinfo, it automatically verifies the TCP stack by:

- retrieving and resolving the local machine name
- retrieving and resolving the IP address

To resolve the IP address, Hostinfo tries to communicate using two UDP sockets on the same machine. It verifies that the IP address obtained while resolving the machine name is the same as the actual IP address of this machine.

To display the results of a test in the Details box, highlight the test name.

Note that the Edit menu in Hostinfo allows you to copy all machine information to the clipboard for sending to support personnel.

Checking TCP/IP Connectivity

Make sure that TCP/IP connectivity is functional on the Controller and Vuser machines. Use a ping utility or type PING <machinename> from the DOS command line to verify communication with a remote machine. Make sure that the remote load generator and Controller machines can ping each other by IP addresses and machine names.

Load Generator Connections

To verify the load generator connectivity, connect to each one of the remote load generators from the Controller's Load Generators dialog box. Specify a platform, WIN or UNIX, for each load generator by double-clicking the Platform field. Select the load generator(s) and click the Connect button. The status changes to *Connecting*.

If the Connection fails, the status changes to *Failed* and details are written to the Details box. Double-click the details box for more information about a failure.

If a connection succeeds, the status changes to *Ready*, and the actual platform name appears in the Platform box (such as WINNT, SUNOS, etc.)

Name	Status	Details
doc9pc	Ready	
goose	Ready	
miro	Ready	
rman	Failed	Connection to host failed. Communication problem: RPC: F
oxygen	Failed	Connection to host failed. Communication problem: RPC: F
jukebox	Failed	Connection to host failed. Communication problem: RPC: F
hammer	Failed	Connection to host failed. Communication problem: RPC: F
steel	Connecting	Connection started.

If your scenario uses several domains (e.g., Vusers on a different domain than the Controller), the Controller may have trouble communicating with the load generators. This happens because by default, the Controller uses the short load generator name, not including the domain. To solve this, you must tell the Controller to determine the full load generator names, including the domains.

➤ Modify the miccomm.ini file in the Controller machine's Windows directory as follows:

```
[tcpnet]
LocalHostNameType= 1
```

The possible values for LocalHostNameType are:

```
0 - Attempt to use the full machine name.
1 - Use the short machine name. This is the default.
```

Note that in certain environments such as WINS, load generators are unable to resolve machine names.

UNIX Shell

For UNIX Vusers, make sure that the Windows Controller can execute a remote shell command. Type the following at the DOS command prompt: rsh <machinename> /bin/ls. If you get a message indicating that permission is denied, make sure the .rhosts file in your UNIX home directory is set up correctly.

Failure to Connect to the AUT Database

If you are running a database application, you must ensure that all remote clients can connect with the database server. If network or configuration errors occur when the client accesses the server, you must correct them before running a scenario. To ensure that your client application can connect with the database server, perform the following tests.

➤ Ping

➤ SQL utilities

Ping: Ensure that the client can communicate with the database server using TCP/IP. Use a ping utility or type PING <machinename> from the DOS command line.

SQL Utilities: Use a simple utility such as ISQL or SQLPLUS to log on to the database server and perform several basic operations.

Failure to Access Files

A LoadRunner scenario will fail if the result path or Vuser script is inaccessible to one or more of the participating machines. Check the following items:

- Path Translation
- Vuser Script
- Result Path

Path Translation: A script's location (path) is always based on the Controller machine's mapping of that location. If a Vuser load generator maps to the script's path using a different name, path translation is required. Path translation translates the Controller's mapping of a given location to the Vuser load generator's mapping. For example, if one machine maps the script directory as *g:\test*, while another maps it as *h:\test*, the paths should be translated.

Path translation is also effective across platforms—between Windows and UNIX. You use path translation to translate the Windows Controller paths into paths recognized by UNIX.

Note: Path translation is only required if you chose to save all scripts and results to a shared network drive. In the default setup, LoadRunner saves files locally and collates them to the Controller machine; no path translation is required.

Suppose that your script is in the /usr/jon/lr_test1 directory and runs on the UNIX machine, *sunny*. To translate it from the Windows Controller machine, *pc1*, where your UNIX directory is mapped as *r*, enter the following line in the path translation table:

pc1	r:\	/usr/jon	sunny

To translate the f:\qa Controller directory to all load generator machines running */m/qa/lr_test2/lr_test2.usr* on a UNIX platform, type:

win	f:\qa	/m/qa	UNIX

If the paths are not translated properly, the scenario will fail. For more information about path translation, see Appendix B, "Performing Path Translation."

Vuser Script: Make sure that the Vuser script is accessible to all load generators participating in the scenario through path translation and permissions. View or run the Vuser script as a stand-alone on each of the participating load generators.

Result Path: Make sure that the result path is accessible to all load generators participating in the scenario through path translation and permissions. Check the permissions of the result directory files and modify them if necessary.

