Astra® LoadTest™

Controller User's Guide

Version 3.0

MERCURY INTERACTIVE

Astra Controller User's Guide, Version 3.0

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

ALTCUG3.0/02

Table of Contents

PART III: RUNNING A SCENARIO

PART IV: MONITORING A SCENARIO

PART V: ANALYZING TEST RESULTS

PART VI: APPENDIXES

Welcome to Astra LoadTest

Welcome to Astra LoadTest, Mercury Interactive's tool for testing the performance of Web-based applications. Astra LoadTest stresses your entire Web-based application to isolate and identify bottlenecks and optimize the end user experience.

Astra LoadTest enables you to test your system under controlled and peak load conditions. To generate load, Astra LoadTest runs hundreds 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 Web-based application just as real users would. Astra LoadTest's in-depth reports and graphs provide the information that you need to evaluate the performance of your Web-based application.

Online Resources

Astra LoadTest includes the following online tools:

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

Books Online displays the complete documentation set in PDF format. Online books can be read and printed using Adobe Acrobat Reader 4.0, which is included in the installation package. Check Mercury Interactive's Customer Support web site for updates to Astra LoadTest online books.

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

Technical Support Online uses your default web browser to open Mercury Interactive's Customer Support web site.

Support Information presents the locations of Mercury Interactive's Customer Support web site and home page, the e-mail address for sending information requests, the name of the relevant news group, the location of Mercury Interactive's public FTP site, 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.

Astra LoadTest Documentation Set

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

- install Astra LoadTest
- create Vuser scripts using the Virtual User Recorder
- use the Astra LoadTest Controller to run and analyze load test scenarios

Using the Astra LoadTest Documentation Set

The Astra LoadTest documentation set consists of an Installation Guide, Controller User's Guide, and a Virtual User Recorder User's Guide.

Installation Guide

For instructions on installing Astra LoadTest, refer to the *Astra LoadTest Installation* guide.

Controller User's Guides

For a description of how to create and run Astra LoadTest scenarios using the Astra LoadTest Controller, refer to the *Astra LoadTest Controller User's Guide* . The Controller User's Guide presents an overview of the Astra LoadTest testing process.

Guide for Creating Vuser Scripts

The *Virtual User Recorder User's* guide describes how to create Vuser scripts that you will use in your load testing.

Note: The Virtual User Recorder is also used to create functional tests for Astra QuickTest.

Throughout this guide, the use of the term *script* is synonymous with the term *test* for the Virtual User Recorder.

Typographical Conventions

This book uses the following typographical conventions:

Bold	**Bold** text indicates function names and the elements of the functions that are to be typed in literally.
Italics	*Italic* text indicates variable names and book names.
Helvetica	The *Helvetica* font is used for examples and statements that are to be typed in literally.
[ ]	Square brackets enclose optional parameters.
{ }	Curly brackets indicate that one of the enclosed values must be assigned to the current parameter.
...	In a line of syntax, three dots indicate that more items of the same format may be included. In a program example, three dots are used to indicate lines of a program that have been intentionally omitted.
|	A vertical bar indicates that either of the two options separated by the bar should be selected.

Part I

Understanding Astra LoadTest

1

Introduction

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

About Load Testing

Modern Web-based 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 single application, Web-based testing focuses on performance and reliability of an entire Web-based application.

For example, a typical Web-based testing scenario might depict 100 Web Users that login simultaneously to a system: What is the response time of the system? Does the system crash? To be able to answer these questions—and more—a complete Web-based application testing solution must:

- test a system that combines a variety of software applications and hardware platforms
- determine the suitability of a Web server for any given application
- emulate an environment where multiple users interact with a single server application
- test a Web-based application under the load of tens or hundreds 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 Astra LoadTest Solution

The Astra LoadTest automated solution addresses the drawbacks of manual performance testing:

- Astra LoadTest 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, Astra LoadTest reduces the hardware requirements.
- The Astra LoadTest Controller allows you to easily and effectively control all the Vusers—from a single point of control.
- Astra LoadTest monitors the Web-based performance online, enabling you to fine-tune your system during test execution.
- Astra LoadTest automatically records the performance of the Web-based application during a test. You can choose from a wide variety of graphs and reports to view the performance data.
- Astra LoadTest checks where performance delays occur: network or client delays, CPU performance, I/O delays, or other issues at the database or Web server. Astra LoadTest monitors the Web resources to help you improve performance.

➤ Because Astra LoadTest tests are fully automated, you can easily repeat them as often as you need.

Using Astra LoadTest

Scenarios

Using Astra LoadTest, you divide your Web-based 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, Astra LoadTest replaces human users with *virtual users* or *Vusers*. When you run a scenario, Vusers emulate the actions of human users—operating your Web-based 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 or hundreds of Vusers.

To emulate conditions of heavy user load, you create a large number of Vusers that perform a series of tasks. For example, you can observe how a Web-based banking system behaves when one hundred Vusers simultaneously check account information. To accomplish this, you create 100 Vusers, and each Vuser:

1 enters a User ID number into the system

2 enters the PIN number for the account

3 request a list of recent transactions for the account

4 checks the balance of the account

5 repeats the process numerous times

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 Web application.

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 in the Web browser.

Rendezvous points

You insert *rendezvous points* into Vuser scripts to emulate heavy user load on the application. *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 application, you can insert a rendezvous point instructing 100 Vusers to transfer funds within their accounts at the same time.

Controller

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

Hosts

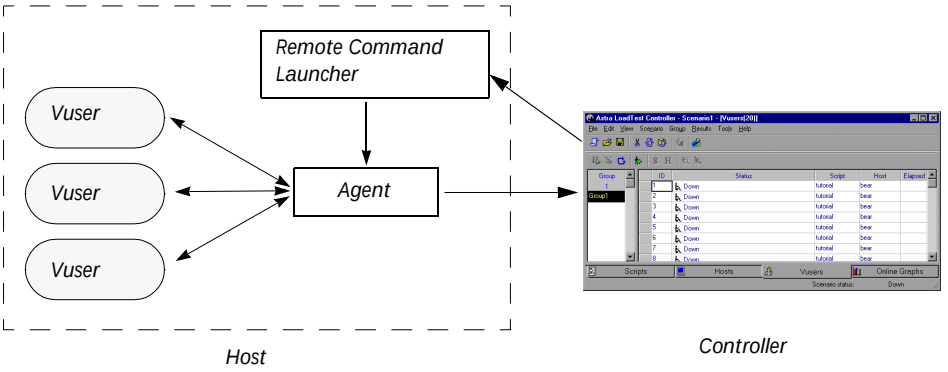
When you execute a scenario, the Astra LoadTest Controller distributes each Vuser in the scenario to a *host*. The host is the machine that executes the Vuser script, enabling the Vuser to emulate the actions of a human user.

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.

Astra LoadTest Vuser Technology

On each Windows host, you install a *Remote Command Launcher* and an *Agent*.



Remote Command Launcher

The Remote Command Launcher enables the Controller to start applications on the host machine.

Agent

The Agent enables the Controller and the host to communicate with each other. When you run a scenario, the Controller instructs the *Remote Command Launcher* to launch the Astra LoadTest *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.

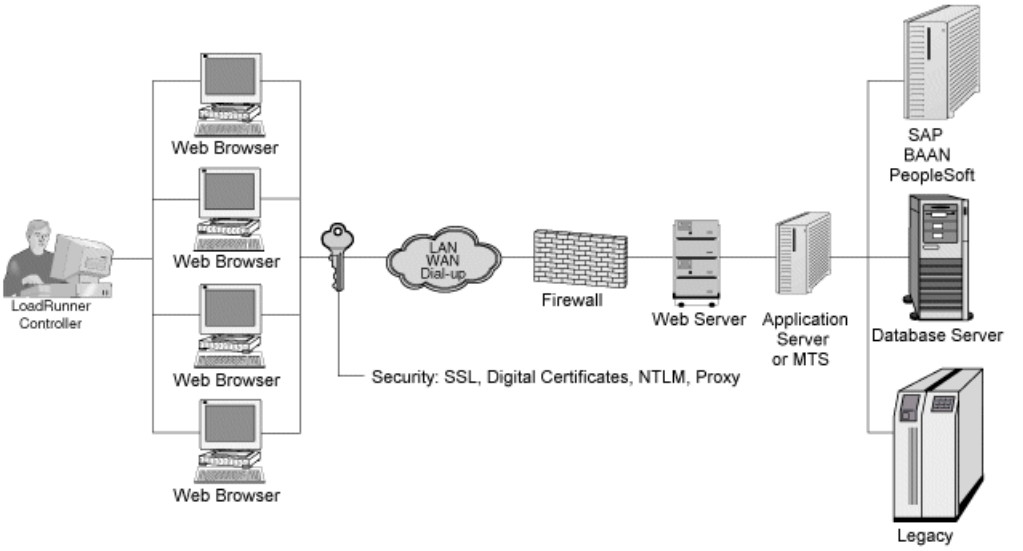
Working with Astra LoadTest

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 using Astra LoadTest, 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 Web server, you want to:

➤ emulate conditions of controlled load on the server

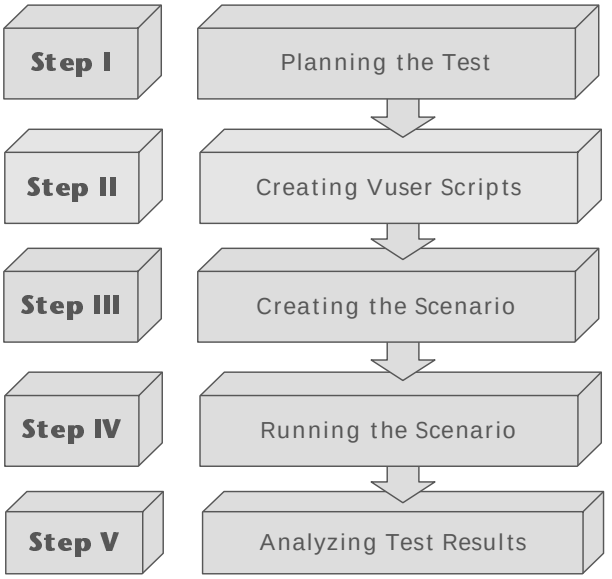
- emulate conditions of maximum load on the server
- measure server performance under load
- check where performance delays occur: network or client delays, CPU performance, I/O delays, database locking, or other issues at the server
- monitor the network and server resources under load



2

The Astra LoadTest Testing Process

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



This chapter gives you an overview of Astra LoadTest's five–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 Astra LoadTest 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 *Virtual User Recorder* guide.

Step III: Creating the Scenario

A scenario describes the events that occur during a Web-based application testing session. A scenario includes a list of machines that "host" Vusers; a list of Vuser scripts that the Vusers run; and a list of Vusers that run during the scenario. You create scenarios using the Astra LoadTest Controller. For an introduction to the Controller, see Chapter 4, "The Astra LoadTest Controller at a Glance."

Creating the List of Hosts to Run Vusers

For each scenario, you create a list of hosts—machines configured to execute Vuser scripts. For more information, see Chapter 5, "Creating a Scenario."

Creating the List of Vuser Scripts

For each scenario, you create a list of scripts that Vusers run during scenario execution. For more information, see Chapter 5, “Creating a Scenario.”

Creating the Vusers

To each Vuser in a scenario, you assign a Vuser script and a host to run the script. For more information, see Chapter 5, “Creating a 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 6, “Using Rendezvous Points.”

Before you run a scenario, you set the scenario configuration. This determines how all the hosts and Vusers behave when you run the scenario. For more information, see Chapter 7, “Configuring a Scenario.”

You can run the entire scenario, individual Vusers, or groups of Vusers (Vuser Groups). While a scenario runs, Astra LoadTest 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 Chapter 10, “Running a Scenario.”

Step V: Analyzing Test Results

During scenario execution, Astra LoadTest records the performance of the Web-based application under different loads. You use Astra LoadTest’s graphs and reports to analyze the application’s performance. For more information about Astra LoadTest’s reports and graphs, see Chapter 14, “Understanding Astra LoadTest Analysis.”

3

Load Test Planning

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

This chapter introduces the load test planning process:

- Analyzing the Web-based application
- Defining Testing Objectives
- Planning Astra LoadTest 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 Web-based application under typical working conditions, and checking for system performance, reliability, capacity, etc.

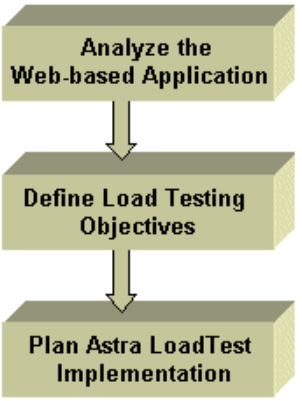
- Understand which resources are required for testing.

  Load 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:



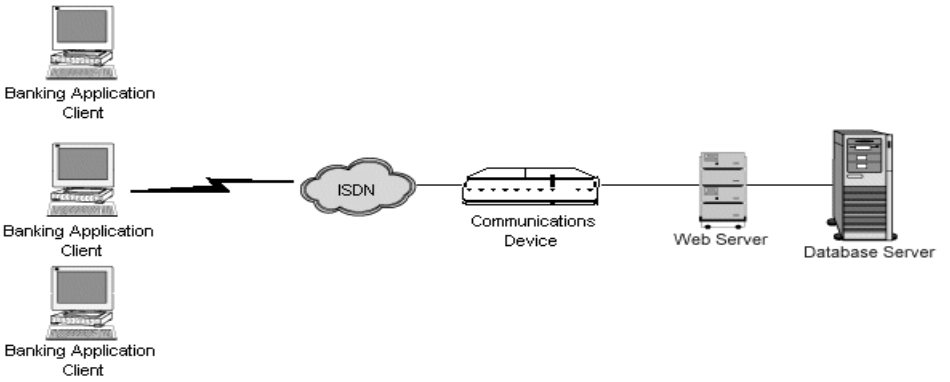
Analyzing the Web-based Application

The first step to load test planning is analyzing the Web-based 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 Astra LoadTest will accurately reflect the environment and configuration of the system under test.

Identifying System Components

Draw a schematic diagram to illustrate the structure of the Web-based application's architecture. If possible, extract a schematic diagram from existing documentation. If the system under test is part of a larger network system, you should identify the components of the system to be tested. Make sure the diagram includes all system components, such as application clients (GUI), source systems, databases, firewalls, 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 the system interface, the multiple browsers and modems of the application clients, and each client's common tasks. In addition, consider any background load that might affect the system response time.

For example, suppose between 9:00 - 9:15 each business day, 200 application clients log on to check their account balance. During the same time period, 50 application clients log on to transfer funds between accounts. You could create an Astra LoadTest scenario with 250 virtual users signing in to the Web-based application, 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 application clients performing various account transactions.

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 Web 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 Web-based application testing objectives that Astra LoadTest 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 Astra LoadTest Implementation

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

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 a browse sequence that a typical end-user would follow. 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.

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 Astra LoadTest Controller on a separate machine.
- Configuration of the test machine for Vusers should be as similar as possible to the actual user's machine.

Refer to the following table to estimate the required hardware for each Astra LoadTest testing component. These requirements are for optimal performance.

Windows Configuration Requirements

Requirement	Controller	Web Vuser
Hardware	Pentium 90 or higher	Pentium 120 Mhz or higher recommended for running large numbers of users (more than 30).
Memory	24 MB or more on Controller machine	At least 1 MB per non-multithreaded Vuser or at least 2.5 MB for 10 users in multithreaded mode (depends on application)
Swap Space	Two times the total physical memory	Two times the physical memory for multithreaded mode or four times the physical memory for non-multithreaded mode
Software	tcp/ip support	HTTP support
1 CPU* supports *x* users	n/a	~250 in multithreaded mode

Note: The results file requires a few MB of disk space for a long scenario run with many transactions. The host machines also require a few MB of disk space for temporary files if there is no NFS. See Chapter 7, "Configuring a Scenario" for more information about run-time file storage.

* Pentium Pro 200 MHz may have multiple CPUs per machine.

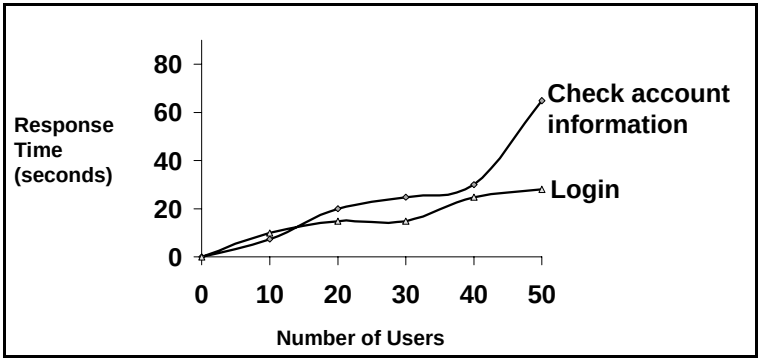
Examining Load Testing Objectives

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 application client to perform a business process and receive a response from the Web server. For example, suppose that you want to verify that while your system operates under normal load conditions, the application client receives 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 Astra LoadTest 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. 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.

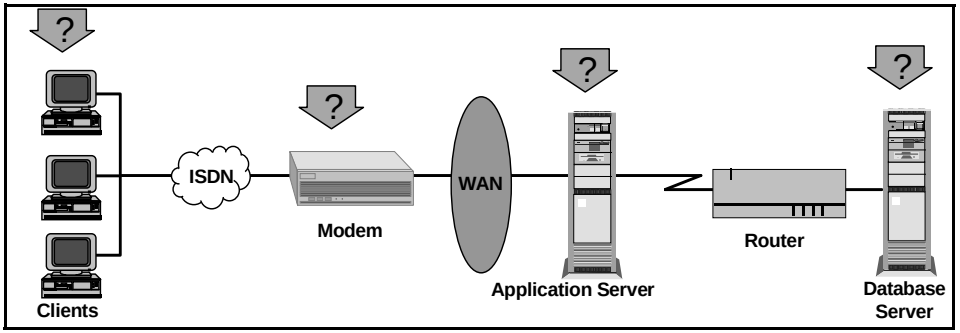
Evaluating New Products

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 Web server machine or the database package based on evaluation tests.

Identifying Bottlenecks

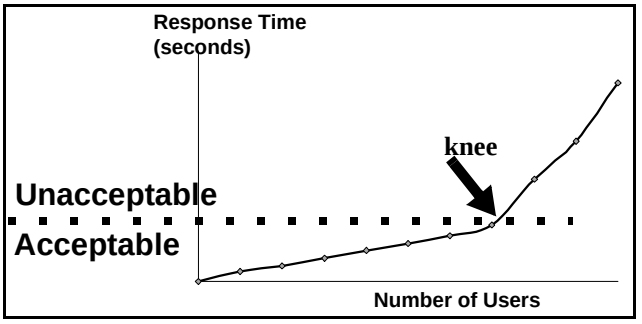
You can run tests which identify bottlenecks on the system to determine which element is causing performance degradation, for example, file locking, resource contention and network overload. Use Astra LoadTest in conjunction with the new network and host monitoring tools to create load

and measure performance at different points in the system. For more information, see Part IV of this manual, *Monitoring Resources*.



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 Astra LoadTest 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 Astra LoadTest Controller
- Selecting Astra LoadTest Commands
- Managing Scenario Files
- Filtering and Sorting Information

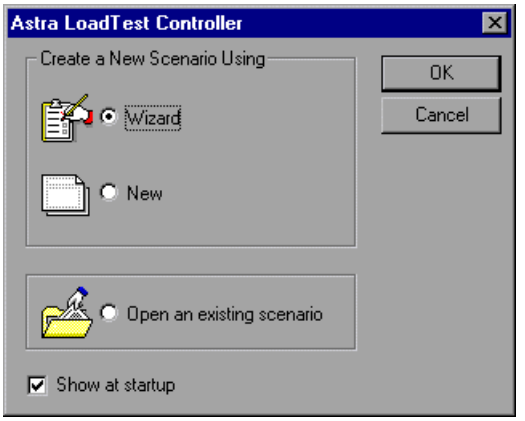
Opening the Controller

Set up the Astra LoadTest environment according to the instructions in the *Installing Astra LoadTest guide.*

To open the Controller:

Select **Start > Programs > Astra LoadTest > Products > Controller**.

The Controller window opens with the Astra LoadTest Controller dialog box inside.



The Astra LoadTest Controller dialog box includes the following options:

Wizard: Creates a scenario with the aid of the Astra LoadTest Scenario wizard, an interactive, step-by-step guide to creating a scenario.

New: Creates a new scenario.

Open an existing scenario: Opens an existing scenario.

Show at Startup: If you do not want this window to appear the next time you start Astra LoadTest, clear the "Show at Startup" check box. Once you clear this checkbox, when you open the Controller, the window layout from the last time you worked with the Controller appears.

If you want to display this dialog box when the Controller starts, select **View** > **Show Startup Dialog** from the Controller's main menu.

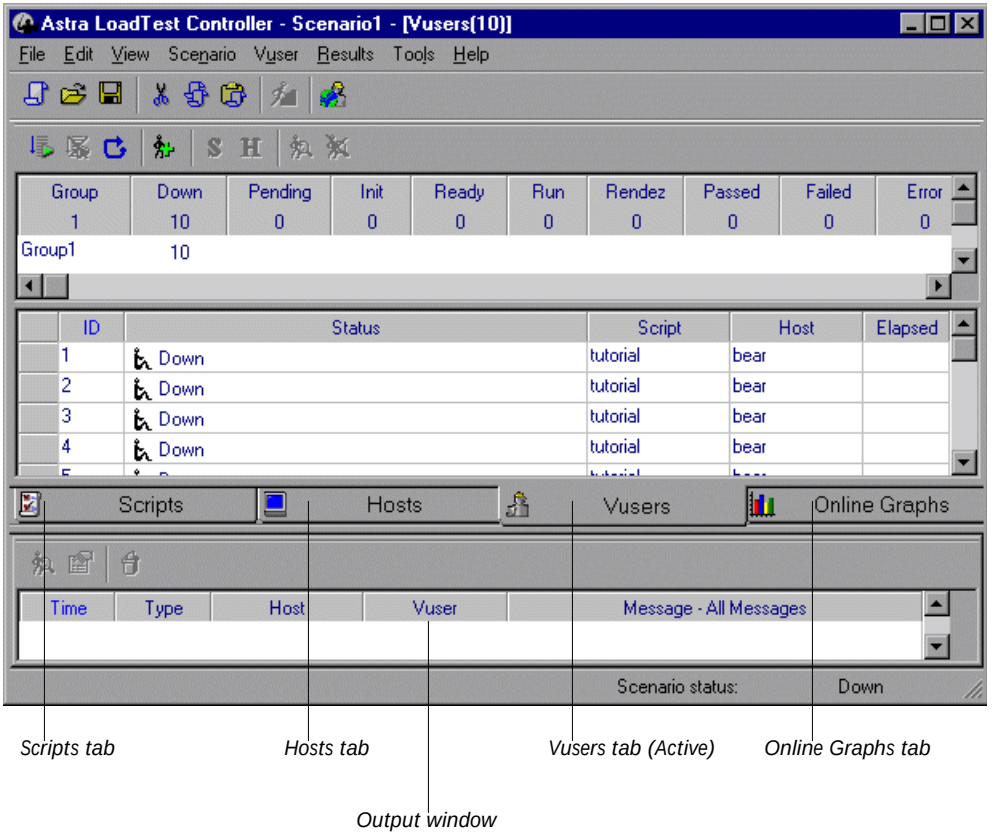
Introducing the Astra LoadTest Controller

The Astra LoadTest 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.