Failed Vusers or Transactions

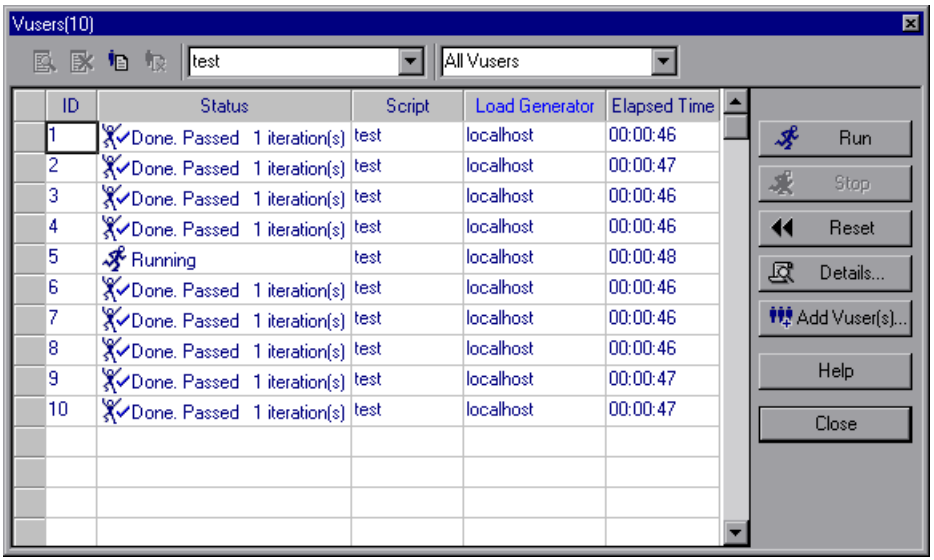
LoadRunner Vusers or transactions may fail for a variety of reasons relating to the network, database, or actual script. You can find information about scenario runs from the following sources:

- Run View
- Output Window
- Output File (excluding GUI Vusers)
- Analysis Reports and Graphs

Run View

The Run view is part of the LoadRunner Controller. The Scenario Groups window in the top left-hand corner of the view indicates the status of the Vuser groups during and after the scenario run. During the scenario run, the columns will show a PENDING, INITIALIZING, READY, RUNNING, or RENDEZVOUS status. You can also view the status of each individual Vuser in the Vusers dialog box. If a Vuser fails and does not complete the script execution, LoadRunner displays an error status. If a Vuser completes the script execution, LoadRunner indicates the transaction status of a completed script run using the DONE.FAILED or DONE.PASSED status.

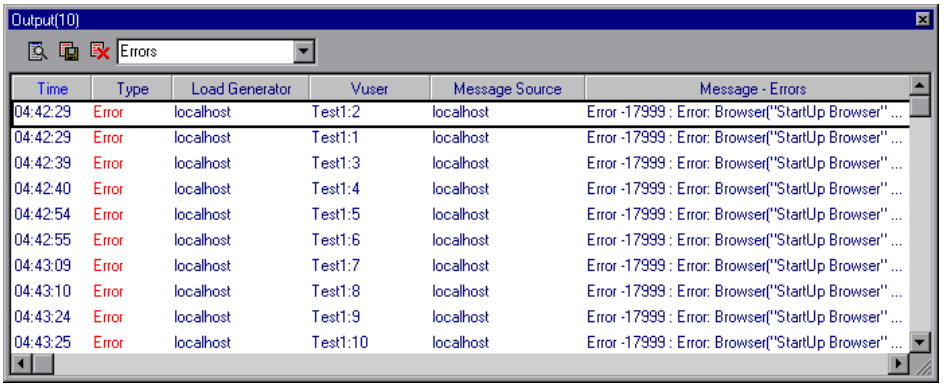
For more information about the Vuser states, see Chapter 12, "Running a Scenario."



Output Window

View the Output window from the Controller. The output window contains useful information for debugging a scenario. The output window lists three types of messages: errors, warnings, and notifications. An error message usually results in a failed script. A warning message indicates that the Vuser encountered a problem, but test execution continued. A notification

provides useful information such as recorded think time values and other run-time information.



For more information about the Output window, see Chapter 13, "Viewing Vusers During Execution."