Tool bar Provides shortcuts for selecting commands. Clicking on a button executes a command.

Status bar Provides information on the current state of the scenario.



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

Scripts view Displays a list of all the Vuser scripts that you can assign to Vusers.

Hosts view Displays the list of machines that can execute Vuser scripts during the scenario.

Vusers view	Displays the Vusers assigned to the scenario (default view).
Online Monitor view	Displays online monitor graphs showing transactions and server resource information.

In addition, the Controller opens the **Output** window which displays error and notification messages generated by Vusers during scenario execution.

Selecting Astra LoadTest Commands

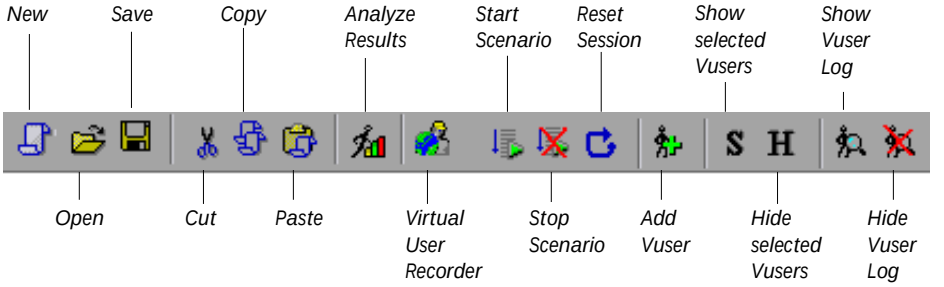
You select Astra LoadTest commands from the menu bar or on the toolbar.

Choosing Commands from a Menu

You select Astra LoadTest commands from the menus on the menu bar. Some commands use standard Microsoft Windows shortcut conventions.

Choosing Commands from the Toolbar

You can execute many Astra LoadTest commands by clicking a button on the toolbar in the Astra LoadTest Controller.



Managing Scenario Files

A scenario describes the events that occur during each load testing session. A scenario contains lists of hosts, Vusers, Vuser tests, transactions, and rendezvous points. You create a scenario using the Astra LoadTestController.

After you create the scenario, Astra LoadTest 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

You record tests for your scenario with the Virtual User Recorder. You can automatically include a recorded test in a new scenario by opening the Controller from the Virtual User Recorder. Click the **Controller** button from the Virtual User tool bar.

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.

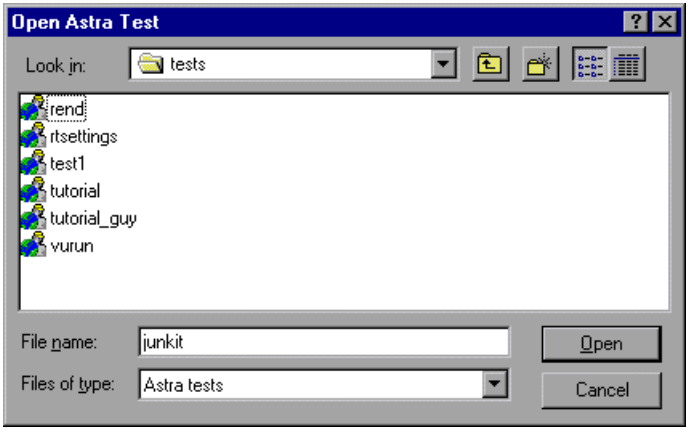
You can also create a new scenario by using the Scenario wizard. The wizard is an interactive, step-by-step guide to creating a scenario. To create a new scenario by using the Scenario wizard, select **Tools** > **Scenario Wizard**.

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 File Open dialog box opens.



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

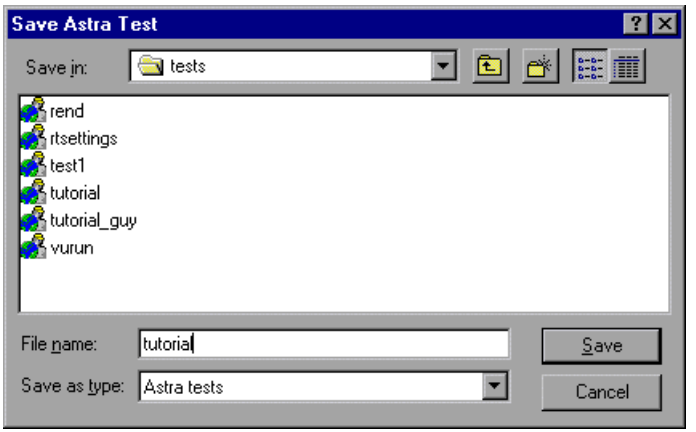
3 Click **OK**. The File Open dialog box closes and the scenario appears in the Astra LoadTest Controller.

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 As 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 **OK**. 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.

Filtering and Sorting Information

Each window in the Astra LoadTest Controller displays information about the scenario. You can filter and sort the information that appears in each window.

Filtering information displays only those items that meet the selected criteria. For example, you can filter the Vuser window to display only those Vusers that are in the READY state.

Sorting information displays all the items in a list in a certain order. For example, you can sort all Vusers in the Vuser list, in order of their Vuser ID number (1,2,3 etc.).

This section describes how to filter and sort the information displayed in the Vuser view.

Filtering Information

You can filter the Vusers according to their status or show all Vusers. In the following example, the filter shows all the Vusers with an ERROR status.

ID	Status	Script	Host	Elapse
2*	Error... The Vuser script can not run on this host pl	test1	doc9pc	
6*	Error... The Vuser script can not run on this host pl	test1	goose	
7*	Error... The Vuser script can not run on this host pl	test1	doc9pc	
8*	Error... The Vuser script can not run on this host pl	test1	doc9pc	
9*	Error... The Vuser script can not run on this host pl	test1	doc9pc	
10*	Error... The Vuser script can not run on this host pl	test1	doc9pc	

To filter information:

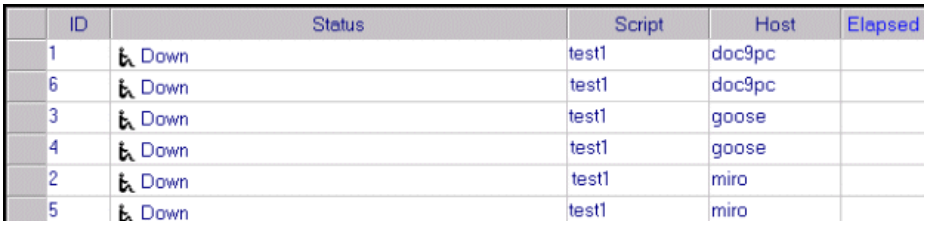
1 Choose **Vuser** > **Filter Vusers.** The menu expands to a list of filter criteria.

2 Select a filter criteria. The Vuser list is filtered.

Sorting Information

You can sort the Vusers according to their ID, status, script, host, or elapsed time (all of the columns shown in the Vuser view). By default the Vusers are

sorted by their ID numbers. In the following example, the Vusers are sorted by host.



ID	Status	Script	Host	Elapsed
1	Down	test1	doc9pc	
6	Down	test1	doc9pc	
3	Down	test1	goose	
4	Down	test1	goose	
2	Down	test1	miro	
5	Down	test1	miro	

To sort information:

1 Choose **Vuser** > **Sort Vusers.** The menu expands to a list of sort criteria.

2 Select a sort criteria. The Vuser list is sorted.

3 Alternatively, you can sort Vusers by clicking on a column heading.

Part II

Building a Scenario

5

Creating a Scenario

To build a scenario, you create a host list, a script list and a Vuser list. This chapter describes how to add these items to a scenario.

This chapter discusses:

- Adding Hosts to the Host List
- Adding Scripts to the Scripts List
- Using Relative Paths for Vuser Scripts
- Creating the Vuser List
- Adding Vusers to the Vuser List

About Creating a Scenario

To test your system with Astra LoadTest 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 number of virtual users (Vusers), the test scripts these users will run, and the host machines upon which to run the scripts.

To create a scenario effectively, you first create a list of available host machines and scripts. Then you create Vusers or Vuser Groups (collections of Vusers performing similar tasks) and assign them a script and host machine.

The Controller displays the host, script, and Vuser lists. To view one of the lists, you click on the appropriate tab.

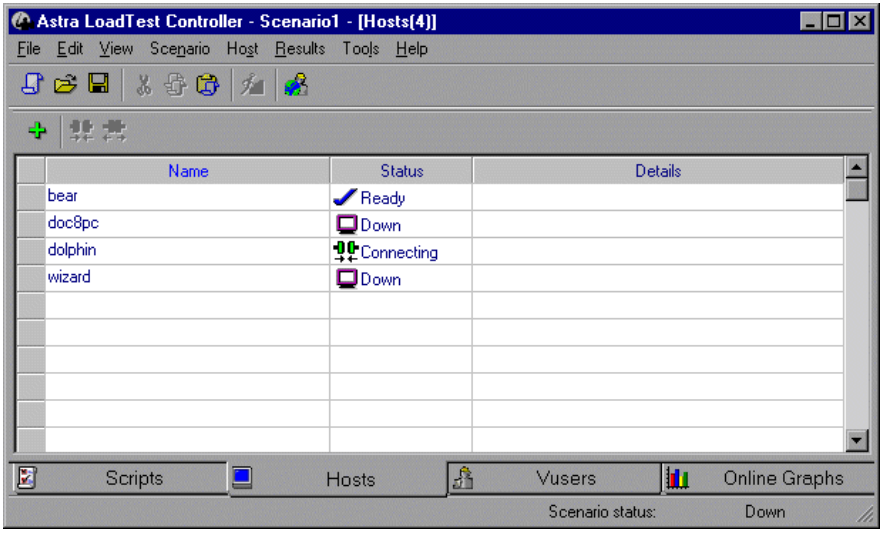
The *host* list—the list of machines that execute Vuser scripts during a scenario run— displays the machines that can run Vusers during scenario execution.

A script defines the actions that a Vuser performs during scenario execution. The *script* list contains all the Vuser scripts that Vusers can run.

The *Vuser* list contains all of the Vusers defined in the scenario, with the script and host that were assigned to them.

Adding Hosts to the Host List

The Host view has its own menu in the Controller menu bar, and a toolbar that lets you create and manage the host list.



For each host in the list, the Controller displays the following information:

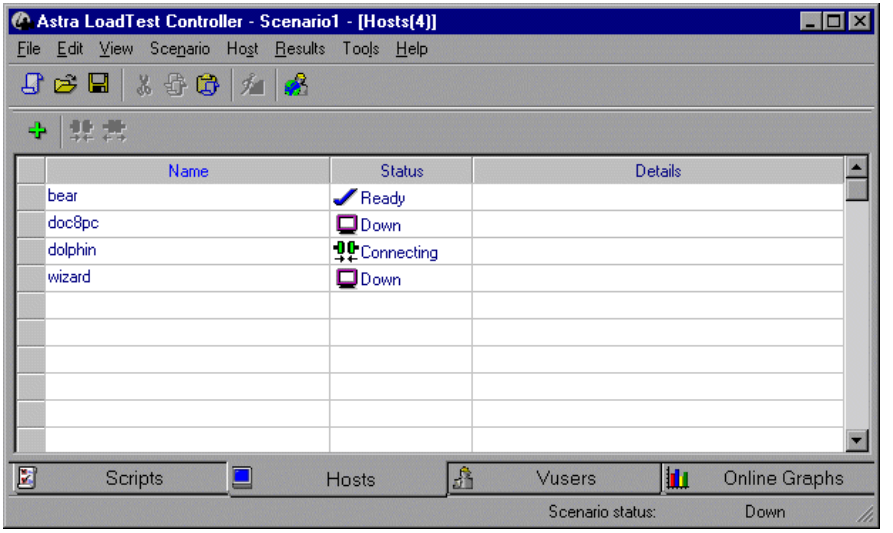
- Name
- Host status
- Details

You add machines to the host list from the Hosts view of the Controller window.

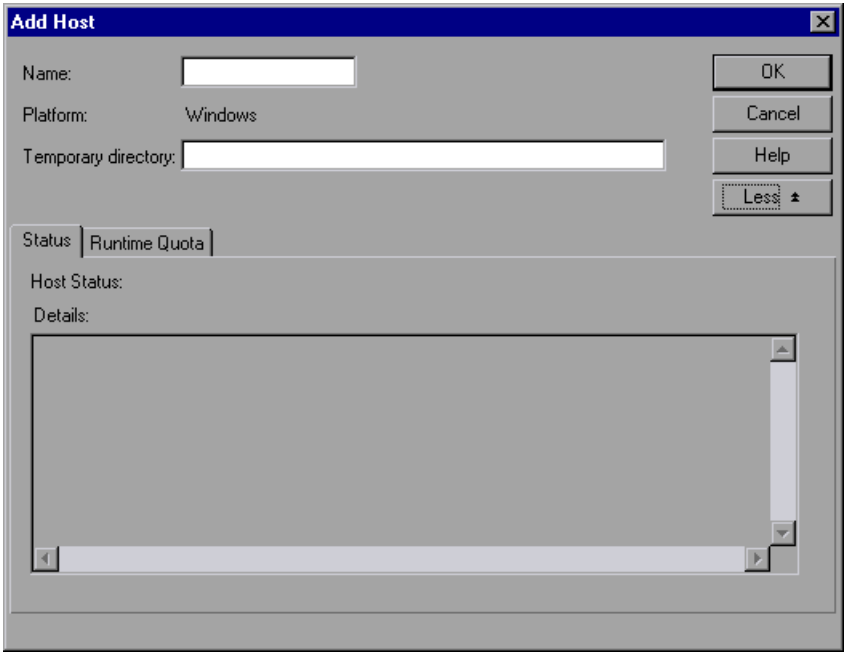
After you create a host, you can modify its attributes and include or exclude it from the scenario. For details, see Chapter 8, “Configuring a Host”.

To add a host to the Host list:

1 Start the Astra LoadTest Controller and click on the **Hosts** tab to open the Hosts view. The Host menu appears in the Astra LoadTest menu bar.



2 Choose **Host** > **Add**. The Add Host dialog box opens.



3 Type the name of the host in the **Name** box.

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

5 Click **More** to expand the dialog box and show the Host tabs. For information on configuring host settings, such as Vuser quotas and display attributes, see Chapter 8, "Configuring a Host."

6 Click **OK** to close the dialog box. The host name you entered appears in the Host list; its status is set to DOWN.

You can configure specific settings for your host such from the Host Information dialog box. For more information about these settings,

Saving and Loading the Default Host List

After you create a host list, you can save it as the standard (default) collection of hosts for creating scenarios. When you create a new scenario, you can load the default host list and adapt it to your specific needs.

To save the host list as the default:

➤ Choose **Host** > **Save List As Default**.

To load the default host list:

➤ Choose **Host** > **Load Default List**. The default host list appears in the Host window, together with any host previously displayed.

Adding Scripts to the Scripts List

The Scripts view, which displays the script list, has its own menu in the Controller menu bar. It includes the following information for each script in the list:

➤ the name of the Vuser script

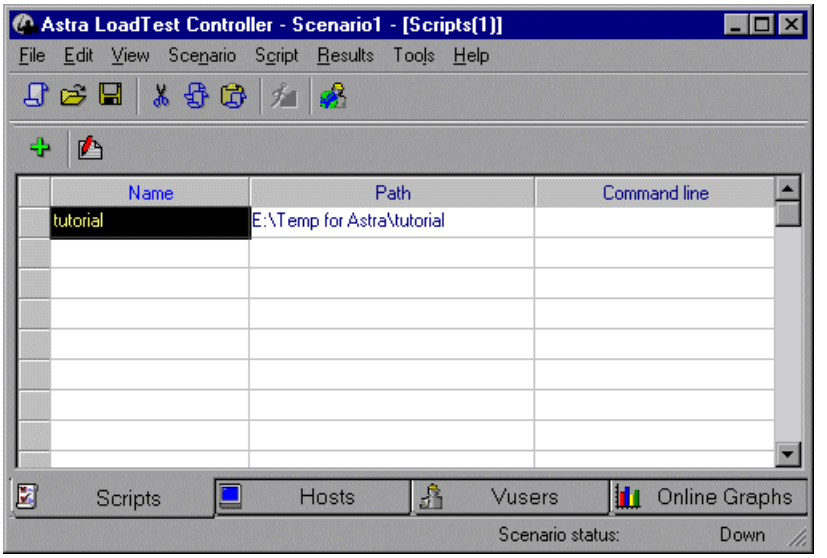
➤ the location (path) of the script

➤ command line options associated with the script

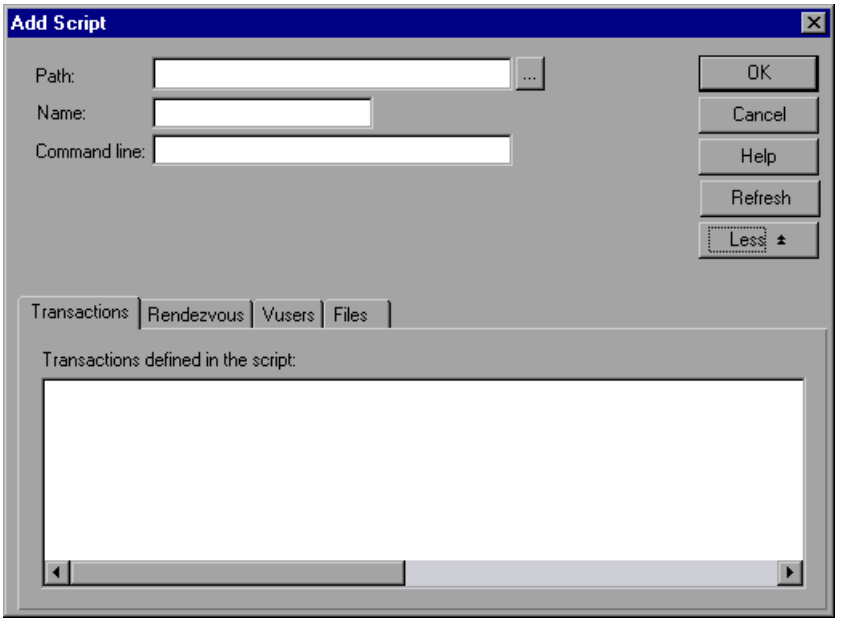
You use the Controller's Script view to add scripts to the script list. After you create a script, you can set and modify its details.

To add a script to the script list:

1 Click the **Scripts** tab in the Astra LoadTestController to open the Scripts view. The **Script** menu appears in the Astra LoadTest menu bar.



2 Choose **Script** > **Add** or click the **Add** button. The Vuser Script Information dialog box opens.



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

4 In the **Files of Type** box select the Vuser type, and then select the path and file name of the new script.

Note: When you specify the location of a Vuser script, you can specify a location that is relative to the current scenario directory. For details, see "Using Relative Paths for Vuser Scripts," on page 55.

---

5 Click **Open** to select the files. The Open Test dialog box closes, and the new script name and its Vuser type appear in the Vuser Script Information dialog box.

6 In the **Name** box, type a name for the script. By default, Astra LoadTest uses the script's file name without an extension.

7 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 *Virtual User Recorder* guide.

8 Click **More** to show the Script tabs.

- ➤ To see the transactions declared in the selected script, click the **Transaction** tab.
- ➤ To see the rendezvous points declared in the selected script, click the **Rendezvous** tab.
- ➤ 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.
- ➤ 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 exclude a file from the list, click on the check box adjacent to it. To add a file to the list, click **Add**.

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

The new script name and its details appear in the Script view. To modify the properties of an existing script, select it from the Scripts list and choose **Script** > **Details**. Enter the changes in the Vuser Script Information dialog box.

Note: If you use the Virtual User Recorder to make changes to a Vuser script while the Controller is running, click the **Refresh** button in the Controller's Scripts view to update the script details in the scenario.

Using Relative Paths for Vuser Scripts

When you specify the location of a Vuser script, you can specify a relative location. The location can be relative to the current scenario directory, or the Astra LoadTest 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 Astra LoadTest installation directory by typing a percent sign (%) at the beginning of the script path. For example, if the Astra LoadTest installation directory is located at F:\LoadTest, to specify a script located at F:\LoadTest\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 machine. This enables the Vuser host to access the script locally instead of over a network.

Creating the Vuser List

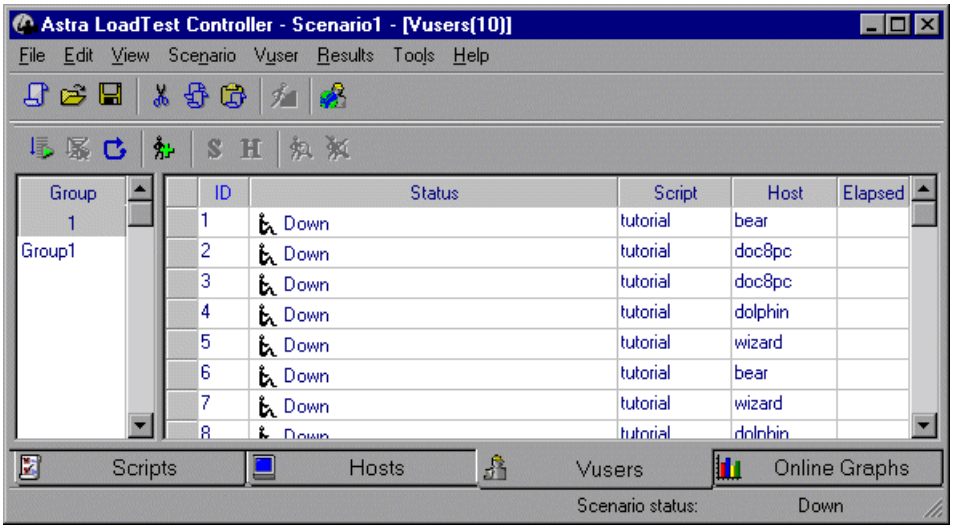
Vusers emulate human users interacting with your Web-based system. Each scenario contains a list of many virtual users (Vusers). When you run a scenario, the Vusers generate load on the server, and Astra LoadTest monitors the server and transaction performance.