Output File

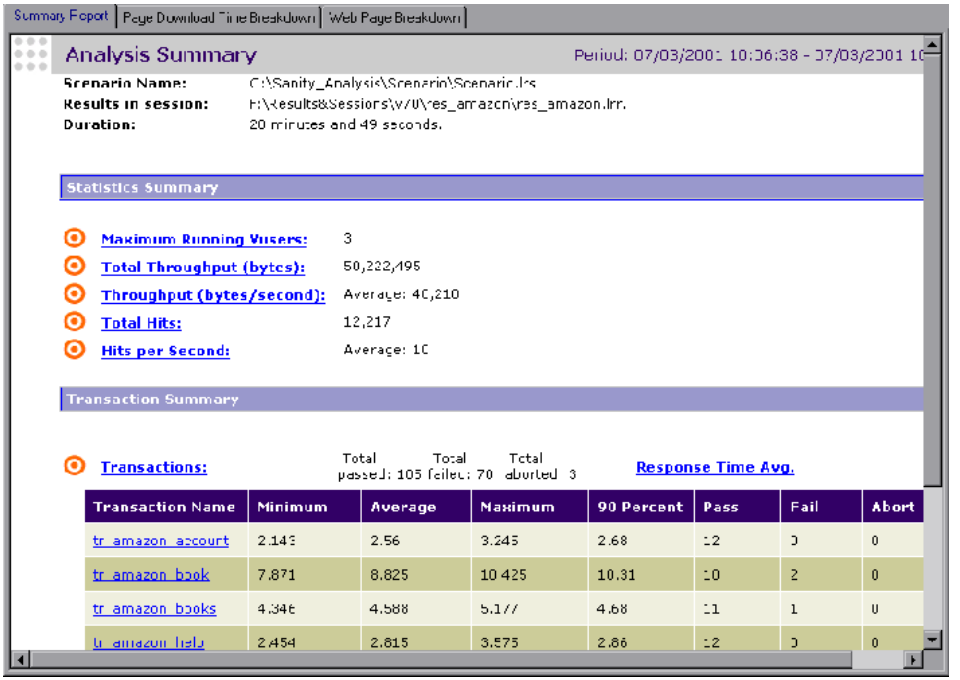
You can view information about script execution in an output file located in the Vuser result directory. The output file, *output.txt*, contains:

- a list of the primary functions called during the scenario
- error messages from the database server
- transactions and rendezvous information

The extent of the information sent to the output file depends on the output file settings. In the VuGen's run-time settings, you specify a Brief or Extended log. For the Extended log, you can specify a full trace, returned data, or current parameter value. An extended log is helpful for debugging a script, but if you are not debugging, Extended log is not recommended as it introduces extra overhead. For more information about configuring run-time settings, refer to the *Creating Vuser Scripts* guide.

Analysis Reports and Graphs

You can generate graphs and reports to view information about the scenario run. For example, the Scenario Summary report displays a table containing the scenario's run-time data and provides links to the following graphs: Running Vusers, Throughput (Web), Hits Per Second (Web), Transaction Summary, and Transaction Response Time - Average.



For more information on the available graphs and reports, see the *LoadRunner Analysis User's Guide*.

Increasing the Number of Vusers on a Windows Machine

Under the normal settings of a Windows machine, you are limited to several hundred Vusers. This limitation is related to the operating system and not to the CPU or memory.

To work around the limitation of the Windows operating system, modify the Windows Kernel as follows:

1. Save a copy of the registry file in case you have trouble with these modifications.
2. Run Regedit.
3. Go to following key in KEY_LOCAL_MACHINE:

   System\CurrentControlSet\Control\Session Manager\SubSystems
4. Select the Windows key. The default Windows key for NT 4.0 looks like this:

   ```
   %SystemRoot%\system32\csrss.exe ObjectDirectory=\Windows
   SharedSection=1024,3072
   Windows=On SubSystemType=Windows ServerDll=basesrv,1
   ServerDll=winsrv:UserServerDllInitialization,3
   ServerDll=winsrv:ConServerDllInitialization,2
   ProfileControl=Off MaxRequestThreads=16
   ```

   The SharedSection=1024,3072 key has the format xxxx,yyyy where:

   xxxx defines the maximum size of the system-wide heap (in kilobytes)

   yyyy defines the size of the per desktop heap.
5. Increase the SharedSection parameter:

   Change the yyyy settings from 3072 to 8192 (which is 8 MB).

This setup successfully allowed 1250 Oracle Vusers to run on a Windows machine using 2 Pentium PRO 200 MHz with 1 GB RAM.

Each Vuser in this setup used approximately 2MB memory. Other Vusers may require more memory.

LoadRunner was able to load over 2500 Vusers when the Windows terminal server was run as the Operating System and the above registry setting was changed.

The above registry changes enable you to run more threads, allowing you to run more Vusers on the machine. This implies that you are not bound by the Windows operating system; you are only bound by hardware and internal scalability limitations.

Working with Firewall Limitations

Working with a firewall means that you can prevent access to the outside world and from the outside world, on specific port numbers.

For example, you can specify that there is no access to any port from the outside world, with the exception of the mail port (23), or you can specify that there is no outside connection to any ports except for the mail port and WEB port (80). The port settings are configured by the system administrator.

LoadRunner uses *rsh* for launching the Agents on UNIX machines and *rcmdl* (via the PortMapper) on PC machines. When the Agent and the Controller are brought up, they look for a free port number from the range of free ports reserved for programs. If there is a firewall, then the Agent or the Controller may not find a free port.

Therefore, make sure that the following ports are open:

- The RSH port number and the PortMapper port number (111) on the Vuser machines should be open to access from the outside.
- The User range on the Vuser and Controller machines should be open to access in both directions.

Note that Vusers usually need to access a server machine (for example, Web server). A firewall in this case also poses a problem: the Vusers will not be able to run the client program at all.

E

Troubleshooting Online Monitors

LoadRunner monitors allow you to view the performance of the scenario during execution.

The following sections describe several tips and known issues relating to the online monitors.

- Troubleshooting Server Resource Monitors
- Troubleshooting the Network Delay Monitor
- Network Considerations

Troubleshooting Server Resource Monitors

In order to monitor resources on a server machine, you must be able to connect to that machine. If monitoring is unsuccessful and LoadRunner cannot locate the specified server, make sure that the specified server is available. Perform a "ping" operation by typing ping *server name* from the Controller machine command line.

Once you verify that the machine is accessible, check this table for additional tips on troubleshooting the monitor.

Problem	Solution
Cannot monitor a Windows machine on a different domain, or "access denied."	To gain administrative privileges to the remote machine, perform the following from the command prompt: %net use \\<MachineName>/ user:[<Domain>\<RemoteMachineUsername>] At the password prompt, enter the password for the remote machine.

Problem	Solution
Cannot monitor an NT/Win 2000 machine (An error message is issued: "computer_name not found" or "Cannot connect to the host")	The NT/Win 2000 machine you want to monitor only enables monitoring for users with administrator privileges. In order to allow monitoring for non-admin users, you must grant read permission to certain files and registry entries (Microsoft tech-note number Q158438.) The required steps are: **a.** Using Explorer or File Manager, give the user READ access to: %windir%\system32\PERFCxxx.DAT %windir%\system32\PERFHxxx.DAT where *xxx* is the basic language ID for the system, for example, 009 for English. These files may be missing or corrupt. If you suspect this; expand these files off of the installation cd. **b.** Using REGEDT32, give the user READ access to HKEY_LOCAL_MACHINE\Software\Microsoft\Windows NT\CurrentVersion\Perflib and all sub keys of that key. **c.** Using REGEDT32, give the user at least READ access to HKEY_LOCAL_MACHINE\System\CurrentControlSet\Control\SecurePipeServers\winreg
Some Win 2000 counters cannot be monitored from an NT machine.	Run the Controller on a Win 2000 machine.
Some Windows default counters are generating errors	Remove the problematic counters and add the appropriate ones using the "Add Measurement" dialog box.
You cannot get performance counters for the SQL server (version 6.5) on the monitored machine.	There is a bug in SQL server version 6.5. As a workaround, give read permission to the following registry key at the monitored machine (use regedt32): HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\MSSQLServer\MSSQLServer (Microsoft tech-note number Q170394)

Problem	Solution
The selected measurements are not displayed in the graph.	Ensure that the display file and online.exe are registered. To register the monitor dll's, without performing a full installation, run the *set_mon.bat* batch file located in lrun/bin.
When monitoring a Windows machine, no measurements appear in the graph.	Check the built-in Windows Performance Monitor. If it is not functional, there may be a problem with the communication setup.
When monitoring a UNIX machine, no measurements appear in the graph.	Ensure that an *rstatd* is running on the UNIX machine (Refer to Chapter 16, "System and Database Resource Monitoring.").
Cannot monitor one of the following Web servers: MS IIS, MS ASP, or ColdFusion	Refer to problem above, "Cannot monitor a Windows machine."

Troubleshooting the Network Delay Monitor

If monitoring is unsuccessful and LoadRunner cannot locate the source or destination machines, make sure that the specified machines are available to your machine. Perform a "ping" operation. At the command line prompt, type:

```
Ping server_name
```

To check the entire network path, use the trace route utility to verify that the path is valid.

For Windows, type tracert <*server_name*>.

For UNIX, type traceroute <*server_name*>.

Network Considerations

If you notice extraordinary delays on the network, refer to one of the following sections to increase the performance:

- Network Bandwidth Utilization
- Ethernet-bus Based Networks
- Working on a WAN or Heavily Loaded LAN

Network Bandwidth Utilization

In most load-testing scenarios, the network card has little impact on scenario performance. Network cards are manufactured to handle the bandwidth of the physical network layer. Packets are transferred over an Ethernet at a rate that complies with IEEE 803.x standards. If the network becomes a bottleneck, the issue is not the brand of the network card, but rather the bandwidth limitations on the physical layer (--i.e. Ethernet, FDDI, ATM, Ethernet Token-ring, etc.).

That is, instead of load testing over a T10 line, upgrade your line to DS3 (45Mbps), or T100 (100Mbps).