For example, to test an online bank you could create Vusers that deposit and withdraw cash using automatic teller machines (ATMs). Astra LoadTest monitors the bank server's performance while a varying numbers of customers simultaneously perform deposit and withdrawal operations.

Vuser Groups

You create Vuser Groups to organize the Vusers in a scenario into manageable groups. The Groups should contain Vusers with shared or similar characteristics. For example, you can create a Vuser Group for all Vusers that run the same Vuser script.

In the following example, the Vusers in Group G2 will run the same script, *gui_test1*, but on different host machines.



Each Group must contain a list of Vusers. You can create a Vuser list by:

➤ Creating a Group and specifying the number of Vusers in the Group.

➤ Creating Vusers and adding them to an existing Group.

The recommended procedure for creating a Vuser list is to create a Vuser Group containing Vusers with similar attributes. At any point, you can assign more Vusers to the Group.

Renumbering Vusers

While controlling a scenario, you may have deleted some Vusers from a given Vuser Group. After deleting the Vusers, the Vuser ID numbers in that group are no longer consecutive. The Astra LoadTest Controller enables you to renumber the Vusers within a Vuser Group.

Note: When you renumber Vusers, the Vusers are assigned new ID numbers. This may conflict with previous results.

To renumber the Vusers within a Vuser Group:

1 Select the Vuser Group that you want to renumber.

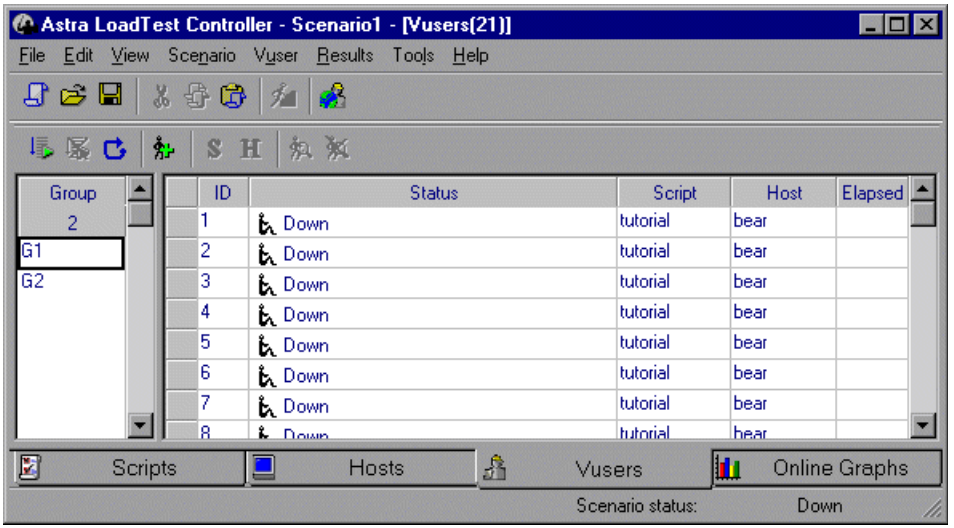
2 Choose **Group** > **Renumber**. The Vusers in the selected group are renumbered from 1.

Adding Vusers to the Vuser List

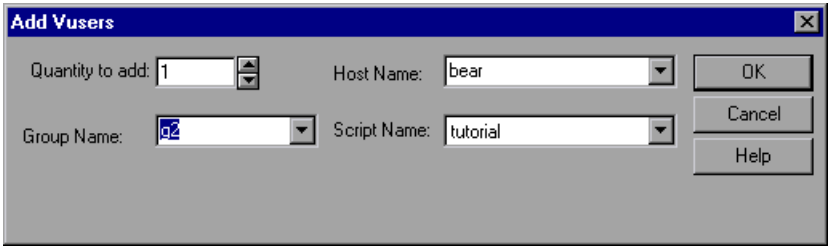
You can create one or multiple Vusers and assign them to any Vuser Group. When you create a Vuser, you assign it a script and a host. You can assign the same script to many Vusers.

To create Vusers:

1 Click the **Vusers** tab in the Astra LoadTest Controller to open the Vusers view. The **Vuser** menu appears in the Astra LoadTest menu bar.



2 Choose **Vuser** > **Add Vuser** or click the **Add Vuser** button. The Add Vusers dialog box appears.



3 In the **Quantity to Add** box, enter the number of Vusers that you want to create.

4 Select a Group from the **Group Name** list for the new Vuser. If no Group exists, type a name in the **Group Name** box.

5 Select a host from the **Host Name** list. The Host list contains all hosts that you previously added to scenario. To use a host that does not appear, select **Add...** from the **Host Name** list to open the Add Host dialog box.

6 Select a script from the **Script Name** list. The Script list contains all scripts that you previously added to the scenario. To use a script that does not appear, select **Add...** from the **Script Name** list to open the Add Script dialog box.

7 Click **More** to expand the dialog box and view the Vuser status. Specify the appropriate scheduling settings for the group.

8 Click **OK** to close the Add Vusers dialog box. The new Vusers appear in the Vuser window. Astra LoadTest assigns unique ID numbers to the Vusers. If you did not create a Vuser Group, Astra LoadTest creates the Vuser Group G1 and assigns the Vusers to it.

You can modify the settings for an existing Vuser, by selecting the Vuser and choosing **Vuser** > **Details** or by clicking the **Details** button.

6

Using Rendezvous Points

Astra LoadTest 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 Astra LoadTest to measure server performance under load.

Suppose you want to measure how a bank server performs when ten Vusers simultaneously withdraw cash from ten different automatic teller machines (ATMs). In order to emulate the required user load on the server, you instruct all the Vusers to withdraw cash 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 64.

You define rendezvous point in the Vuser script. For information about inserting rendezvous points into Vuser scripts, refer to the *Virtual User Recorder* 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 an online bank 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 to a scenario, Astra LoadTest scans the associated Vuser script for the names of the rendezvous points and adds them to the list in the Rendezvous window. If you create another Vuser that runs the same script, the Controller adds the new Vuser 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).**

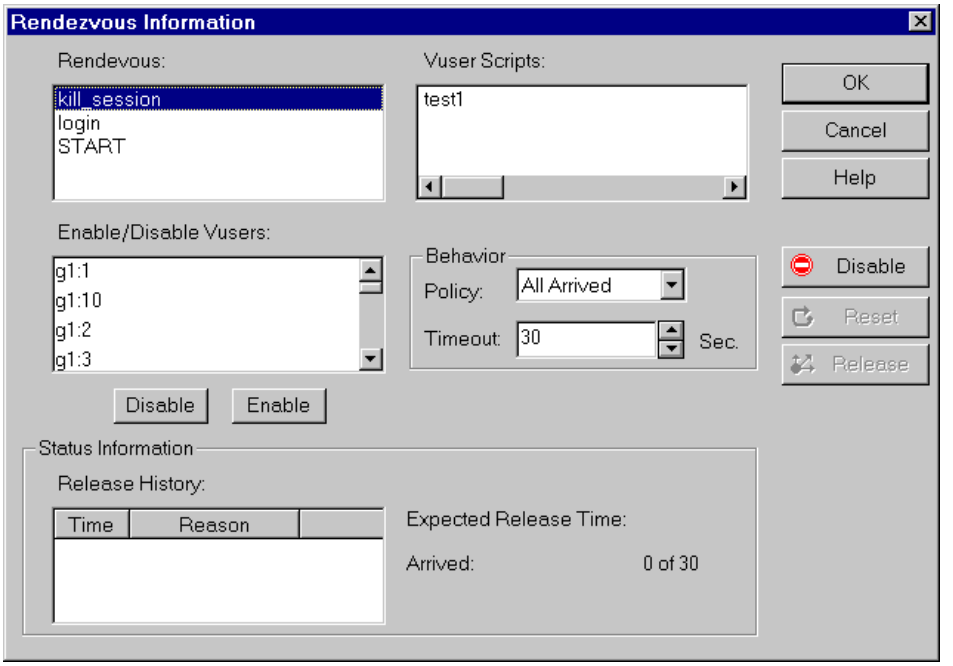
   For each rendezvous you can set the *timeout* and *policy attributes*. For more information, see "Setting the Rendezvous Behavior," on page 64.
5. **Run the scenario.**

Setting the Rendezvous Attributes

You can set the following rendezvous attributes from the Rendezvous Information dialog box:

- 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 and Reset commands, see Chapter 10, “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

The *timeout* determines the maximum time (in seconds) that Astra LoadTest waits for each Vuser to arrive at a rendezvous. After each Vuser arrives at the rendezvous, Astra LoadTest waits up to *timeout* seconds 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 From the Rendezvous list, select the rendezvous for which you want to set a timeout.

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 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:

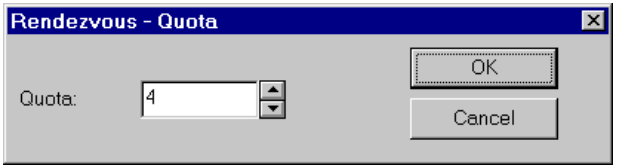
1 Choose **Scenario** > **Rendezvous**. The Rendezvous Information dialog box opens.

2 Select a rendezvous from 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 appears.



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 on 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.

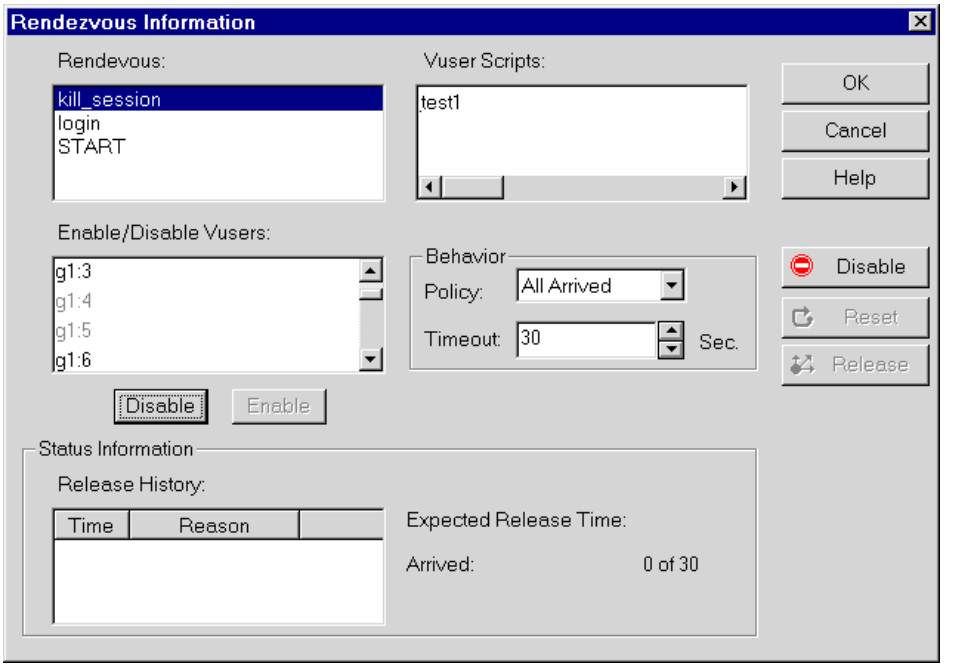
Disabling and Enabling Vusers at Rendezvous Points

In addition to disabling a rendezvous for all Vusers in a scenario. Astra LoadTest 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 from the Rendezvous list for which you want to disable Vusers.



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*.

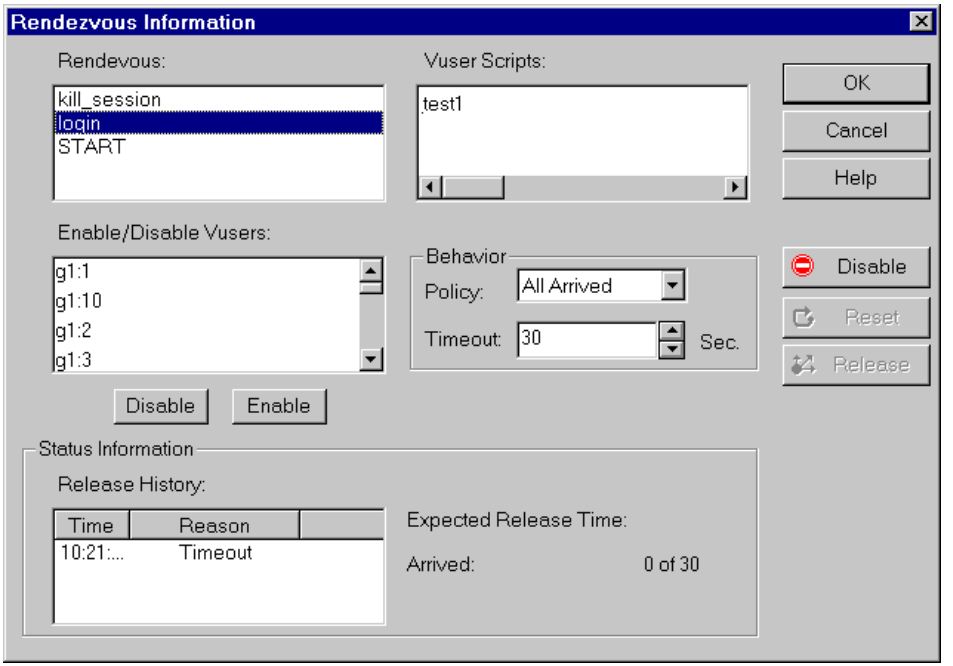
Expected Release Time: The time you expect the Vusers to be released from the scenario.

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.



7

Configuring a Scenario

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

This chapter describes:

- Saving Messages to the Output Files
- Setting Timeout Intervals

About Configuring a Scenario

Before you run a scenario, you can configure both the host and Vuser behaviors for the scenario. The default settings correspond to most environments. Astra LoadTest 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 hosts in a scenario. To change the settings for individual host machines refer to Chapter 8, “Configuring a Host.” If the global scenario settings differ from those of an individual host, the host settings override them.

The settings discussed in this chapter are unrelated to the Vuser Runtime settings. These settings, which apply to individual Vusers or scripts, contain information about logging, think time, and the number of iterations.

For information on setting the options for online monitors, see Chapter 12, “Online Monitoring.”

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

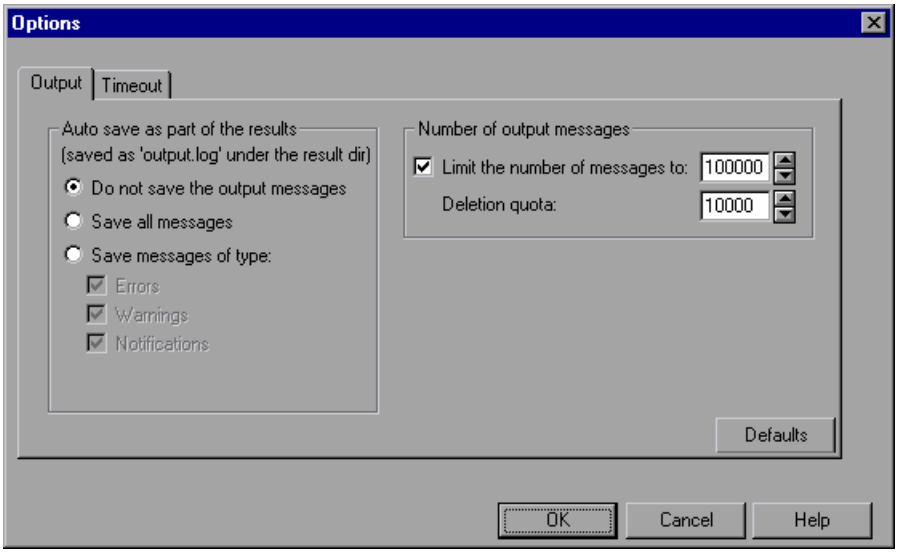
Saving Messages to the Output Files

By default, Astra LoadTest 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 instruct Astra LoadTest which 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, they 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 file:

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



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

3 Select **Save all messages** to instruct Astra LoadTest 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.

Setting Timeout Intervals

Astra LoadTest enables you to set the timeout interval for commands, pinging of hosts, and Vuser elapsed time.

The command timeouts are the maximum time limits for various Astra LoadTest commands. When a command is issued by the Controller, you set a maximum time for the host 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 hosts and Vusers. The host commands for which you can specify a timeout interval are Connect, Disconnect, and Heartbeat. The Vuser commands for which you can specify a timeout interval are Init, Run, Pause, Abort.

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

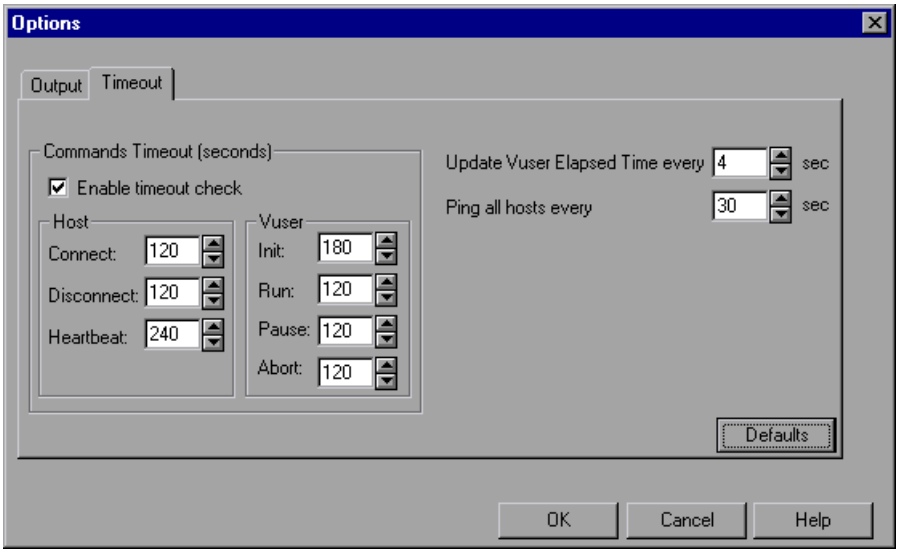
In the Vuser view, the Elapsed column (the last column) indicates the amount of time that elapsed from the beginning of the scenario. You can specify the frequency in which Astra LoadTest updates this value. The default is 4 seconds.

During scenario execution, Astra LoadTest periodically pings all of the hosts participating in the scenario to ensure that a connection still exists. You can set the ping frequency —the default is 30 seconds.

Note: Astra LoadTest's calculations consider the number of active Vuser and their influence on the timeout values. For example, 1000 Vusers trying to initialize will take much longer than 10 Vusers. Astra LoadTest 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 appears. Click the **Timeout** tab.



2 Clear the **Enable timeout checks** check box to disable the timeout monitoring. Astra LoadTest 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 checks** check box and specify the appropriate timeouts.

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

5 Enter the frequency at which Astra LoadTest pings the various host machines participating in the scenario in the **Ping all hosts every** box.

Additional timeout related settings are available through Expert mode. For more information, see Chapter , “Working in Expert Mode.”

8

Configuring a Host

After you add hosts to the Controller's Hosts list, you can indicate which hosts to use for each scenario and configure their properties.

This chapter describes:

- Selecting the Hosts for a Scenario
- Setting the Runtime Quota

About Configuring a Host

You can set a host's attributes while adding it to the host list. You can modify the attributes of an existing host at any time using the Host Information dialog box.

To configure global settings for all hosts participating in the scenario, use Astra LoadTest's **Settings** dialog box. For more information, see Chapter 7, "Configuring a Scenario." To set properties specific for each host, use the **Host Information** dialog as described in this chapter.

The settings that can be configured both globally and individually for each host are: Runtime File Storage, and Vuser Quota.

In addition to configuring hosts, you can disable one or more hosts for a scenario. This is particularly useful if you want to isolate a specific machine to test its performance.

Selecting the Hosts for a Scenario

Before you run a scenario, you can indicate which hosts will run Vusers in the scenario. For example, if a host is unavailable for a particular scenario run, you can exclude it temporarily instead of removing it entirely from your list of hosts.

You select which hosts will take part in the scenario by using the Enable and Disable commands. Disabling a host temporarily removes it from the list. Enabling a host reinstates it.

Disabling Hosts

When you disable a host, the host will not execute Vuser scripts during a scenario execution.

To disable a host:

1 Click the **Hosts** tab. The Host menu appears in the Astra LoadTest menu bar.

2 Select a host that you want to disable from the host list.

3 Choose **Host** > **Disable**. The host name changes from blue to gray. The host is disabled.

Enabling Hosts

Enabling a host reinstates a previously disabled host.

To enable a host:

1 Click the **Hosts** tab. The Host menu appears in the Astra LoadTest menu bar.

2 Select a host that you want to enable from the host list.

3 Choose **Host** > **Enable**. The host name changes from gray to blue, and the host is enabled.

Setting the Runtime Quota

The runtime quota determines how many Vusers a host can initialize or stop at a time. Initializing or stopping a large number of Vusers simultaneously places large stress on a host. To reduce stress on a host, you can initialize or stop smaller batches of Vusers.

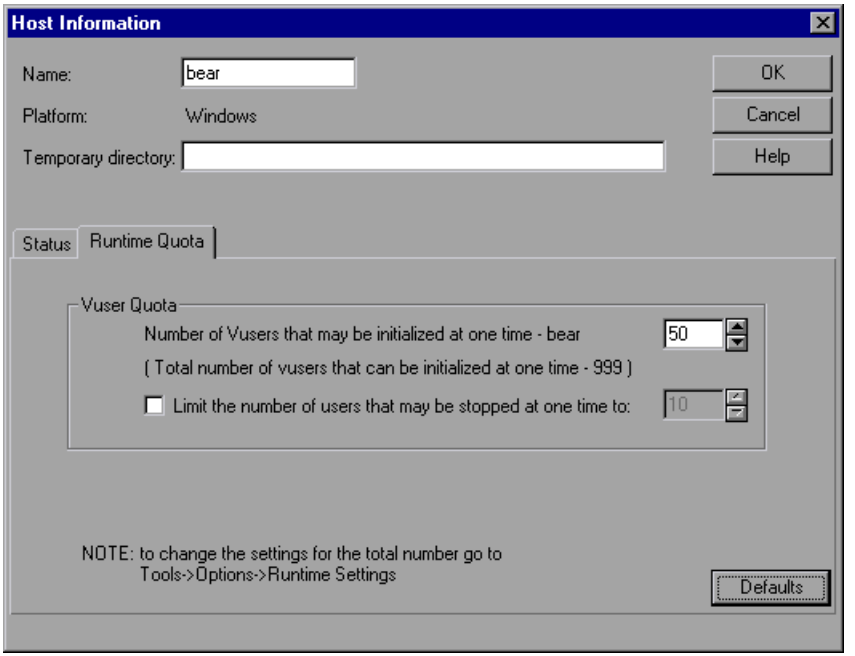
You can set runtime quotas for an entire scenario or for individual hosts. For information on setting quotas globally for an entire scenario, see Chapter 7, "Configuring a Scenario."

To set the runtime quotas for a specific host:

1 Click the **Host** tab to display a list of the available hosts. Select a Vuser host.

2 Choose **Host** > **Details** or click the **Details** button. The Host Information dialog box opens.

3 Click the **Runtime Quota** tab.



4 In the **Vuser Quota** section, enter the maximum number of Vuser types that the host will initialize or stop simultaneously.

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

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

9

Preparing to Run a Scenario

Before you run a scenario, you specify a location for the scenario results and other runtime 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 Astra LoadTest scenario.

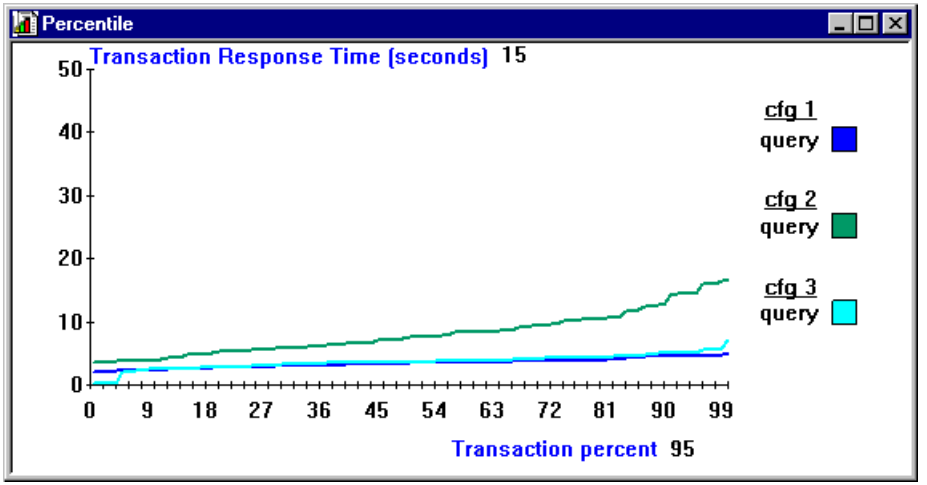
For information on one-time configuration settings such as timeout, output, and quotas, see Chapter 7, “Configuring a Scenario.”

Specifying a Results Location

When you run a scenario, by default the runtime files are stored locally on each Vuser host. After the scenario, the results are collated together and processed on the Controller machine. Alternatively, you can instruct Astra LoadTest to save the results on a shared network drive. For information about specifying a file storage method, see the Runtime File Storage settings in Chapter 7, "Configuring a Scenario."

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

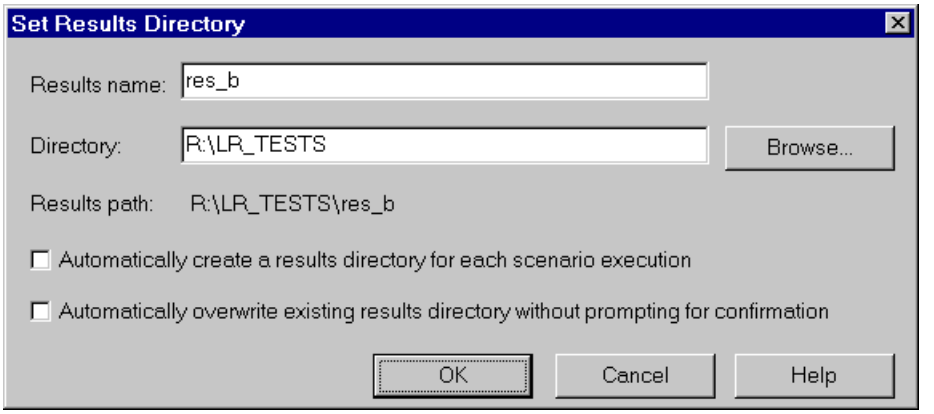
In the example below, the results of three scenario runs are superimposed. The result sets are cfg_1, cfg_2, and cfg_3.



For more details on cross-scenario graphs, see Chapter 18, "Cross-Scenario Analysis."

To specify where results are stored:

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 Vusers should write during the 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.

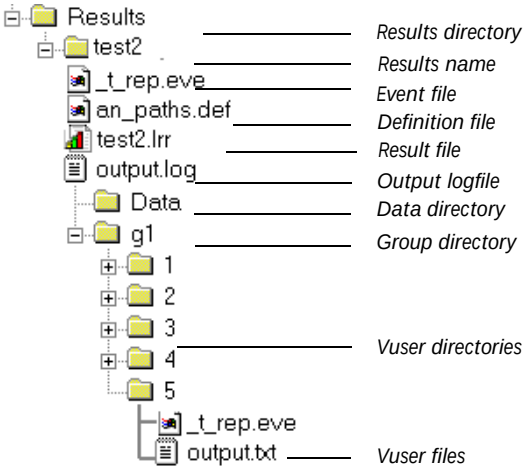
4 Select the appropriate check box for subsequent executions: **Automatically create a results directory for each scenario execution** or **Automatically overwrite existing results directory without prompting for confirmation**.

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

Results Directory File Structure

When you set the results directory, you also specify a results name. Astra LoadTest 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, Astra LoadTest also gathers data from each Vuser and stores it in an event file *_t_rep.eve* and an output file *output.txt*. Astra LoadTest 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 Astra LoadTest 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 and Data Point information.
- *output.txt* in each Vuser directory contains output information generated during replay.

When you generate analysis graphs and reports, the Astra LoadTest 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 Astra LoadTest Analysis, see Chapter 14, "Understanding Astra LoadTest Analysis."

Collating Results

When you run a scenario, by default all Vuser information is stored locally on each Vuser host. After scenario execution the results are automatically *collated* or consolidated—results from all of the hosts are transferred to the results directory. You set the location of the results directory as described in "Specifying a Results Location," on page 80.

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 Astra LoadTest Analysis 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 69 for details on changing how results are stored.

To limit access to a host machine:

You can limit access to a host by creating a list of authorized machines in the host's Remote Command Launcher (shortcut menu). Only machines which appear on the list are authorized to launch applications on the host.

1. Make sure that the Remote Command Launcher, *rmcdl.exe*, is active on the host machine.
2. Select **Edit Permissions** from the Remote Command Launcher's shortcut menu.

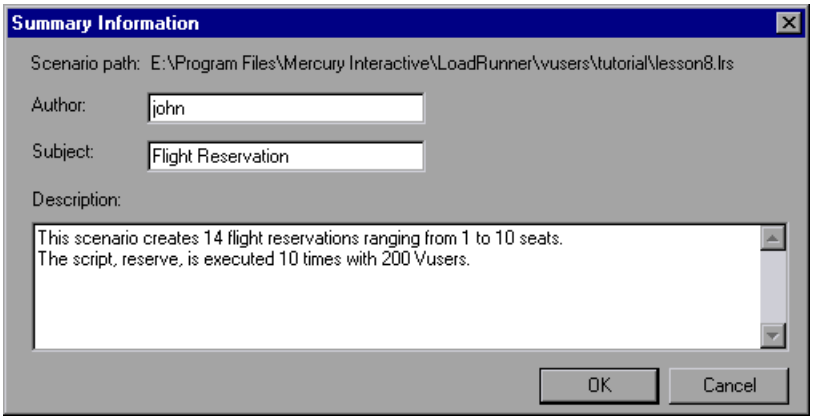
   The host list opens in Notepad.
3. Create or update the list of machines which are authorized to access this host. Place each machine name on a separate line.
4. Close the Notepad file.
5. To save the permissions list, select **Update Permissions** from the Remote Command Launcher's shortcut menu.

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.

Part III

Running a Scenario

10

Running a Scenario

When you run a scenario, Astra LoadTest generates load on the client/server system, and measures the system's performance.

This chapter describes:

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

About Running a Scenario

When you run a scenario, the Vusers are assigned to their hosts and execute their Vuser scripts. During scenario execution, Astra LoadTest:

- records the durations of the transactions you defined in the Vuser scripts
- performs the rendezvous in the rendezvous list
- collects error 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 host. When the Vusers are ready, they start executing their Vuser scripts. While the scenario runs, you can pause and stop individual Vuser Groups and Vusers.

The scenario ends when all the Vusers have completed their Vuser 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 the scenario.

3 Set the results directory.

4 Run and monitor the scenario.

While the scenario runs, you can pause and stop individual Vuser Groups and Vusers; monitor each Vuser in the scenario; and view error and notification messages generated by the Vusers.

Running an Entire Scenario

You can run all the Vusers 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, Astra LoadTest does not begin running Vusers until all of them have reached the *Ready* state. However, if you run individual Groups or Vusers, LoadRuner runs the Vusers as soon as they reach the *Ready* state.

The following section describes how to run an entire scenario. "Controlling Individual Vusers and Vuser Groups," on page 91, describes how to manipulate individual Vusers.

To run an entire scenario:

1 Open an existing scenario or create a new one.

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

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

During scenario execution you can manipulate individual Vusers and Vuser Groups in the scenario. This is described in "Controlling Individual Vusers and Vuser Groups," on page 91.

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

Controlling Individual Vusers and Vuser Groups

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

Initializing Vusers

Initializing a Vuser distributes the Vuser to its designated host so that the Vuser is ready to execute its script. By initializing all of the Vusers before running them, you can insure that they begin executing the scenario at the same time.

To initialize a Vuser:

1 Select the Vuser Groups or Vusers that you want to initialize.

2 Choose **Vuser** > **Initialize** or **Group** > **Initialize**. The Vuser's status changes from DOWN to PENDING to INITIALIZING to READY. If a Vuser fails to initialize, the Vuser status changes to ERROR.

Running Vusers

Running a Vuser tells the Vuser to execute its script.

To run a Vuser:

1 Select the Vuser Groups or Vusers that you want to run.

2 Choose **Vuser** > **Run** or **Group** > **Run**. The Vusers execute their scripts. If you run a Vuser in the DOWN or ERROR state, Astra LoadTest initializes and then runs the Vuser.

Stopping Vusers

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

To stop a Vuser:

1 Select the Vuser Groups or Vusers you want to stop.

2 Choose **Vuser** > **Stop**, or click the **Stop** button. The Vusers stop executing their scripts.

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 dialog box opens.

2 Select a rendezvous from the Rendezvous list.

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

11

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
- Viewing the Agent Summary

About Viewing Vusers During Execution

Astra LoadTest lets you view Vuser activity during a scenario.

On the Controller host 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 Vuser view to monitor the actions of all the Vusers in the scenario.

The Status field of each Vuser displays the current state of the Vuser. The following table describes the possible Vuser states.

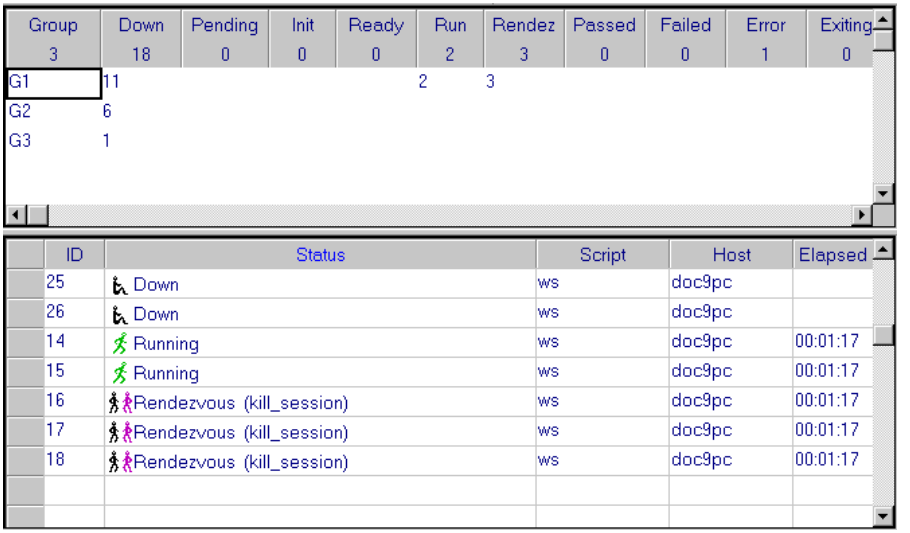
Status	Description
DOWN	The Vuser is down.
PENDING	The Vuser is ready to be initialized and is waiting for an available host, or is transferring files to the host. The Vuser will run when the conditions set in its scheduling attributes are met.
INITIALIZING	The Vuser is being initialized on the remote 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 host.
PAUSED	The Vuser has stopped running and is waiting for the signal to resume.
RENDEZVOUS	The Vuser has arrived at the rendezvous and is waiting to be released by Astra LoadTest.
EXITING	The Vuser has finished running or has been stopped, and is now exiting.
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.
STOPPED	The Vuser stopped when the Stop command was invoked.

You can display scenario statistics by expanding the Vuser view. This allows you to see how many Vusers in each Vuser Group are in the DOWN, PENDING, INIT, READY, RUN, RENDEZ, PASSED, FAILED, ERROR, EXITING, and STOPPED states.

To show the Group totals:

1 Click the **Vuser** tab to open the Vuser view. Click within the Vuser list.

➤ If Group totals are not displayed, choose **Vuser** > **Show Group Totals**, or the **Show Group Totals** button.



Group	Down	Pending	Init	Ready	Run	Rendez	Passed	Failed	Error	Exiting
3	18	0	0	0	2	3	0	0	1	0
G1	11				2	3				
G2	6									
G3	1									

ID	Status	Script	Host	Elapsed
25	Down	ws	doc9pc	
26	Down	ws	doc9pc	
14	Running	ws	doc9pc	00:01:17
15	Running	ws	doc9pc	00:01:17
16	Rendezvous (kill_session)	ws	doc9pc	00:01:17
17	Rendezvous (kill_session)	ws	doc9pc	00:01:17
18	Rendezvous (kill_session)	ws	doc9pc	00:01:17

To hide the Group totals:

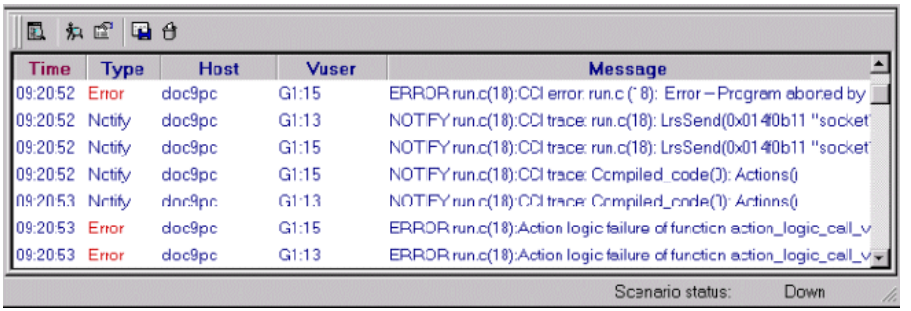
1 Click the **Vuser** tab to open the Vuser view. Click within the Vuser list.

➤ Choose **Vuser** > **Hide Group Totals**, or click the **Hide Totals** button.

Viewing the Output Window

While the scenario runs, the Vusers and hosts send error and notification messages to the Controller. You can view these messages in the Output

window. The messages in this window are cleared 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
HOST	the Vuser's host
VUSER	the ID of the Vuser that sent the message
MESSAGE	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 A, "Working in Expert Mode."

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

➤ To show (or hide) the Output window choose **View** > **Output**.

➤ To sort output messages, click the appropriate column header or click within the Output window and choose **Output** > **Sort Messages**.

➤ To filter output messages to display only certain message types, choose **Output** > **Filter Messages**.

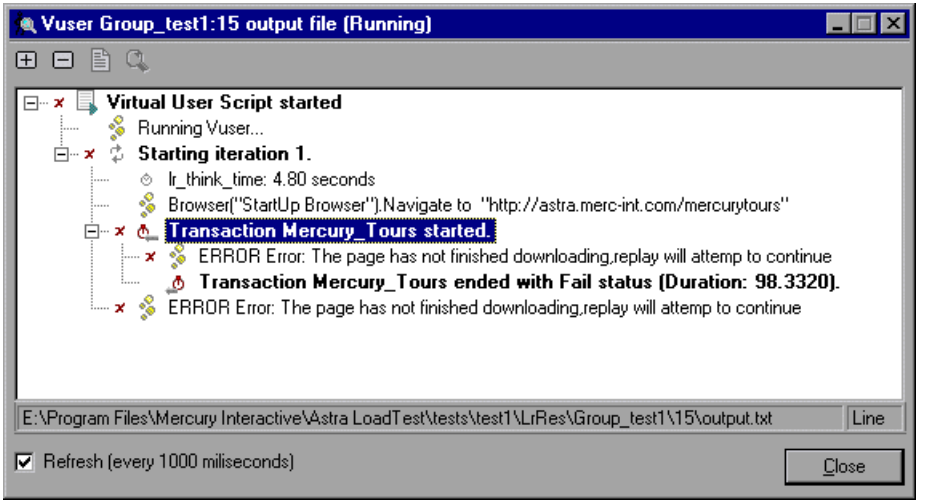
➤ To view all of a message's details, select a message and choose **Output** > **Details** or click the **Message Details** button.

➤ To save messages, choose **Output** > **Export To File**, or click the **Export** button.

➤ To clear scenario execution messages, choose **Output** > **Remove All**, or click the **Remove All** button.

➤ To view more details about a Vuser who sent a message, select a message and choose **Output** > **Vuser Details** or click the **Vuser Details** button.

➤ To view the complete Vuser log, select a message and choose **Output** > **Show Vuser Log**. This log contains runtime information about the Vuser and is refreshed every 1000 milliseconds. To disable the refreshing of this log, clear the **Refresh** check box.

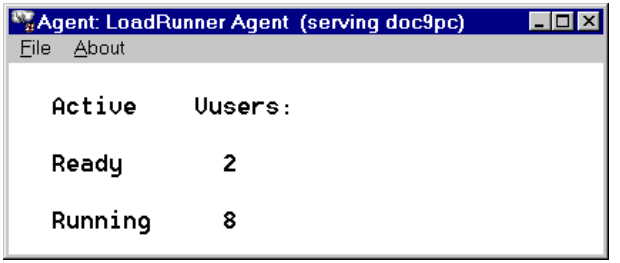


➤ To view a snapshot of the Web page where an error occurred, highlight the error in the Vuser Log and click the **Show Snapshot** button.

Viewing the Agent Summary

When you run a scenario, the machine running the Vusers invokes an agent that controls the Vuser execution on that host. During scenario execution, the agent displays a summary of the Ready and Running 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

12

Online Monitoring

You can monitor a scenario execution using Astra LoadTest's online Transaction and server monitors.

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

- Starting the Monitors
- Understanding Online Monitor Graphs
- Exporting Online Monitor Graphs
- Configuring Online Measurements
- Configuring Online Graphs
- Viewing Data Offline

About Online Monitoring

Astra LoadTest provides the following online monitors:

- Vuser Status
- Transaction
- Web

The *Vuser Status* monitor displays the current states of the Vusers participating in the scenario. (READY, RUNNING, etc.) 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 13, "Runtime, Transaction, and Web Graphs."

The *Web* monitor displays Web-related statistics. For more information, see Chapter 13, "Runtime, Transaction, and Web Graphs."

All of the monitors allow you to view a summary of the collected data at the conclusion of the scenario. Using Astra LoadTest Analysis, you can generate a graph for any of the monitors. For more information, see Chapter 14, "Understanding Astra LoadTest Analysis."

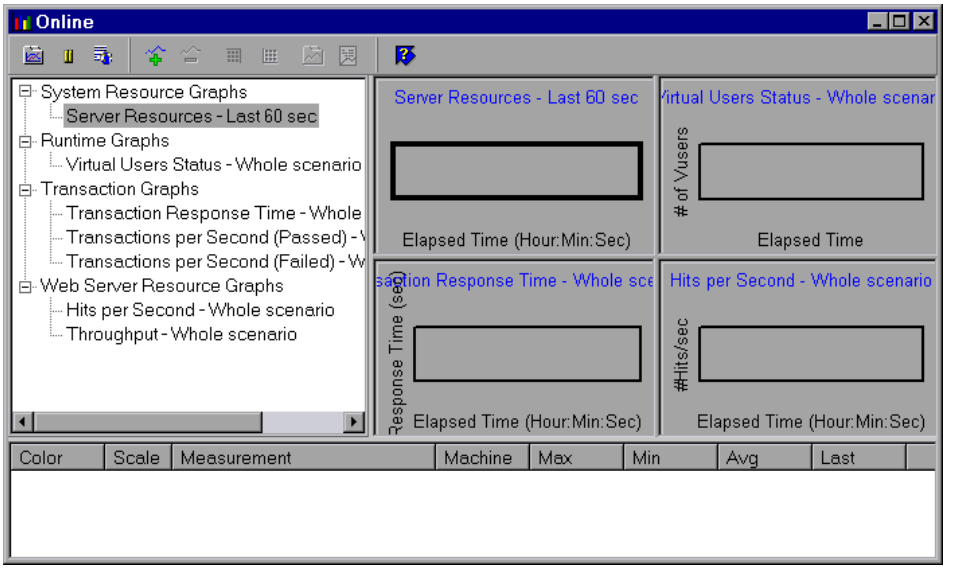
Starting the Monitors

After you configure the monitors, you can start monitoring the resources or transactions in the Online Monitor view.

To start the online monitors:

1 Start the scenario. Select the Vusers you want to run and choose **Scenario** > **Start**.

2 Click the **Online Graphs** tab to open the Online Monitor view. The default graphs are displayed.



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

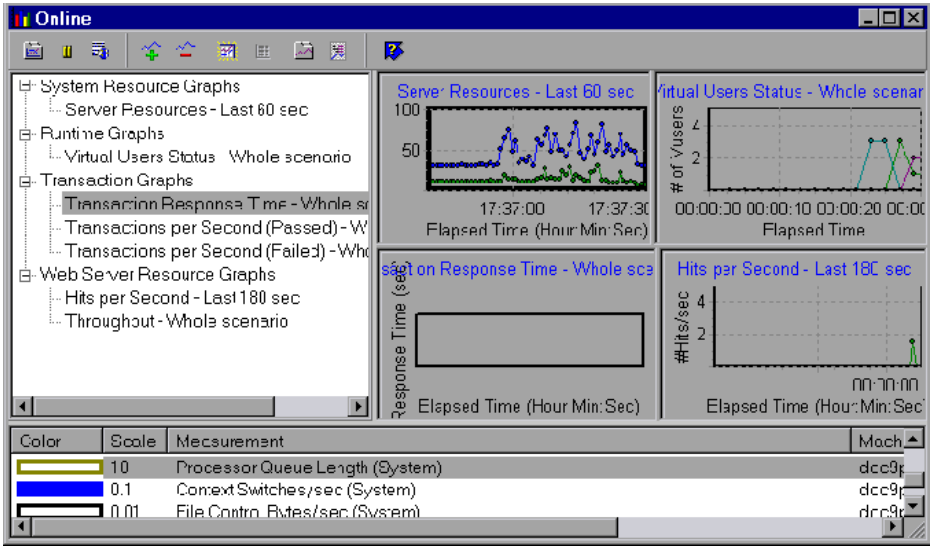
4 Click the "**+**" in the left pane to expand the graph tree.