Below are a few tips that will help qualify the need to upgrade the network:

1) Run the performance monitor on the Vuser load generators. As the number of Vusers increase, check the network byte transfer rate for saturation. If a saturation point has been reached, do not run any more Vusers without upgrading the network—otherwise performance of Vusers will degrade. Degradation is exponential in networking environments.

2) Run the performance monitor on the server machine. Run many Vusers on several load generator machines. Check the kernel usage and network transfer rate for saturation. If saturation is reached with less than the desired Vuser load, upgrade the network.

3) Every network has a different Maximum Transmission Unit or MTU, which is set by the network administrator. The MTU is the largest physical packet size (in bytes) that a network can transmit. If a message is larger than the MTU, it is divided into smaller packets before being sent.

If clients and servers are passing large data sets back and forth, instruct the network administrator to increase the MTU in order to yield better bandwidth utilization. On Windows 95, you can also set the MTU of your machine. Ideally, you want the MTU to be the same as the smallest MTU of all the networks between your machine and a message's final destination.

If you send a message that is larger than one of the MTUs, it will be broken up into fragments, slowing transmission speeds. If the MTU is too high, it may cause unintended degradation. Trial and error is the only sure way of finding the optimal MTU, but there are some guidelines that can help. For example, most Ethernet networks have an MTU of 1500, the default MTU setting for Windows 95.

If the desired MTU reduces performance, upgrade the network or reduce the MTU to improve performance.

Ethernet-bus Based Networks

The following guidelines apply to Ethernet-bus based networks:

Networks with only 2 active machines communicating yield a maximum of 90% bandwidth utilization.

Networks with 3 active machines communicating yield a maximum of approximately 85% bandwidth utilization.

As the number of active machines on the network increases, the total bandwidth utilization decreases.

Working on a WAN or Heavily Loaded LAN

When you work with LoadRunner on a WAN or heavy loaded LAN, you may notice some unusual LoadRunner behavior, which indicates network problems. The Output window may contain messages about retries, lost packets, or message mismatch. This is because some of the messages from the Controller may not be reaching the Agent. To solve this problem, you should reduce the network traffic or improve the network bandwidth.

The following steps may help reduce network traffic:

➤ Click the **Run-Time Settings** button. In the Log tab, select **Disable logging**.

➤ Choose **Tools** > **Options**. In the Output tab, select **Do not save the output messages**.

➤ Initialize all users before running them. Run them only after initialization is completed.

F

Working with Server Monitor Counters

When you configure the System Resource, Microsoft IIS, Microsoft ASP, ColdFusion, and SQL Server monitors, you are presented with a list of default counters that you can measure on the server you are monitoring. Using the procedure described below, you can create a new list of default counters by including additional counters, or deleting existing counters.

In addition, there are specific counters that are especially useful for determining server performance and isolating the cause of a bottleneck during an initial stress test on a server.

The following sections describe:

- Changing a Monitor's Default Counters
- Useful Counters for Stress Testing

Changing a Monitor's Default Counters

You can change the default counters for the System Resource, Microsoft IIS, Microsoft ASP, ColdFusion, or SQL Server monitors by editing the res_mon.dft file found in the LoadRunner/dat directory.

To change the default counters:

1. Open a new scenario and click the **Run** tab.
2. For each of the monitors, select the counters you want to measure.
3. Save the scenario and open the scenario *.lrs file with an editor.

4 Copy the MonItemPlus section of the each counter you selected into the res_mon.dft file.

5 Count the number of new counters in the file and update the **ListCount** parameter with this number.

Useful Counters for Stress Testing

Certain counters are especially useful for determining server performance and isolating the cause of a bottleneck during an initial stress test on a server.

The following is a list of counters that are useful for monitoring Web server performance:

Object	Counter
Web Service	Maximum Connections
Web Service	Bytes Total/sec
Web Service	Current NonAnonymous Users
Web Service	Current Connections
Web Service	Not Found Errors
Active Server Pages	Requests/sec
Active Server Pages	Errors/sec
Active Server Pages	Requests Rejected
Active Server Pages	Request Not Found
Active Server Pages	Memory Allocated
Active Server Pages	Requests Queued
Active Server Pages	Errors During Script Run Time
Memory	Page Faults/sec
Server	Total Bytes/sec
Process	Private Bytes/Inetinfo

The following is a list of counters that are useful for monitoring SQL Server performance:

Object	Counter
SQLServer	User Connections
SQLServer	Cache Hit Ratio
SQLServer	Net-Network Reads/sec
SQLServer	I/O-Lazy Writes/sec
SQLServer-Locks	Total Blocking Locks
PhysicalDisk	Disk Queue Length

The following is a list of counters that are useful for monitoring both Web and SQL server performance:

Object	Counter
Processor	% Total Processor Time
PhysicalDisk	% Disk Time
Memory	Available Bytes
Memory	Pool Nonpaged Bytes
Memory	Pages/sec
Memory	Committed Bytes
System	Total Interrupts/sec
Object	Threads
Process	Private Bytes:_Total

Note: The % Disk Time counter requires that you run the diskperf -y utility at the command prompt and reboot your machine.