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

Note that Transaction Monitor graphs will not contain any data unless transactions are being executed.

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 all of the transactions in a scenario and all resources in the measurement list. The legend below the graph only displays the measurements for the selected graph.

To get additional information about a measurement, select it in the legend and choose **Online** > **Measurement** > **Description** or click the **Description** button.

To focus on a particular line, you can:

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

➤ **Hide a measurement:** To hide one or more measurements, select it in the legend and choose **Online** > **Measurement** > **Hide** or click the **Hide** button.

To show a hidden measurement, select it in the legend and choose **Online** > **Measurement** > **Show** or click the **Show** button.

➤ **Pause the monitor:** To pause a monitor during scenario execution, choose **Online** > **Graph** > **Freeze > Selected** or click the **Freeze** button. To resume, repeat the above action. When you resume, the graph displays the data for the paused period.

Exporting Online Monitor Graphs

Astra LoadTest 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 an online graph to HTML:

1 In the Online Monitor view, choose **Online** > **Graph** > **Export to HTML file** or click the **Export to HTML** button.

Configuring Online Measurements

You can configure the following online measurement settings:

➤ line color

➤ scale

➤ show/hide

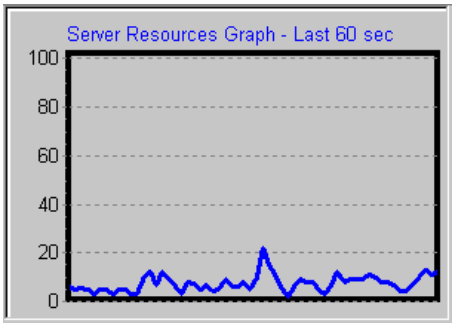
Changing Line Colors

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

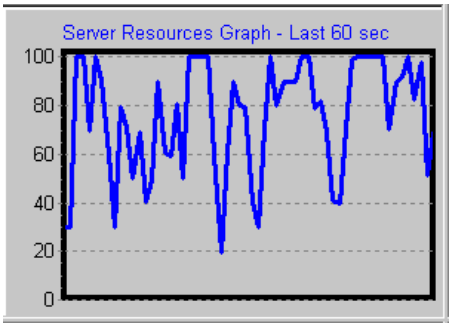
Setting the Scale of the Measurement

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.



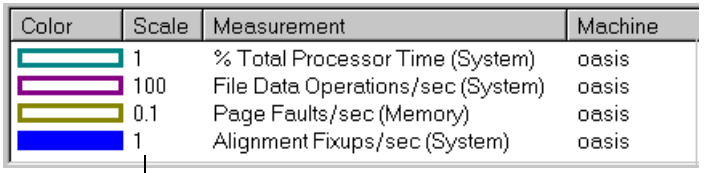
scale = 1



scale = 10

The actual graph values range from 0-20, 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 first value shown in the graph is 30. Since the scale is 10, the actual value is 3.

The legend below the graph indicates the scale factor.



Color	Scale	Measurement	Machine
	1	% Total Processor Time (System)	oasis
	100	File Data Operations/sec (System)	oasis
	0.1	Page Faults/sec (Memory)	oasis
	1	Alignment Fixups/sec (System)	oasis

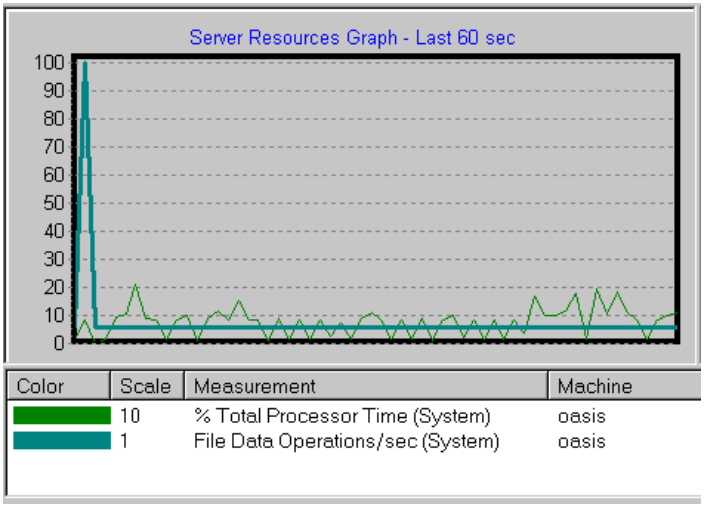
scale factor

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

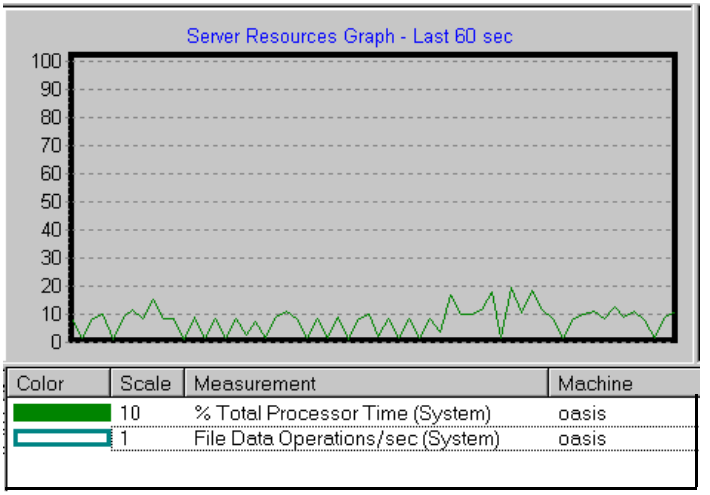
Showing and Hiding Resources

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

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

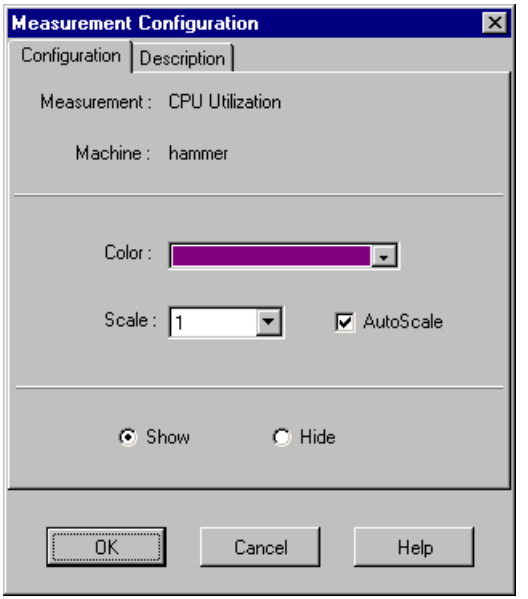


In this example, the second item in the legend is hidden.



To configure a measurement:

1 Click the **Configure** button or choose **Online** > **Measurement** > **Configure**. The Measurement Configuration dialog box opens.



2 Click inside the graph.

3 Highlight a measurement in the legend below the graph.

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

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

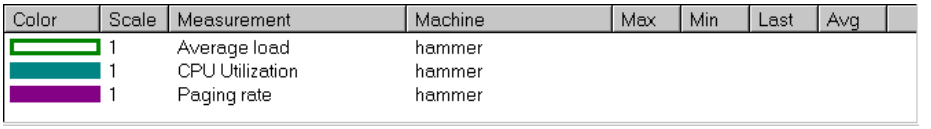
6 To hide a measurement, click **Hide**. To show a hidden resource click **Show**.

Note that you can also show and hide measurements without opening the Measurement Configuration dialog box. Click the **Hide** button or select Hide from the shortcut menu.

7 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

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



Color	Scale	Measurement	Machine	Max	Min	Last	Avg
	1	Average load	hammer				
	1	CPU Utilization	hammer				
	1	Paging rate	hammer				

Configuring Online Graphs

You can customize your graph in the following areas:

- sample rate
- graph and axis titles
- graph time
- duplicate views

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

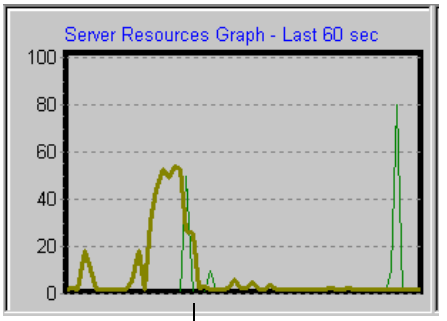
Sample Rate

The sample rate is the period of time (in seconds) between consecutive samples. By default, the online monitor samples the data at intervals of one second. If you increase the sampling rate, the data is monitored less frequently.

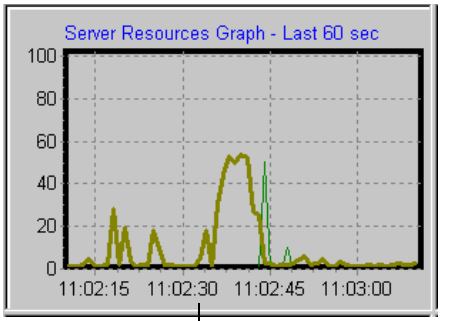
X-axis Time

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 Astra LoadTest not to display a name for the x-axis. The **Clock time** setting displays the absolute time, based on the system clock. The **Relative to Scenario Start** 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



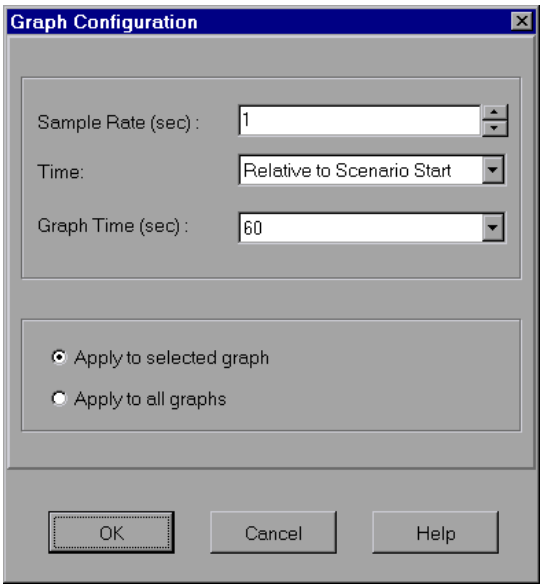
clock time

Graph Time

The Graph Time settings indicate the scale for a graph's x-axis when it time-based. By default, the x-axis shows 60 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.

To configure graph time:

1 Choose **Online > Graph** > **Configure** or click the **Configure** button. 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 sampling rate, the time in seconds between two sequential samples in the **Sample Rate** box.

4 Select a style for the x-axis from the **Time** box. The available values are: *Don't Show, Clock time,* or *Relative to Scenario Start*.

5 Select a value from the **Graph Time** box. The graph time is the time in seconds displayed by the x-axis. The available values are: *Whole scenario, 60, 180, 600,* and *3600* seconds.

6 Click **OK**.

The online monitor displays the graph with the specified settings. To customize the scale of the y-axis for each individual graph, see "Configuring Online Measurements" on page 105.

Duplicate Views

You can monitor information from more than one server by creating duplicate views.

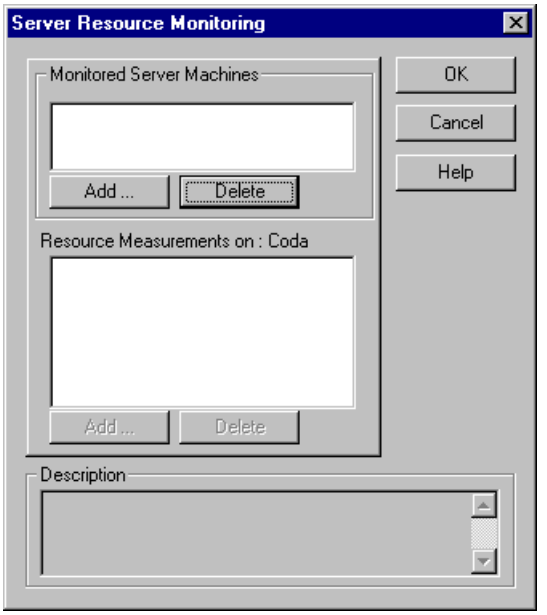
To create a duplicate graph:

1 Right click the graph whose duplicate you want to create, in the left pane of the Controller window (Online Graphs view).

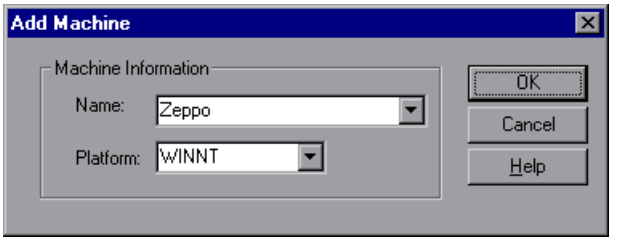
2 Choose **Duplicate** from the displayed menu list. The duplicate graph is added to the list of Online graphs.

You can rename the graph by right clicking the duplicate graph and selecting **Rename** from the menu list.

3 Choose **Monitors** > **Add Online Measurement**.



4 Click **Add...** in the **Monitored Server Machines** section to add the host you want to monitor. The Add Machine dialog box opens.



5 Type a machine name and select a platform. Click **OK** to accept the settings and close the dialog box.

6 To delete a duplicate graph, right clich it and select **Delete** from the displayed menu list.

Viewing Data Offline

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

To view a graph, choose **Graphs** in the Analysis window and select the appropriate graph.

For more information about working with the Astra LoadTest Analysis at the conclusion of the scenario, see Chapter 14, “Understanding Astra LoadTest Analysis.”

13

Runtime, Transaction, and Web Graphs

While running a scenario, you can use Astra LoadTest's online monitors to view graphs of runtime status, transaction performance, and Web statistics.

This chapter describes:

- Runtime Graphs
- Enabling the Transaction Monitor
- Adding Transactions to a Script
- Transaction Monitor Graphs
- Web Monitor Graphs

About Online Graphs

Astra LoadTest's online monitor lets you view Vuser performance during scenario execution. You can view the current Vuser status, transaction rate and response time, and Web statistical information.

The *Runtime* monitor provides information about the current state of the Vusers participating in a scenario. It also provides the Data Point graph that displays the real time values for user- defined points.

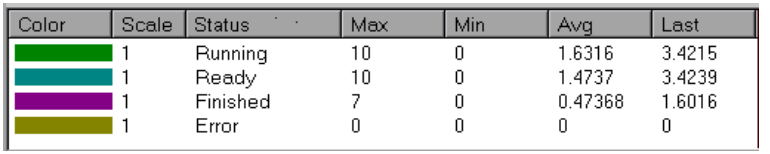
The *Transaction* monitor displays the transaction rate and response time during scenario execution. For more information about transactions, see "Adding Transactions to a Script" on page 118.

The *Web* monitor provides information about the number of web connections and throughput volume during the scenario.

Once the scenario finishes execution, you can use the Astra LoadTest Analysis to view summary graphs of all the data collected during execution. For more information, see Chapter 14, "Understanding Astra LoadTest Analysis."

Runtime Graphs

The monitor's *Runtime* graph provides information about the status of the Vusers running in the current scenario on all host 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	Last
	1	Running	10	0	1.6316	3.4215
	1	Ready	10	0	1.4737	3.4239
	1	Finished	7	0	0.47368	1.6016
	1	Error	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 hosts.
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.

User-Defined Data Point Graph

The User-Defined Data Point graph displays the real-time values of user-defined data points. User data points are represented by **lr_user_data_point**

functions in the Vuser script. For further information about this function, see the *Astra LoadTest Online Function Reference*.

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

By default, Astra LoadTest 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 Chapter 17, “Analyzing Scenario Performance.”

Transaction Monitor Graphs

The *Transaction* monitor provides the following graphs:

- Transaction Response Time
- Transaction per Second (Passed)
- Transaction per Second (Failed)

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

The **Transaction 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).

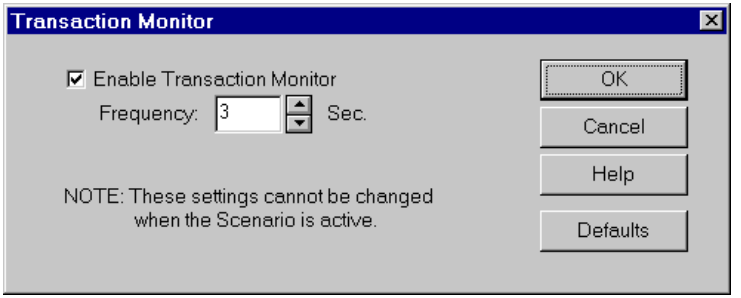
The **Transaction per Second (Failed)** graph shows the number of failed transactions per second (y-axis) as a function of the elapsed time in the scenario (x-axis).

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. You can also set the frequency in which the monitor plots transaction performance. The default value is 3 seconds. For quicker transactions, reduce this frequency.

To configure the Transaction monitor:

1 Choose **Monitor** > **Vuser Transactions**. The Transaction Monitor dialog box opens.



2 To use transaction monitoring, ensure that the **Enable Transaction Monitor** box is checked.

3 Specify a frequency for the sampling of the transaction response times.

4 To disable transaction monitoring, clear the **Enable Transaction Monitor** check box.

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

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 Virtual User Recorder.

To add a transaction to a script:

1 Click the **Scripts** tab to view the script list.

2 Select the script you want to edit.

3 Choose **Script** > **Edit script** to open the script generation tool.

4 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.

Web Monitor Graphs

The *Web* monitor provides the following graphs:

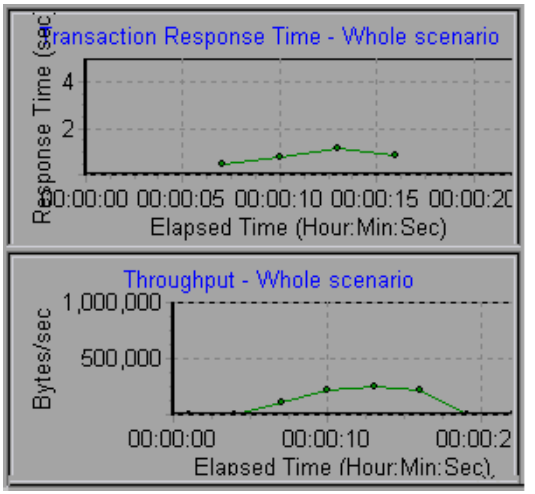
- Hits Per Second
- Throughput

The **Hits Per Second** graph shows the number of hits on 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 server (y-axis) during each second of the scenario run (x-axis). Throughput is measured in kilobytes 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 12 seconds into the scenario. The highest response time also occurred at this time.



Part V

Analyzing Test Results

14

Understanding Astra LoadTest Analysis

After running a scenario, you can use Astra LoadTest's graphs and reports to analyze the performance of your client/server system.

This chapter describes:

- Starting the Analysis
- Collating Execution Results
- Viewing a Part of the Scenario
- Working with Analysis Graphs
- Working with Analysis Reports
- Spreadsheet and Raw Data Views

About Astra LoadTest Analysis

During scenario execution, Vusers generate result data as they perform their transactions. To monitor the scenario performance during test execution, use the Online analysis (See Chapter 13, "Runtime, Transaction, and Web Graphs.")

After running a scenario, you can view a summary of the results to determine how your system performed. You can view the results in several ways:

- The **Vuser output file**, *output.txt*, contains a full trace of the scenario run for each Vuser. This file is located in the scenario results directory. (When you run a Vuser script in standalone mode, this file is placed in the Vuser script directory.) For more information on Vuser output files, refer to the *Virtual User Recorder* guide.

- The **Controller Output window** displays information about the scenario run. If your scenario run fails, look for debug information in this window. For more information, see Chapter 10, "Running a Scenario."
- The **Analysis graphs and reports** provide information about transactions and Vusers that can be used to determine system performance. You can study individual graphs or you can compare graphs from several scenarios using Cross-Scenario analysis.
- The **Spreadsheet** and **Raw Data** views display the actual data used to generate the graph in a spreadsheet format.
- The **Export data** utility enables you to export scenario results to text or database files for use with external applications such as spreadsheet programs.

This chapter discusses the graphs, reports, and the data exporting utility.

Starting the Analysis

The Analysis is the Astra LoadTest component that processes results gathered during a scenario execution and generates graphs and reports. You can open the Analysis through the Astra LoadTest program group or directly from the Controller.

When you open the Analysis as an independent application from the program group, you select a Astra LoadTest Results (*.lrr*) file. The Analysis processes the data and automatically displays the Percentile graph. This graph indicates the percentage of transactions completed with a given response time.

To open the Analysis directly from the Controller, select **Results** > **Analyze Results**. This option is only available after running a scenario. It automatically takes the latest result file from the current scenario, and opens it. You can also instruct Astra LoadTest to automatically open the Analysis after it completes scenario execution by selecting Auto Load Analysis. To enable automatic analysis, select **Results** > **Auto Load Analysis**.

Collating Execution Results

When you run a scenario, by default all Vuser information is stored locally on each Vuser host. After scenario execution the results are automatically *collated* or consolidated—results from all of the hosts are transferred to the results directory. 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, Astra LoadTest Analysis will automatically collate the results before generating the analysis data. For more information about collating results, see Chapter 10, "Running a Scenario."

Analysis Graphs and Reports

Astra LoadTest's graphs and reports are divided into the following categories:

- Activity
- Performance
- Web

Activity graphs and reports provide information about the number of Vusers and the number of transactions executed during the scenario run. For more information, see Chapter 16, "Analyzing Scenario Activity."

Performance graphs and reports analyze Vuser performance and transaction times. For more information, see Chapter 17, "Analyzing Scenario Performance."

Web graphs provide information about the throughput and hits of Web Vusers. For more information see Chapter 19, "Web Vuser Graphs."

Astra LoadTest cross-scenario analysis lets you compare the graphs from several scenarios. For example, this feature can be useful for benchmarking hardware to determine how a superior system will improve performance. For more information, see Chapter 18, "Cross-Scenario Analysis."

The following tables shows the available graphs and reports. Several graphs allow you to display their raw data (see See "Spreadsheet and Raw Data Views" on page 137.).