G

Configuring Multiple IP Addresses

When you run a scenario, the Vusers on each load generator machine use the machine's IP address. You can define multiple IP addresses on a load generator machine to emulate a real-life situation in which users sit on different machines.

This appendix describes:

- Adding IP Addresses to a Load Generator
- Using the IP Wizard
- Configuring Multiple IP Addresses on UNIX
- Updating the Routing Table
- Enabling Multiple IP Addressing from the Controller

The multiple IP address feature is applicable to the Web and WinSocket protocols. This feature can be implemented on Windows and UNIX platforms.

About Multiple IP Addresses

Application servers and network devices use IP addresses to identify clients. The application server often caches information about clients coming from the same machine. Network routers try to cache source and destination information to optimize throughput. If many users have the same IP address, both the server and the routers try to optimize. Since Vusers on the same load generator machine have the same IP address, server and router optimizations do not reflect real-life situations.

LoadRunner's multiple IP address feature enables Vusers running on a single machine to be identified by many IP addresses. The server and router recognize the Vusers as coming from different machines and as a result, the testing environment is more realistic.

Adding IP Addresses to a Load Generator

LoadRunner includes an IP Wizard program that you run on each Windows NT or Windows 2000 load generator machine to create multiple IP addresses. You add new IP addresses to a machine once and use the addresses for all scenarios. For information about adding IP addresses on Unix machines, see "Configuring Multiple IP Addresses on UNIX" on page 367.

The following procedure summarizes how to add new IP addresses to a load generator:

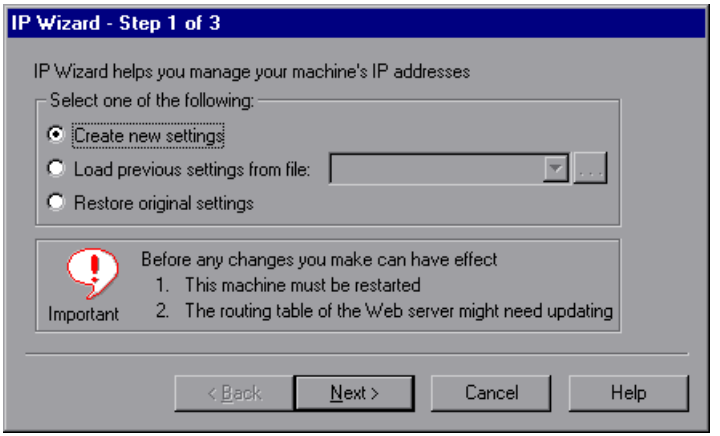
1. Run the IP Wizard on the load generator machine to add a specified number of IP addresses. Manually configure the new IP addresses for UNIX load generator machines.
2. Restart the machine.
3. Update the server's routing table with the new addresses, if necessary.
4. Enable this feature from the Controller. Refer to "Enabling Multiple IP Addressing from the Controller" on page 369.

Using the IP Wizard

The IP Wizard resides on each load generator machine. You run this process once to create and save new IP addresses on Windows machines. The new addresses can be a range of addresses defined by the Internet Assignment Numbers Authority. They are for internal use only, and cannot connect to the Internet. This range of addresses is the default used by the IP Wizard.

To add new IP addresses to a load generator machine:

1 Invoke the IP Wizard from the LoadRunner program group.

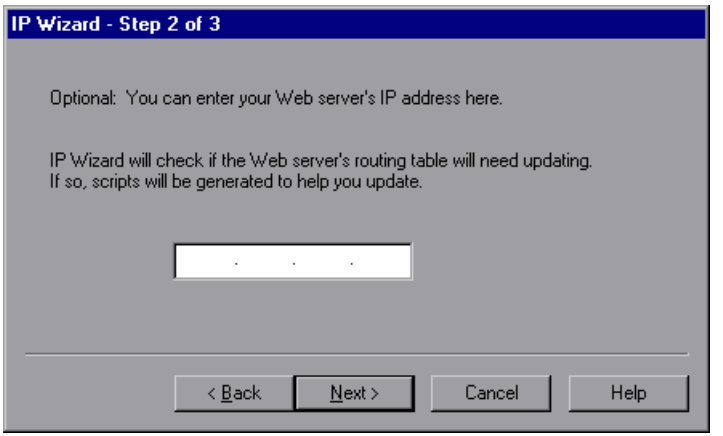


2 If you have an existing file with IP address settings, select **Load previous settings from file** and choose the file.