Activity Graphs and Reports

Name	Graph	Report	Raw Data	Cross Scenario	Description
Running Virtual Users	✓			✓	The number of Running Vusers at any moment
Rendezvous	✓	✓		✓	Rendezvous data: The number of arrived members, etc.
Transactions per Second (Pass/Fail)	✓		✓	✓	The number of transactions executed per second
Scenario Execution		✓		✓	Detailed scenario information for each Vuser

Performance Graphs and Reports

Name	Graph	Report	Raw Data	Cross Scenario	Description
Data Point	✓	✓	✓	✓	The performance of an external utility
Percentile	✓			✓	The percentage of transactions performed within a given time range
Performance Under Load	✓	✓		✓	The transaction response time according to the number of running Vusers
Transaction Performance	✓	✓*	✓	✓	The transaction response time as a function of the elapsed scenario time
Transaction Performance Summary	✓	✓		✓	The minimum, maximum, and average transaction time
Transaction Performance Summary by Vuser	✓	✓		✓	The minimum, maximum, and average transaction times for each Vuser
Transaction Distribution	✓		✓	✓	The distribution of time for transaction executions

* The report is titled Detailed Transactions.

Web Graphs

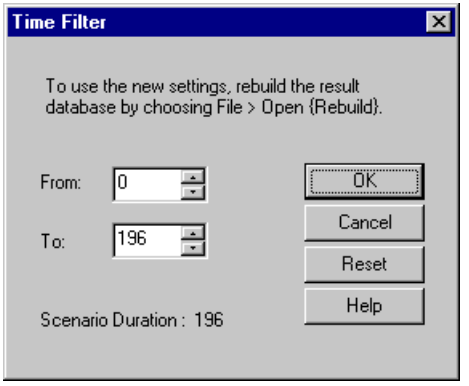
Name	Graph	Report	Raw Data	Cross Scenario	Description
Hits per second	✓		✓	3	The number of web server hits per second
Throughout	✓		✓	3	The number of kilobytes transferred per second

Viewing a Part of the Scenario

You can instruct Astra LoadTest to calculate graphs and reports based on a part of the scenario. For example, if you have a scenario with a duration of 400 seconds, but you are only interested in the period beginning at 60 and ending with 360 seconds, you specify that interval in the Time Filter dialog box. After setting the time filter, you must rebuild the result database in order for the changes to take effect. All subsequently generated graphs and reports are calculated using data only from the specified interval.

To filter the scenario time:

1 Choose **Options** > **Time Filter** in the Analysis window. The Time Filter dialog box opens.



2 Enter the desired time interval.

3 Click **OK** to accept the settings.

4 Choose **File > Open** (Rebuild).

Working with Analysis Graphs

You generate graphs in the Analysis window by selecting the desired graphs from the Graph menu. For information about the available graphs, see Chapter 16, "Analyzing Scenario Activity" and Chapter 17, "Analyzing Scenario Performance."

Astra LoadTest also provides tools which enable you to modify the way that graphs are displayed. For each graph you can:

- filter specific transactions and Vusers
- enlarge a section of the graph
- change the interval of the *x-axis*

Viewing Specific Transactions and Vusers

When you generate a graph, the graph initially includes all the transactions in the scenario. You can filter the graph to display one or more specific transactions. For example, if three transactions were monitored during a scenario, you can generate a graph that displays only one of the transactions.

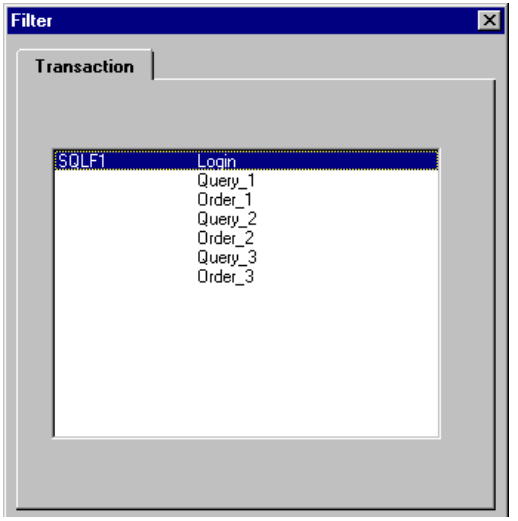
By default, the Transaction Performance and Transaction per Second graphs display all the transactions, while the Transaction Performance Summary and Transaction Distribution graphs display only the first transaction.

To select transactions and Vusers to display:

1 In the **Analysis** window, choose **File** > **Open** to open the desired results file (if it is not already open). The Percentile graph is displayed.

2 Select a graph from the **Graphs** menu. The graph is displayed. You can display multiple copies of the same graph.

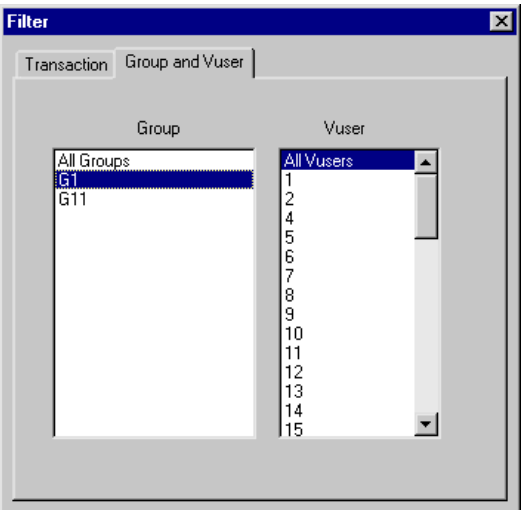
3 Select **View** > **Filter**, or click the Graphic Filters button.



4 Select the transactions that you want to include in the graph. Use the CTRL key to select multiple transactions.

For graphs that show individual Vuser performance, such as Transaction Performance Summary by Vuser graph, you can filter the graphs to display

information for a single Vuser. Click the Group and Vuser tab and select the Vuser that you want to display.



5 Click **OK**. The Filter dialog box closes. The graph appears and displays the selected transactions or Vusers.

Using Location Lines

To determine the precise value of a point in a graph, you can use *location lines*. When location lines are enabled, a vertical and horizontal line (parallel to each of the axes) are displayed in the graph.

To enable location lines, choose **View > Location lines** in the Analysis window. Click on the point within your graph whose coordinates you want to determine. Note that even when you disable location lines, the coordinates of the cursor are displayed on each axis.

Enlarging a Section of a Graph

Graphs initially display data representing the entire duration of the scenario. You can enlarge any section of a graph to focus on a specific period in the scenario run. For example, if a scenario ran for ten minutes, you can enlarge and focus on the scenario events that occurred between the second and the fifth minutes. You can enlarge graphs using the mouse, or the Zoom In command.

To enlarge a section of a graph using the mouse:

1 Select a graph from the Graphs menu. The graph is displayed.

2 Move the mouse pointer to the beginning of the section that you want to enlarge.

3 Hold down the left mouse button. A vertical bar appears on the *x-axis*. This marks the first boundary of the section to enlarge.

4 While holding down the left mouse button, move the mouse to the left or right of the vertical bar. A second vertical bar appears. This marks the second boundary of the section to enlarge.

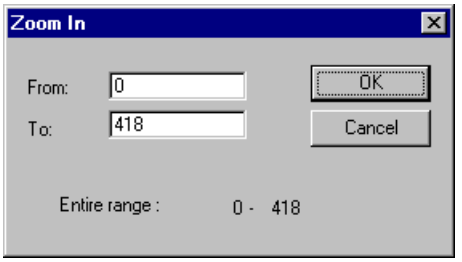
5 Release the left mouse button. The section is enlarged.

6 Click the right mouse button to return the graph to its original size.

To enlarge a section of a graph using the Zoom In command:

1 Select a graph from the Graphs menu.

2 Select **View** > **Zoom In**, or click the **Zoom** button. The Zoom In dialog box opens.



3 In the **From** box, enter the point that marks the left boundary of the section to enlarge. Note that this point must be on the *x-axis*.

4 In the **To** box, enter the point that marks the right boundary of the section to enlarge.

5 Click **OK**. The section is enlarged.

6 Select **View** > **Zoom Out** to return the graph to its previous size.

7 Select **View** > **Full View** to return the graph to its original size.

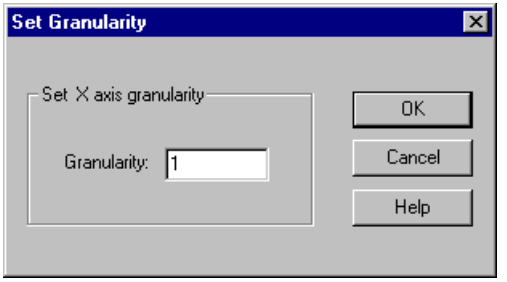
Changing the Granularity of the X-axis for a Graph

You can make Astra LoadTest graphs easier to read and analyze by changing the granularity (scale) of the *x-axis*.

To change the granularity of the x-axis for a graph:

1 Select a graph from the **Graphs** menu.

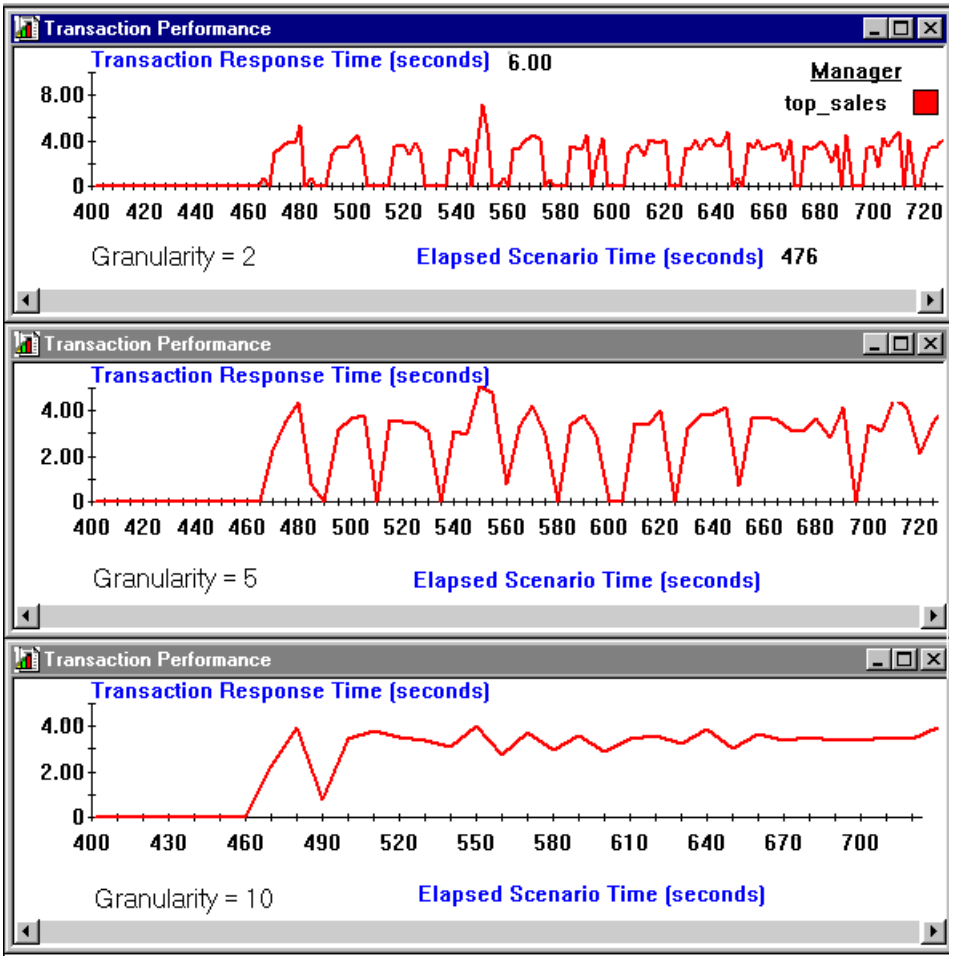
2 Select **View** > **Granularity**, or click the **Granularity** button. The Set Granularity dialog box opens.



3 Enter a new scale in the Granularity box.

4 Click **OK**. Astra LoadTest regenerates the graph.

In the following example, the Transaction Performance graph is displayed using different granularities. The y-axis represents the average transaction response time within the granularity interval. For a granularity of 1, the *y-axis* shows the average response time for each one second period of the scenario. For a granularity of 2, the *y-axis* shows the average response time for every two second period of the scenario.



In the above graphs, the same scenario results are displayed in a granularity of 2, 5, and 10 (from top to bottom). The lower the granularity, the more detailed the results. It is useful to use a higher granularity to study the overall Vuser behavior throughout the scenario. For example, using a low granularity as in the upper graph, you see the intervals when no transactions were performed. By viewing the same graph with a higher granularity, you can easily see that the overall transaction response time averaged 3.5 seconds.

Working with Analysis Reports

In order to view a report, you must generate the report from the Astra LoadTest Analysis window. Astra LoadTest reports are displayed in a Report Viewer. You can print, save, or export the data using the viewer.

Selecting and Displaying Reports

Astra LoadTest provides several built-in reports which contain detailed summaries about the scenario, the transactions and Vusers. For a list of the available reports, see "Analysis Graphs and Reports" on page 125.

To display a report:

1 In the **Analysis** window, choose **File** > **Open** to open the desired results file (if it is not already open).

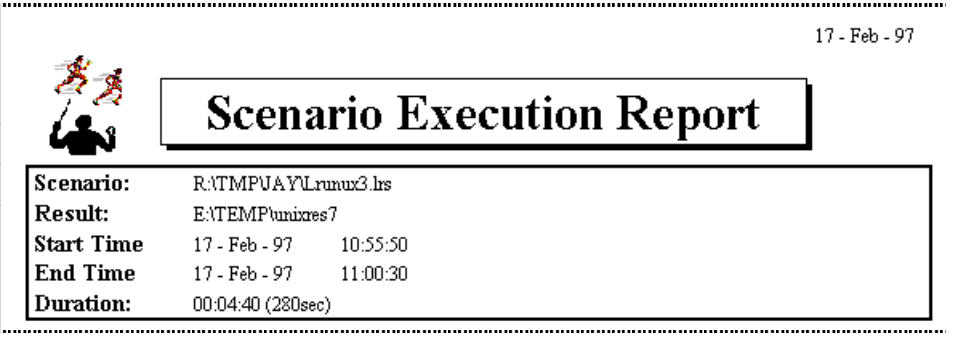
2 From the **Reports** menu choose a report. The report is generated and displayed. You can display multiple copies of the same report.

The Report Viewer

Each report is displayed in its own report viewer. Each viewer contains a header and a toolbar.

Report Header

The header displays general scenario information.



The report header contains the following information:

Title: The name of the report.

Scenario: The name of the scenario described in the report.

Result: The pathname of the scenario results directory.

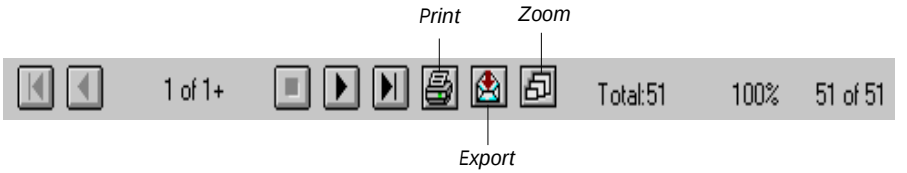
Start time: The time at which the Run Scenario command was executed.

End time: The time at which the scenario script was terminated.

Duration: The total run time of the scenario.

Report Viewer Toolbar

Each report viewer has a toolbar that lets you perform operations on displayed reports.



The report viewer toolbar contains the following buttons:

Zoom: Toggles between an actual size, full page, and magnified views of the report.

Print: Prints the displayed report.

Export to file: Exports the displayed information to a disk file.

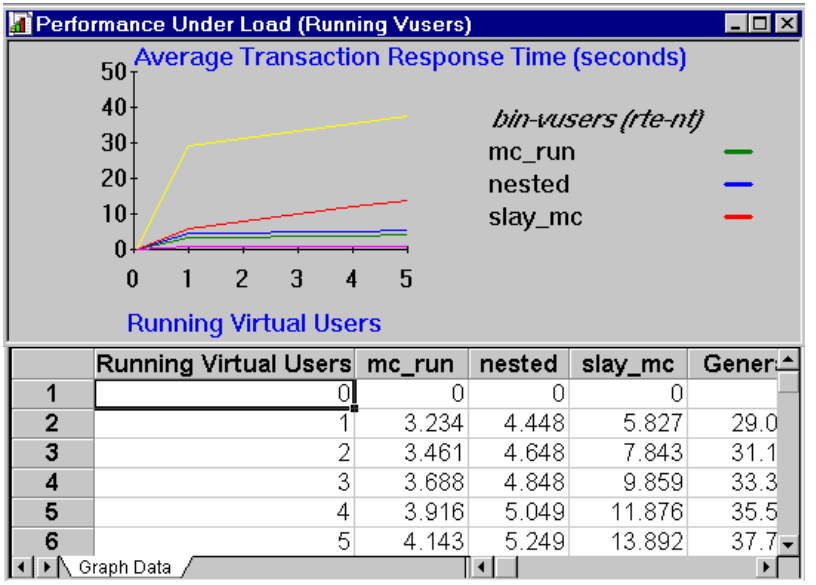
Spreadsheet and Raw Data Views

Astra LoadTest allows you to view the data in a grid form in two ways:

- **Spreadsheet**: The graph values displayed in a grid.
- **Raw Data**: The actual raw data collected during the scenario.

Spreadsheet View

You can view the graph displayed by Astra LoadTest Analysis in spreadsheet format. The Spreadsheet view displays the data in a grid directly below the active graph.



	Running Virtual Users	mc_run	nested	slay_mc	Gener
1	0	0	0	0	
2	1	3.234	4.448	5.827	29.0
3	2	3.461	4.648	7.843	31.1
4	3	3.688	4.848	9.859	33.3
5	4	3.916	5.049	11.876	35.5
6	5	4.143	5.249	13.892	37.7

Graph Data

The first column displays the values of the x-axis. The following columns show the y-axis values for each transaction. For time based graphs, the absolute time is displayed in the second column.

To display a graph's Spreadsheet view, select **Spreadsheet** > **View Spreadsheet**. To close the view, repeat the above step.

If there are multiple values for the y-axis, as in the Transaction Performance Summary graph (min, average, and max), all of the plotted values are displayed. If you filter out a transaction, it will not appear in the view.

To locate the spreadsheet data for a specific point on the graph, double-click that point on the graph—the spreadsheet displays the selected point.

The Spreadsheet shortcut menu (right-click) has the following additional features:

- **Export:** You can export the spreadsheet data to an Excel file. Once you have the data in Excel, you can generate your own customized graphs.
- **Copy to clipboard:** You can copy one or more columns to the clipboard in order to paste it into an external spreadsheet program.
- **Print:** You can print the data in spreadsheet format. A Print Preview option is also available from the print dialog box.
- **Sort:** You can sort the data according to any of the columns.

In the following example, the data for the Transaction per Second graph shows the Elapsed Scenario Time, Absolute Time, and number of transactions per seconds for each transaction.

	Scenario Time	Absolute Time	login	graph2	sort	graphik	clear_and_end
341	1020	11:30:49 AM	0	0	0	0	0
342	1023	11:30:52 AM	0	0.333	0	0	0.667
343	1026	11:30:55 AM	0	0	0	0	0.667
344	1029	11:30:58 AM	0	0	0.33	0	0
345	1032	11:31:01 AM	0	0	0	0.333	0.333
346	1035	11:31:04 AM	0	0	0	0	0

All Astra LoadTest graphs can be viewed in spreadsheet format.

Raw Data Format

You can view the actual raw data collected by Astra LoadTest for the active graph. The Raw Data view is not available for all graphs. Refer to the chart on page 126 for a list of the graphs with the Raw Data option. Note that in the raw data, the elapsed time is split into seconds and milliseconds.

Viewing the raw data can be especially useful for:

➤ determining specific details about a peak—for example which Vuser was running the transaction that caused the peak value(s).

➤ performing a complete export of unprocessed data for your own spreadsheet application. This capability complements the **File** menu's **Export** feature.

To display a graph's Raw Data view:

1 Open the Spreadsheet view. (if it is not already open) Select **Spreadsheet** > **View Spreadsheet**.

2 Select **Spreadsheet** > **View Raw Data**. The **Show Raw Data** dialog box opens.

3 Specify a time range—the entire graph or a specific range of time— and click OK. Astra LoadTest displays the data in a grid directly below the active graph.

	psed Time (s	ed Time	Scenario(Result)	nsaction N	Group	Vuser	sponse Tir
1	72	701	bin-vusers (rte-nt	mc_run	zorb	5	4327
2	72	764	bin-vusers (rte-nt	mc_run	zorb	1	4453
3	72	764	bin-vusers (rte-nt	mc_run	zorb	2	4437
4	72	764	bin-vusers (rte-nt	mc_run	zorb	3	4421

Graph Data | Raw Data

To show a different range, repeat the above procedure. When the Raw Data view is active, switch between the views by clicking on the appropriate tab.

15

Exporting Analysis Data

You can export the data from Astra LoadTest's Analysis graphs or reports into your own spreadsheet program.

This chapter describes:

- Exporting Graph Data
- Exporting from the Analysis Window
- Exporting Data from the Report Viewer

About Exporting Data from the Analysis

You can export performance data to a spreadsheet application or to a statistical analysis application for further processing. You can export from:

- **Graphs** - Use the Spreadsheet or Raw Data view to copy and paste data directly into a spreadsheet program or save graph data as an Excel file.
- **Analysis Window** - Generate raw data text files with performance information.
- **Reports** - Save report data in a format compatible with a spreadsheet application or Microsoft Mail.

Exporting Graph Data

You can export graph data directly through the clipboard using the Spreadsheet or Raw Data views. You select the columns you want to copy and copy them to the clipboard. You then paste the data directly into your spreadsheet program.

Alternatively, you can save the spreadsheet as an Excel file. To save the file, click anywhere in the spreadsheet grid and choose **Save As** from the shortcut menu. For more information, see Chapter 14, "Understanding Astra LoadTest Analysis."

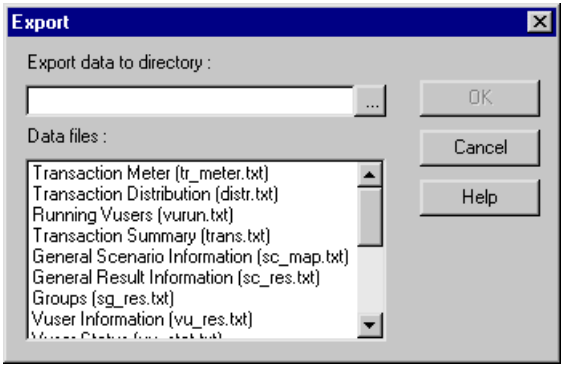
Exporting from the Analysis Window

To generate simple text files with performance information, export the data from the Analysis window. The data files are exported to the specified directory in text format with.*txt* extensions. Each file begins with an explanation of the report and its fields.