3 If you are defining new settings, select **Create new settings**.

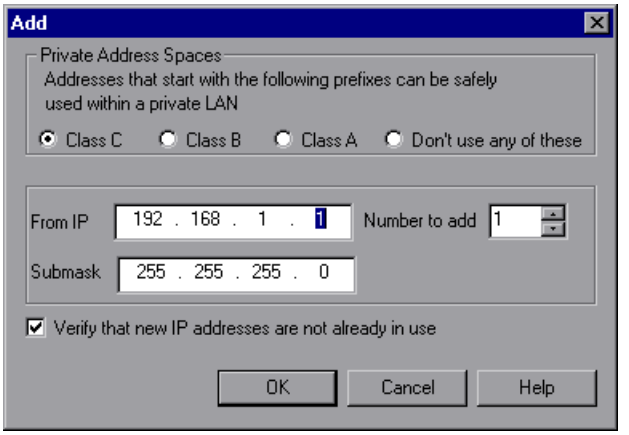
4 Click **Next** to proceed to the next step. If you have more than one network card, choose the card to use for IP addresses and click **Next**.

The optional Web server IP address step enables the IP Wizard to check the server's routing table to see if it requires updating after the new IP addresses are added to the load generator.



5 To check the server's routing table directly after adding the addresses, enter the server IP address. Refer to "Updating the Routing Table" on page 368 for more information.

6 Click **Next** to see a list of the machine's IP address(es). Click **Add** to define the range of addresses.



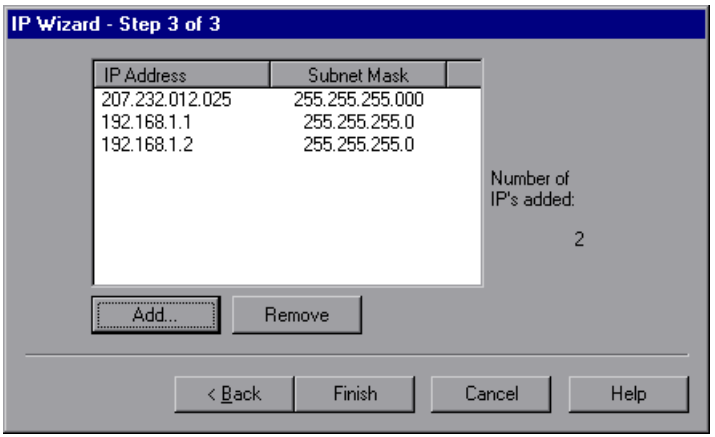
IP addresses include two components, a *netid* and *hostid*. The submask determines where the netid portion of the address stops and where the hostid begins.

7 Select a class that represents the correct submask for the machine's IP addresses.

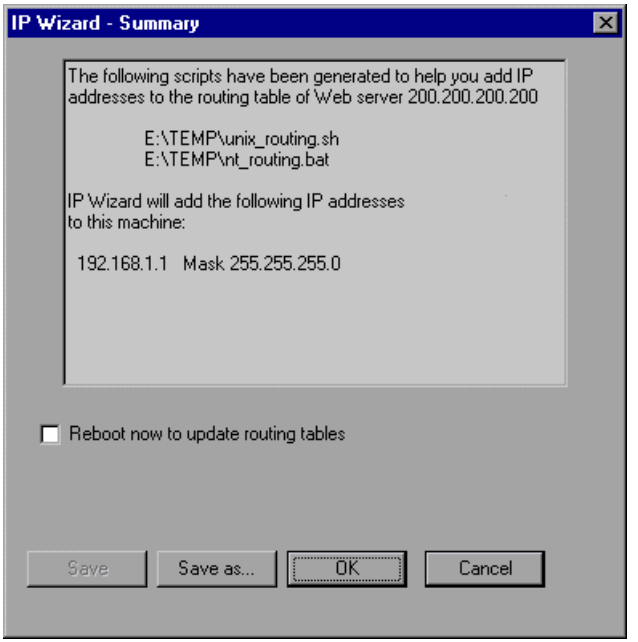
8 Specify the number of addresses to create. Select **Verify that new IP addresses are not already in use** to instruct the IP Wizard to check the new addresses. The IP Wizard will only add the addresses not in use.

9 Click **OK** to proceed.

After the IP Wizard creates the new addresses, the summary dialog box lists all of the IP addresses.



10 Click **Finish** to exit the IP Wizard. The IP Wizard Summary dialog box is displayed.



11 Note the address of the .bat file, and see "Updating the Routing Table" on page 368 for information about using the batch file to update the routing table, if necessary.

12 After you update the routing table, check **Reboot now to update routing tables** to initialize the NT device drivers with the new addresses.

13 Click **OK**.

Configuring Multiple IP Addresses on UNIX

To configure multiple IP addresses on UNIX, manually configure the addresses on the load generator machine.