To export performance data from the Analysis window:

1 In the **Analysis** window, choose **File** > **Open** to open the desired results file (if it is not already open).

2 Click the **Export** button or choose **File** > **Export**. The Export dialog box opens.



3 In the **Export Data to Directory** box, type the path of the directory to which you want to export the information.

Alternatively, click the browse button next to the box and select a directory with the Export Path dialog box.

4 From the **Data files** list, select the information you want to export. Use the CTRL key to select multiple files. For a list of the exportable data files, see the list below.

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

You can export the following types of performance data:

Transaction Meter contains information on all the transactions in the scenario, such as the ID of the Vusers performing the transaction and transaction performance times. The exported file name is *tr_meter.txt*.

Transaction Distribution contains the distribution of the time taken to perform a transaction. The exported file name is *distr.txt*.

Running Vusers contains the total number of active Vusers during scenario execution. The exported file name is *vurun.txt*.

Transaction Summary contains general information about all transactions in the scenario, such as the minimum, maximum, and average performance times for all transactions in the scenario. The exported file name is *trans.txt*.

General Scenario Information contains the path of the scenario file and the total number of Vusers in the scenario. The exported file name is *sc_map.txt*.

General Result Information contains the path of the results file and the start and end times for the scenario. The exported file name is *sc_res.txt*.

Groups contains the name and ID of each Vuser Group in the scenario. The exported file name is *sg_res.txt*.

Vuser Information contains all the Vusers that were in the RUNNING state during scenario execution. The exported file name is *vu_res.txt*.

Vuser Status Contains the status of each Vuser at the end of the scenario (PASS, FAIL, ABORT, ERROR). The exported file name is *vu_stat.txt*.

Transactions per Second contains the number of completed transactions during each second of the scenario. The exported file name is *tps.txt*.

Performance by Vuser contains transaction performance information for each Vuser in the scenario. The exported file name is *vu_trans.txt*.

Performance Under Load contains the performance time for transactions according to the total number of active Vusers during scenario execution. The exported file name is *perf_run.txt*.

Data Point contains performance information about user defined points known as data points. The exported file name is *dpoint.txt*.

Data Point Map contains a list of the data points measured throughout the scenario. The exported file name is *dp_map.txt*.

Rendezvous contains information about the rendezvous such as the number of members arrived, the arrival and release times, and the reason for the release. The exported file name is *rendz.txt*.

Rendezvous Map contains a list of the rendezvous points in the scenario. The exported file name is *rendzmap.txt*.

Percentile contains the percentage of transactions performed within a given time range. The exported file name is *percent.txt*.

Exporting Data from the Report Viewer

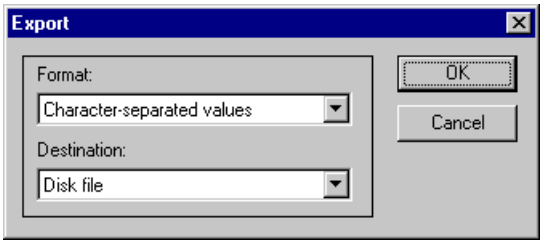
From the Report Viewer, you can export analysis data to a file, an Exchange folder, or a Lotus Notes database for further processing. You can also export Astra LoadTest reports through Microsoft Mail. For a list of the available formats, click the arrow to the right of the Destination box to open the drop-down list box. The formats that are available for export depend on the type of destination—only the appropriate formats are enabled.

To export report data to a file:

1 In the **Analysis** window, choose **File** > **Open** to open the desired results file (if it is not already open).

2 Open the **Reports** menu and choose a report to generate. Wait for the report to open.

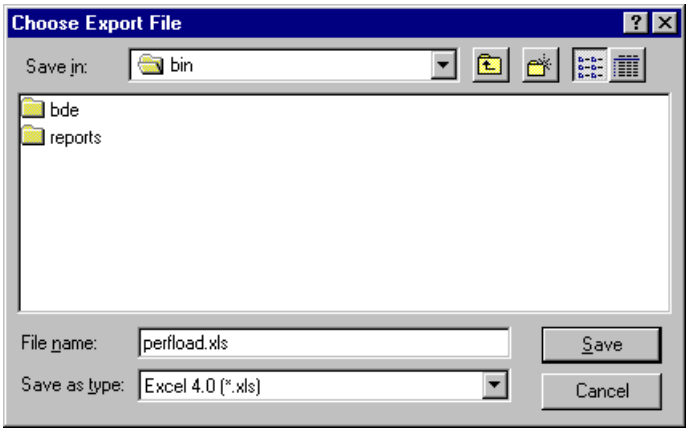
3 Click the **Export** button from the report viewer toolbar. The Export dialog box opens.



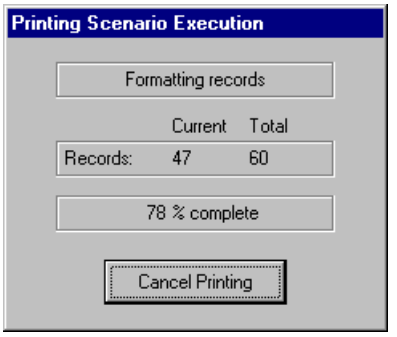
4 Click the arrow to the right of the **Format** box and select a file format from the list.

5 Click the arrow to the right of the **Destination** box and select a destination from the list.

6 Click **OK**. The Choose Export File dialog box opens.



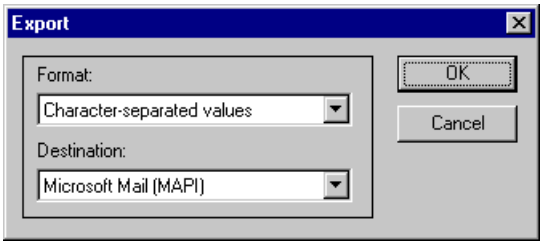
7 Specify a name for the export file and click **OK.** Astra LoadTest displays the report title in the window banner and exports the information. A dialog box displays the progress of the export. For example, if you are exporting the Detailed Vuser report, the dialog box title will read Printing Detailed Vuser.



To export report information using Microsoft Mail:

1 In the Analysis window, open the **Reports** menu and choose a report to generate. Wait for the report to open.

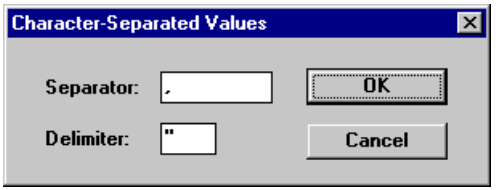
2 Click the **Export** button from the report viewer toolbar. The Export dialog box opens.



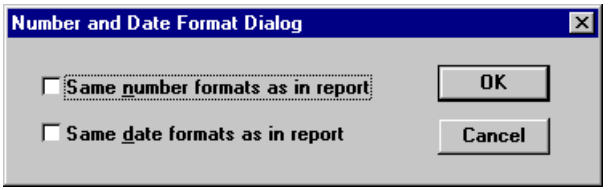
3 Select **Character separated** values from the Format box.

4 Select **Microsoft Mail** from the **Destination** box.

5 Click **OK** to close the dialog box. The Character-Separated Values dialog box opens.



6 Specify a character to separate between values in the data file to which you are exporting. The default is a comma. Click **OK** to close the dialog box. The Number and Date Format dialog box opens.



7 Check the appropriate boxes and then click **OK**. Astra LoadTest begins to export the information.

16

Analyzing Scenario Activity

After running a scenario, you can analyze the Vuser activity that occurred during the scenario to check that the scenario ran as planned. You can use the following graphs or reports to analyze Vuser activity:

- Running Virtual Users Graph
- Rendezvous Graph and Report
- Transactions per Second Graph (Passed)
- Transactions per Second (Failed) Graph
- Failed Transactions Report
- Failed Vuser Report
- Scenario Execution Report

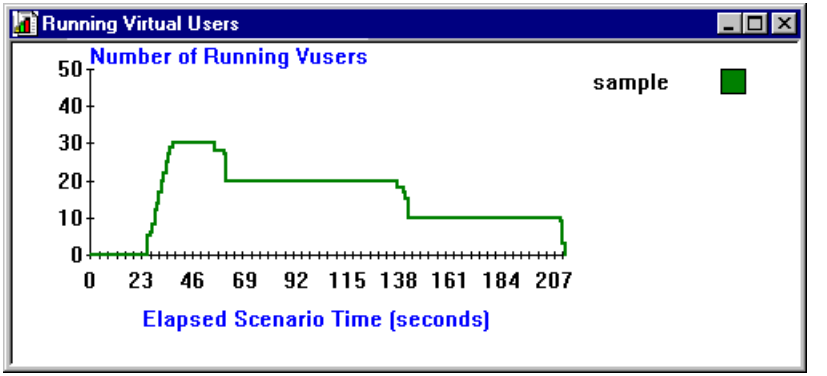
About Scenario Activity

During scenario execution, Vusers generate data as they perform transactions. Astra LoadTest enables you to generate graphs that show scenario activity or overall scenario performance. Scenario activity graphs and reports let you track Vuser behavior throughout the scenario. The graphs map the transactions, rendezvous, and active status of the Vusers, according to the elapsed scenario time.

When working with graphs, you can focus on specific segments of the scenario using the zoom tools. In addition, you can change the granularity of the graph. For more information about working with graphs, see Chapter 14, “Understanding Astra LoadTest Analysis.”

Running Virtual Users Graph

The Running Virtual User graph displays the number of Vusers that executed Vuser scripts during each second of a scenario run. Only Vusers in the RUNNING and RENDEZ states are included in the graph. Vusers in the LOADING, READY and PAUSE states are not displayed.



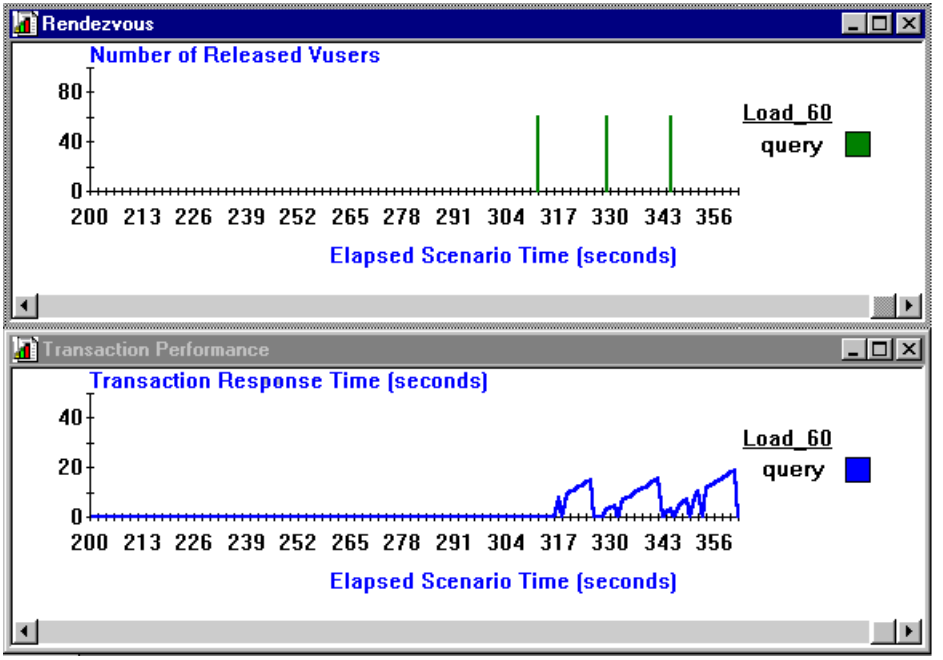
This graph is useful for determining the Vuser load on your server at any given moment. The *x-axis* represents the elapsed time (in seconds) since the start of the scenario run. The *y-axis* represents the number of running Vusers in the scenario.

For example, the above graph indicates that there was a maximum load of thirty Vusers. Until the 37th second of the scenario run, Vusers were gradually loading. Thereafter, the number of running Vusers decreased to twenty, and then to ten.

Rendezvous Graph and Report

The Rendezvous graph indicates when Vusers were released from rendezvous points, and how many Vusers were released at each point.

This graph helps you understand transaction performance times. If you compare the Rendezvous graph to the Transaction Performance graph, you can see how the load peak created by a rendezvous influences transaction times.



On the Rendezvous graph, the *x-axis* indicates the number of seconds that elapsed since the start of the scenario run. The *y-axis* indicates the number of Vusers that were released from the rendezvous. If you set a rendezvous for 60 Vusers, and the graph indicates that only 20 were released, you can see that the rendezvous ended when the timeout expired because all of the Vusers did not arrive.

In the above Rendezvous graph, the first *query* rendezvous takes place during the 312th second of the scenario, and 60 Vusers were released. By

comparing these results to the Transaction Performance graph, you see that the peak time for the query transaction occurred under the load of sixty Vusers, with a response time of 20 seconds.

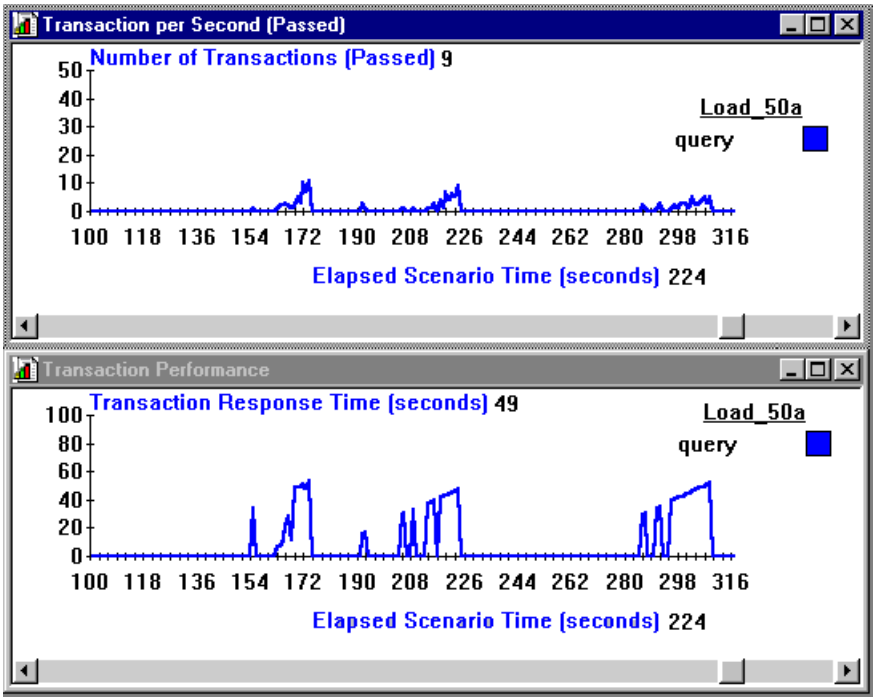
The Rendezvous report provides additional information about the rendezvous.

Rendezvous	*First arrival time*	*Release time*	*Members*	*Released*	*Reason*
query_rnd	11:54:11	11:55:06	50	40	Time out
query_rnd	11:55:11	11:55:42	50	10	by vuser
query_rnd	11:55:43	11:56:01	50	50	OK
query_rnd	11:56:19	11:57:21	50	48	Time out

In the above report, the rendezvous policy was set to All Arrived, requiring all 50 Vusers to arrive at the rendezvous point. In the first rendezvous, 40 Vusers were released after the timeout period while 10 Vusers were manually released by the operator. In the next rendezvous, all 50 Vusers arrived. In the last rendezvous, two Vusers never arrived, causing the others to reach the timeout.

Transactions per Second Graph (Passed)

The Transactions per Second (Passed) graph displays the number of completed, successful transactions performed during each second of a scenario run. This graph helps you determine the actual transaction load on your system at any given moment. You can compare this graph to the Transaction Performance graph in order to analyze the effect of the number of transactions on the performance time.



The *x-axis* represents the elapsed time (in seconds) since the start of the scenario run. The *y-axis* represents the number of transactions successfully performed during the scenario run. For example, the above graphs indicate that in the 224th second of the scenario, nine query transactions were successfully completed. The response time at that point was 49 seconds.

Transactions per Second (Failed) Graph

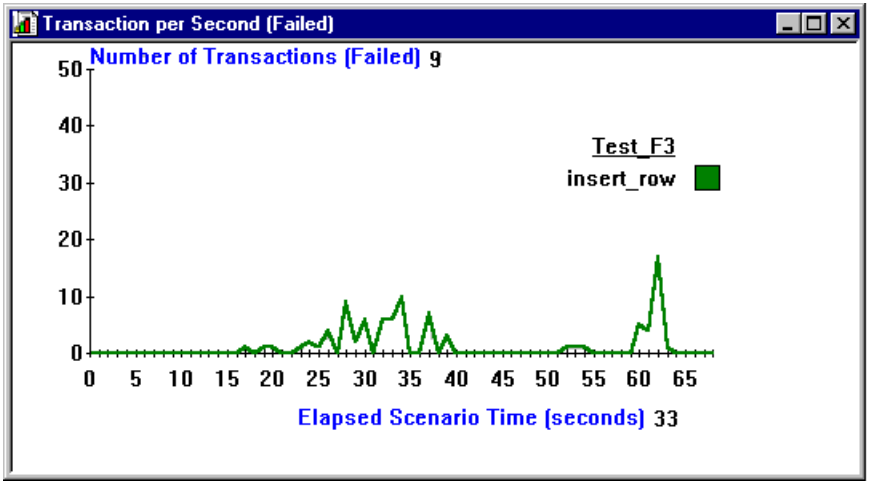
The Transactions per Second (Failed) graph displays the number of completed, unsuccessful transactions performed during each second of the scenario run. This graph contains information about transactions that were assigned an LR_FAIL value in the **lr_end_transaction** statement.

Note: The **lr_end_transaction** statement must be executed in order to generate the Failed Transaction graph. If your program aborts immediately upon an error, all current transactions are terminated—no data is generated for failed transactions.

The following example uses a Web Vuser script. Web Vuser statements return zero for success and a positive value for failure. The Web Vuser script below tries to access an HTML page using a GET statement. If the GET request succeeds, the script assigns the LR_PASS status to the transaction. If the GET request fails, the script assigns LR_FAIL to the transaction.

```
lr_start_transaction(“wizard”);
if (URL("http://www.wizard.com/index.html") == 0)
        lr_end_transaction(“wizard”, LR_PASS);
else {
        lr_end_transaction(“wizard”, LR_FAIL);
        lr_log_message("Wizard home page not returned");
        }
```

On the Transactions per Second (Failed) graph, the *x-axis* represents the elapsed time (in seconds) since the start of the scenario run. The *y-axis* represents the number of transactions unsuccessfully performed during the scenario.



For example, the above graph indicates that in the 33rd second of the scenario, nine insert_row transactions failed.

Failed Transactions Report

The Failed Transaction report provides detailed information about the beginning, end, and duration of the failed, yet completed transaction.

Sgroup: group7

Vuser id: 1

Transaction	*Start time*	*End time*	*Duration*
insert_row	09:16:14.223	09:16:14.831	00:00:00.608
insert_row	09:16:16.168	09:16:16.359	00:00:00.191
insert_row	09:16:17.733	09:16:17.980	00:00:00.247

Vuser id: 2

Transaction	*Start time*	*End time*	*Duration*
insert_row	09:16:25.320	09:16:25.788	00:00:00.468
insert_row	09:16:26.499	09:16:27.156	00:00:00.657
insert_row	09:16:28.800	09:16:29.229	00:00:00.429

Failed Vuser Report

The Failed Vuser report provides details about all Vusers that were in the ERROR, ABORTED, or DONE:FAILED states during the scenario execution. The *Ready At* and *Running At* times are relative to the computer's system clock.

Sgroup: group7

Vuser Id	*Vuser Type*	*Host*	*Ready At*	*Running At*	*Duration*	*Term Status*
4	DB	wizard	17:41:27	17:41:37	00:02:04 (124sec)	Failed
11	DB	wizard	17:41:31	17:41:37	00:01:25 (85sec)	Aborted
12	DB	wizard	17:41:30	17:41:37	00:01:29 (89sec)	Error
13	DB	wizard	17:41:32	17:41:37	00:01:29 (89sec)	Error
17	DB	wizard	17:41:32	17:41:37	00:01:39 (99sec)	Aborted
19	DB	wizard	17:41:33	17:41:37	00:01:44 (104sec)	Aborted

Summary:

Vusers: 6
Failed: 1 *Error:* 2 *Aborted:* 3

In this scenario, one Vuser failed, two had errors, and three were aborted.

Scenario Execution Report

The Scenario Execution report details the major events that occurred during the scenario run. This includes information on every Vuser, such as when it was ready to run and for how long it ran.

Sgroup: group17

Vuser Id	*Vuser Type*	*Host*	*Ready At*	*Running At*	*Duration*	*Termination St*
1	DB	wizard	11:35:17	11:44:15	00:12:00 (720sec)	Pass
2	DB	wizard	11:35:17	11:44:14	00:10:28 (628sec)	Aborted
3	DB	wizard	11:35:11	11:44:14	00:10:47 (647sec)	Aborted
4	DB	wizard	11:35:11	11:44:15	00:12:06 (726sec)	Pass
5	DB	wizard	11:35:14	11:44:14	00:12:08 (728sec)	Pass
6	DB	wizard	11:35:16	11:44:14	00:11:50 (710sec)	Pass
7	DB	wizard	11:35:15	11:44:14	00:11:58 (718sec)	Pass
8	DB	wizard	11:35:17	11:44:14	00:12:07 (727sec)	Pass
9	DB	wizard	11:35:17	11:44:14	00:12:01 (721sec)	Pass
10	DB	wizard	11:35:17	11:44:14	00:11:46 (706sec)	Pass

Summary:

Vusers: 10
Passed: 8 *Failed:* 0 *Error:* 0 *Aborted:* 2

The *Ready At* and *Running At* times are based on the computer's system clock. The *Termination Status* field indicates the status of the Vuser at the end of the scenario run (PASSED, FAILED, ERROR, or ABORTED). At the end of the Group information, there is a summary of the Group's activities.

17

Analyzing Scenario Performance

After running a scenario, you can create graphs and reports to help you locate bottlenecks and analyze overall Vuser performance. You can determine if the transaction response time was adequate, and how your system handled a load of many users.