Solaris 2.5, 2.6, 7.0, 8.0

To configure the hme0 device to support more than one IP address:

1 Create entries in /etc/hosts for each hostname on your physical machine:

```
128.195.10.31  myhost
128.195.10.46  myhost2
128.195.10.78  myhost3
```

2 Create */etc/hostname.hme0:n* files that contain the hostname for the virtual host *n*. Note that *hostname.hme0:0* is the same as *hostname.hme0*.

```
/etc/hostname.hme0   (Contains name myhost)
/etc/hostname.hme0:1 (Contains name myhost2)
/etc/hostname.hme0:2 (Contains name myhost3)
```

The above changes will cause the virtual hosts to be configured at boot time.

3 You can also directly enable/modify a logical hosts configuration by running *ifconfig* directly on one of the logical hosts, using the *hme0:n* naming scheme:

```
% ifconfig hme0:1 up
% ifconfig hme0:1 129.153.76.72
% ifconfig hme0:1 down
```

To verify the current configuration, use *ifconfig –a.*

Linux

To define multiple IP addresses for a single Ethernet card, you need IP Aliasing compiled into the kernel. To do this, use the *ifconfig* command:

```
/sbin/ifconfig eth0:0 x.x.x.x netmask 255.255.x.x up
```

Substitute the new IP address for x.x.x.x, and insert the correct information for subnet mask. Place this command in the *rc.local* file so that it executes upon boot.

HP 11.0 or higher

To define multiple IP addresses for a single Ethernet card, you need IP Aliasing compiled into the kernel. To do this, use the *ifconfig* command:

```
/sbin/ifconfig lan1:0 x.x.x.x netmask 255.255.x.x up
```

Substitute the new IP address for x.x.x.x, and insert the correct information for subnet mask. Place this command in the *rc.local* file so that it executes upon boot.

Updating the Routing Table

Once the client machine has new IP addresses, the server needs the addresses in its routing table, so that it can recognize the route back to the client. If the server and client share the same netmask, IP class, and network, the server's routing table does not require modification.

Note: If there is a router between the client and server machines, the server needs to recognize the path via the router. Make sure to add the following to the server routing table: route from the Web server to the router, and routes from the router to all of the IP addresses on the load generator machine.

To update the Web server routing table:

1 Edit the batch file that appears in the IP Wizard Summary screen. An example .bat file is shown below.

```
REM This is a bat file to add IP addresses to the routing table of a
server
REM Replace [CLIENT_IP] with the IP of this machine that the server
already recognizes
REM This script should be executed on the server machine

route ADD 192.168.1.50 MASK 255.255.255.255 [CLIENT_IP] METRIC 1
route ADD 192.168.1.51 MASK 255.255.255.255 [CLIENT_IP] METRIC 1
route ADD 192.168.1.52 MASK 255.255.255.255 [CLIENT_IP] METRIC 1
route ADD 192.168.1.53 MASK 255.255.255.255 [CLIENT_IP] METRIC 1
route ADD 192.168.1.54 MASK 255.255.255.255 [CLIENT_IP] METRIC 1
```

2 For each occurrence of **[CLIENT_IP]**, insert your IP address instead.

3 Run the batch file on the server machine.

Enabling Multiple IP Addressing from the Controller

Once you define multiple IP addresses, you set an option to tell the Controller to use this feature.

To enable multiple IP addressing from the Controller:

1 In the Controller Design view, select **Scenario** > **Enable IP Spoofer**.

Note: You must select this option before connecting to a load generator.

---

2 Use the **General Options** of the Controller Expert Mode to specify how the Controller should implement this feature.

For more information, refer to Appendix C, “Working in Expert Mode.”

Index

L

M

N

O

P

R

S

T

U

V

W

Host Resolution Functions Copyright Agreement

Copyright (c) 1980, 1983, 1985, 1987, 1988, 1989, 1990, 1993

The Regents of the University of California. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. All advertising materials mentioning features or use of this software must display the following acknowledgement:

   This product includes software developed by the University of California, Berkeley and its contributors.

4. Neither the name of the University nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS ''AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Portions Copyright (c) 1993 by Digital Equipment Corporation.

Permission to use, copy, modify, and distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies, and that the name of Digital Equipment Corporation not be used in advertising or publicity pertaining to distribution of the document or software without specific, written prior permission.

THE SOFTWARE IS PROVIDED "AS IS" AND DIGITAL EQUIPMENT CORP. DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL DIGITAL EQUIPMENT CORPORATION BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Copyright (c) 1996 by Internet Software Consortium.

Permission to use, copy, modify, and distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND INTERNET SOFTWARE CONSORTIUM DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL INTERNET SOFTWARE CONSORTIUM BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Mercury Interactive Corporation
1325 Borregas Avenue
Sunnyvale, CA 94089 USA

Main Telephone: (408) 822-5200
Sales & Information: (800) TEST-911
Customer Support: (877) TEST-HLP
Fax: (408) 822-5300

Home Page: www.mercuryinteractive.com
Customer Support: support.mercuryinteractive.com