This chapter discusses the graphs and reports that you can use to analyze Vuser performance:

- Percentile Graph
- Performance Under Load Graph
- Transaction Performance Graph
- Transaction Performance Summary Graph and Report
- Transaction Performance by Vuser Graph and Report
- Transaction Distribution Graph
- Detailed Transaction Report
- Data Point Graph and Report

About Scenario Performance

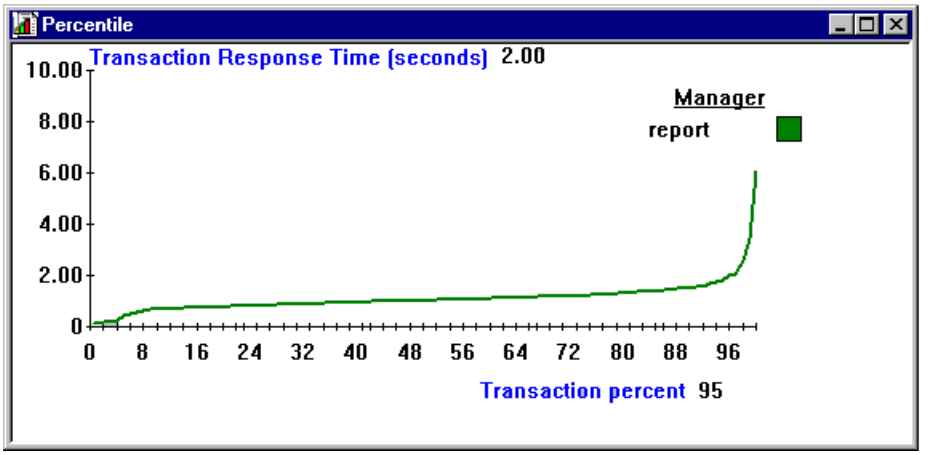
During scenario execution, Vusers generate data as they perform their transactions. Astra LoadTest allows you to generate graphs that show scenario activity or overall scenario performance. Scenario performance graphs and reports let you analyze Vuser behavior throughout the scenario. The Performance graphs map the transaction response time, and other general information about the scenario.

When working with graphs, you can focus on specific segments of the scenario using the zoom tools. In addition, you can change the granularity of the graph. When working with reports, you can export the data in ASCII format or in a custom format for common spreadsheet applications.

For more information about working with graphs and reports, see Chapter 14, "Understanding Astra LoadTest Analysis."

Percentile Graph

The Percentile graph analyzes the percentage of transactions that were performed within a given time range. This graph helps you determine the percentage of transactions that met the performance criteria defined for your system. In many instances, you need to determine the percent of transactions with an acceptable response time. The maximum response time may be exceptionally long, but if most transactions have acceptable response times, the overall system is suitable for your needs.



The *x-axis* represents the percentage of the total number of transactions measured during the scenario run. The *y-axis* represents the time taken to perform the transactions.

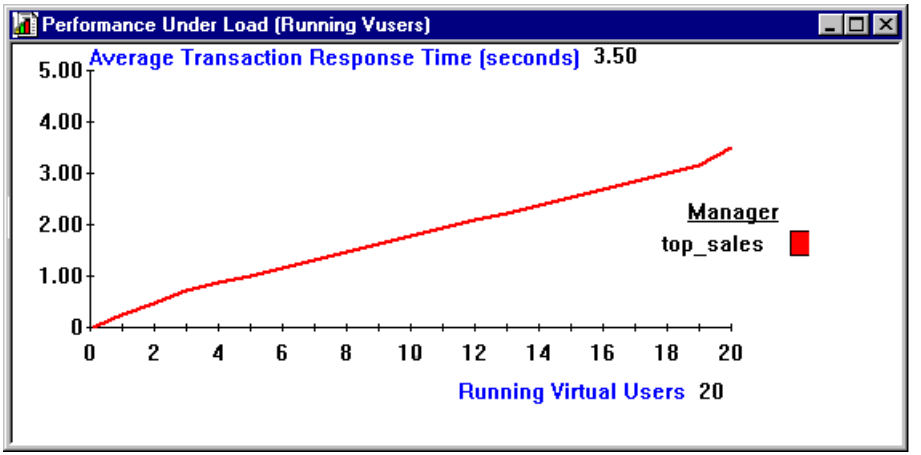
In the above example, a system administrator determines that 2 seconds is an acceptable time for a particular transaction. If 95 percent of the transactions are 2 seconds or less, the current configuration is suitable. If more than five percent of the transactions have a response time greater than fifteen, the configuration must be optimized. In the above graph, 95 percent of the *report* transactions were performed in 2 seconds or less.

Compare the Percentile graph to a graph indicating average response time such as the Transaction Performance Summary graph. A high response time for several transactions may raise the overall average. However, if the

transactions with a high response time occurred fewer than five percent of the times, that factor may be insignificant.

Performance Under Load Graph

The Performance Under Load graph indicates transaction times relative to the number of Vusers running at any given point during the scenario. This graph helps you view the general impact of Vuser load on performance time and is most useful when analyzing a scenario with a gradual load. You can create a gradual load using the scheduling options in the Vuser information box.



The *x-axis* indicates the number of running Vusers, and the *y-axis* indicates average transaction time in seconds. In the above graph, the execution time for the top_sales transaction increases with the number of running Vusers. For twenty running Vusers, the response time for the transaction was 3.5 seconds.

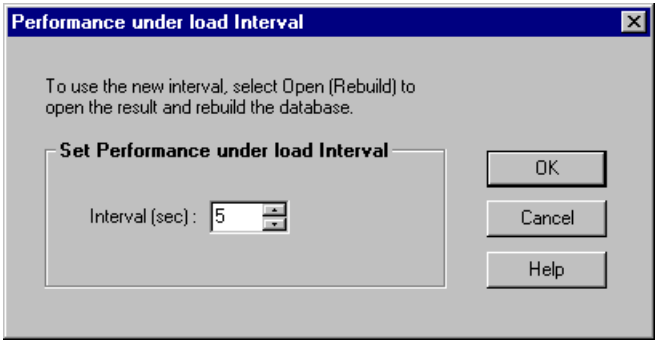
The Performance Under Load Graph indicates transaction times relative to the number of Vusers running at any given point during the scenario. In order for this graph to be meaningful, the performance is calculated when there is a stable load (constant number of running Vusers) for at least five seconds (by default). If the Vuser load is not stable for at least five seconds, the transaction time is not calculated and the graph will indicate zero.

For example, a scenario with a load of 50 to 70 Vusers had an average performance of 10 seconds, but its graph displayed zero. This occurred because the Vusers did not stabilize for five seconds.

You can instruct Astra LoadTest to measure transaction time for shorter periods of steady load.

To change the Performance Under Load interval:

1 Choose **Options** > **Performance Under Load Interval** in the Analysis window.



2 Enter the desired interval.

3 Click **OK** to accept the settings.

4 Regenerate the Performance Under Load graph.

The Performance Under Load report provides the same information as the graph for each transaction in table form.

Transaction: top_sales

Number of Running Vusers	*Performance (sec)*
3	0.71
19	3.15
20	3.51

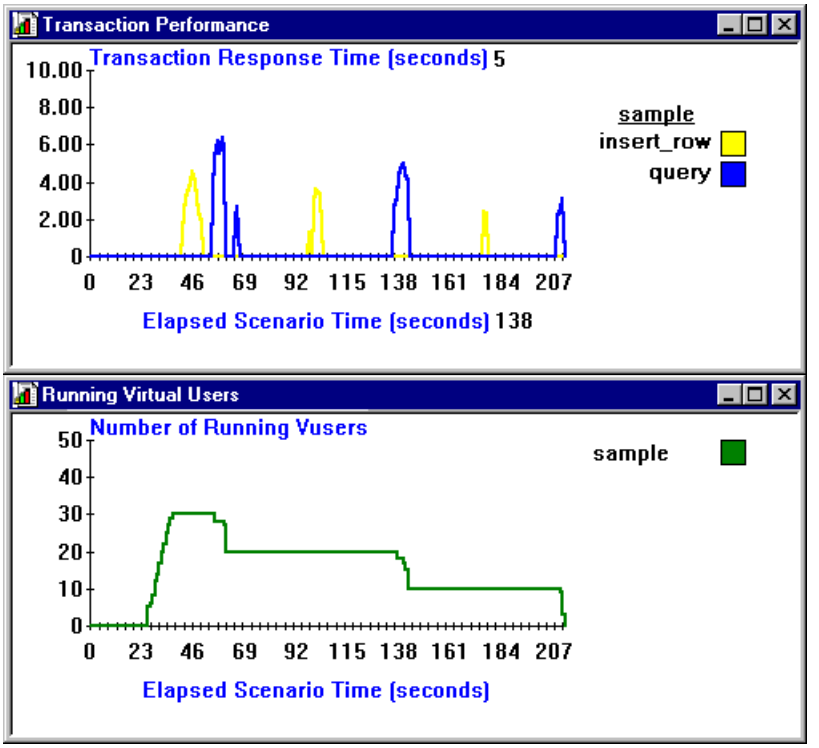
Transaction Performance Graph

The Transaction Performance graph displays the average time taken to perform transactions during each second of the scenario run.

The *x-axis* represents the elapsed time (in seconds) since the start of the scenario run. The *y-axis* represents the average time (in seconds) taken to perform each transaction.

You can compare the Transaction Performance graph to the Running Virtual Users graph to see how the number of running Vusers affects the transaction performance time.

If you have defined acceptable minimum and maximum transaction performance times, you can use this graph to determine whether the performance of the server is within the acceptable range.

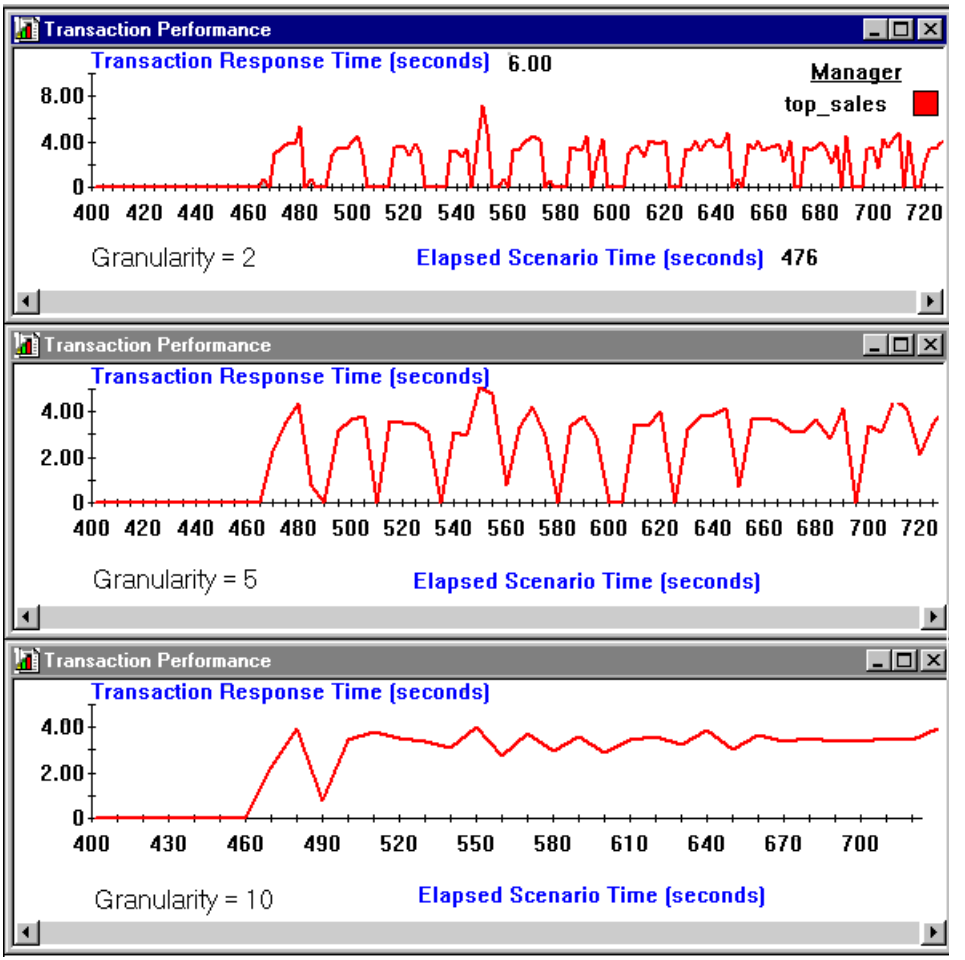


For instance, the graph above shows the transaction performance times for two transactions. You can see that each time the transaction was executed, performance time decreased. If you compare this graph to the Running Virtual Users graph, you can see that the performance time improved because the Vuser load decreased.

Note that the transaction times are only indicated by the peaks in the graph. When running multiple Vusers, one transaction may begin before the previous one ends, resulting in a continuous line. The line does not indicate the actual start and end times of transactions; it is only a mechanism used by the graph to join the peaks.

The Transaction Performance graph may be displayed using different granularities. The y-axis represents the average transaction response time within the granularity interval. For example, for a granularity of 1, the *y-axis* shows the average response time for each one second period of the scenario.

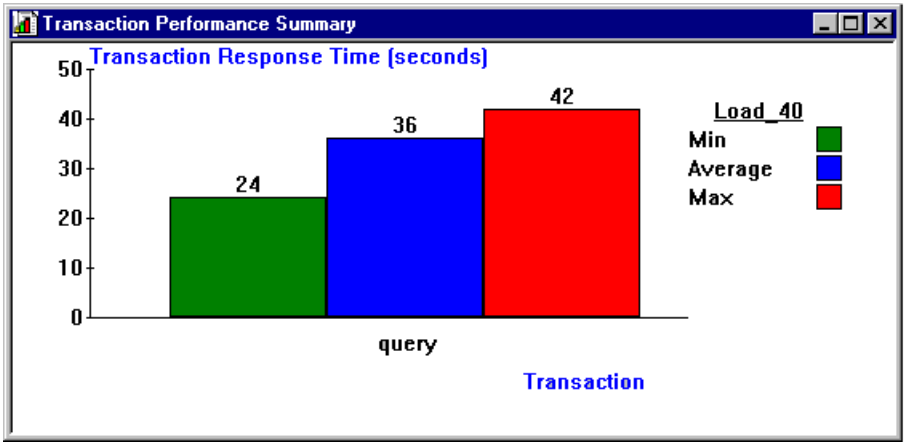
For a granularity of 2, the *y-axis* shows the average response time for every two second period of the scenario.



In the above graphs, the same scenario results are displayed in a granularity of 2, 5, and 10 (from top to bottom). The lower the granularity, the more detailed the results. However, it may be useful to view the results with a higher granularity to study the overall Vuser behavior throughout the scenario. For example, using a low granularity as in the upper graph, you see the intervals when no transactions were performed. By viewing the same graph with a higher granularity, you can easily see that the overall transaction response time averaged 3.5 seconds.

Transaction Performance Summary Graph and Report

The Transaction Performance Summary graph displays the minimum, maximum and average performance time for all the transactions in the scenario.



The *x-axis* specifies the name of the transaction. The *y-axis* shows the time, rounded off to the nearest second, taken to perform each transaction.

For example, the above graph displays the statistics of the query transaction. The transaction was performed in a minimum time of 24 seconds, an average time of 36 seconds, and a maximum time of 42 seconds.

The report shows similar information in table form, with several additional statistical details.

Duration: 00:07:27 (447sec)

			Performance (sec)				
Transaction	*Pass*	*Fail*	*Min*	*Avg*	*Max*	*90th Percentile*	*PopStdDev*
lr_change_list_in_reference	10	0	0.05	0.08	0.17	0.11	0.038
lr_copy_items	10	0	0.06	0.47	0.88	0.49	0.232
lr_insert_claim_data	10	0	30.37	30.63	31.04	30.76	0.236
lr_insert_claim_details	10	0	10.71	12.74	20.70	14.34	2.972
lr_insert_memo_data	9	0	0.17	1.02	3.68	3.18	1.294
lr_insert_new_cust_data	10	0	0.05	0.22	0.28	0.27	0.066
lr_insert_request_data	10	0	0.50	1.11	2.09	1.48	0.491
lr_login_process	10	0	43.17	46.87	52.79	49.10	3.220

The following values are reported:

Pass, Fail: the number of Vusers that passed or failed the transaction.

Min, Avg, Max: the minimum, average, and maximum transaction times.

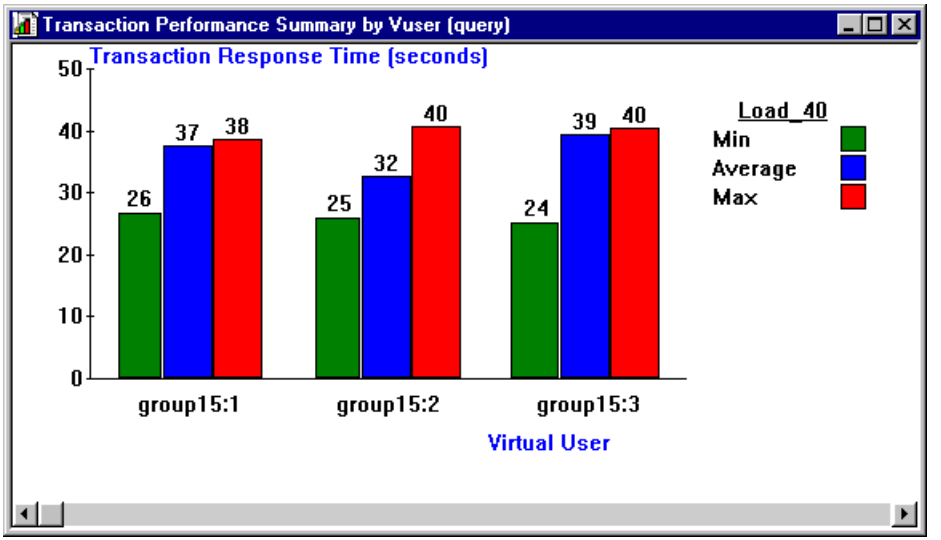
90th Percentile: the mean transaction time for ninety percent of the transactions.

PopStdDev: The popular standard deviation of transaction times from the average value.

Note that the data in the report is not rounded off to the nearest second as it is in the graph.

Transaction Performance by Vuser Graph and Report

The Transaction Performance Summary by Vuser graph displays the time taken by an individual Vuser to perform transactions during the scenario. The minimum, maximum, and average times are plotted for each transaction performed by the Vuser. This graph is useful when you have different types of Vusers in a scenario, and you want to characterize performance for each type.



The *x-axis* specifies the name of the Vuser and the Group to which it belongs. The *y-axis* shows the time, in seconds, it takes to perform each transaction. For example, the above graph displays transaction processing times for the Vusers in the group15. Vuser two performed the query transaction in a minimum of 25 seconds, an average of 32 seconds, and a maximum of 40 seconds.

The Performance Summary by Vuser report shows similar information for each Vuser in table format.

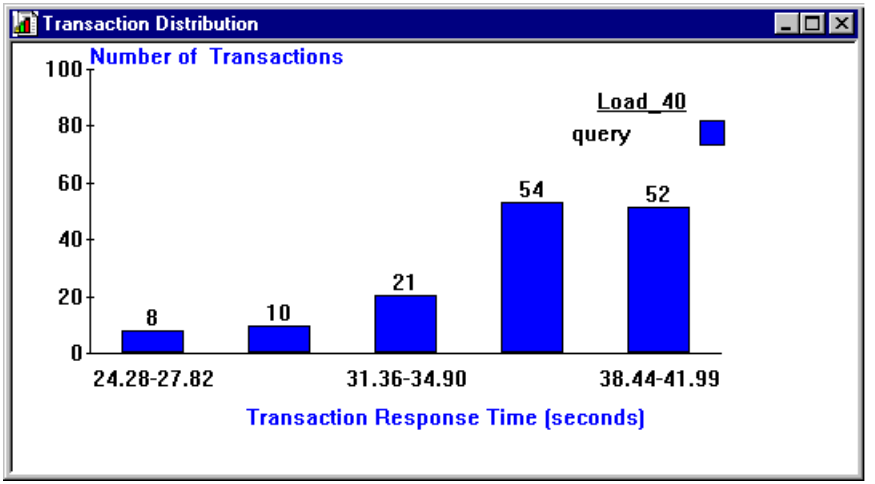
Transaction: query

Sgroup: group15

Vuser id	Pass	Fail	Performance (sec) Min	Avg	Max
1	3	0	26.01	37.39	38.46
2	3	0	25.37	32.48	40.40
3	3	0	24.46	39.03	40.22
4	3	0	24.87	38.16	40.87
5	3	0	25.33	38.26	38.52
6	3	0	25.45	37.95	39.30
7	3	0	26.56	30.57	40.03
8	3	0	25.77	37.48	40.27
9	3	0	25.54	35.34	40.85
10	3	0	24.98	31.25	37.31

Transaction Distribution Graph

The Transaction Distribution graph displays the distribution of the time taken to perform a transaction. The transaction response times are divided into five *bars*. Each bar shows how many times the transaction was completed within a given time range. If you compare it to the Transaction Performance Summary graph, you can see how the average performance was calculated. Note that this graph only displays information for a single transaction.



The *x-axis* represents the time range, in seconds, to perform the selected transaction. The *y-axis* represents the number of transactions executed during the scenario.

For example, the above graph indicates that the performance time for 52 query transactions was between 38.44 and 41.99 seconds.

Detailed Transaction Report

The Detailed Transaction report (by Vuser) is a list of all transactions executed by each Vuser during a scenario. The report provides details about the execution time of each transaction per Vuser.

Group: zorb

Vuser_id: 1

Transaction	*Start time*	*End time*	*Duration*	*Think Time*	*Wasted Time*	*Result*
end_section	14:07:59.029	14:08:00.045	00:00:01.016	00:00:01.016	00:00:00.000	Pass
General	14:05:27.748	14:06:06.482	00:00:38.618	00:00:21.715	00:00:00.116	Pass
General	14:06:06.482	14:07:02.701	00:00:37.104	00:00:21.716	00:00:19.115	Pass
General	14:07:02.701	14:07:59.029	00:00:37.212	00:00:21.715	00:00:19.116	Pass
mc_run	14:05:37.310	14:05:41.764	00:00:04.453	00:00:02.609	00:00:00.001	Pass
mc_run	14:06:16.045	14:06:20.248	00:00:04.202	00:00:02.609	00:00:00.001	Pass
mc_run	14:07:12.264	14:07:15.920	00:00:03.655	00:00:02.609	00:00:00.001	Pass

The following values are reported:

Start time: the system time at the beginning of the transaction

End time: the actual system time at the end of the transaction, including the think time and wasted time.

Duration: The duration of the transaction in the following format: hrs:minutes:seconds:microseconds. This value includes think time, but does not include wasted time.

Think time: the Vuser's think time delay during the transaction.

Wasted time: the Astra LoadTest internal processing time not attributed to the transaction time or think time.

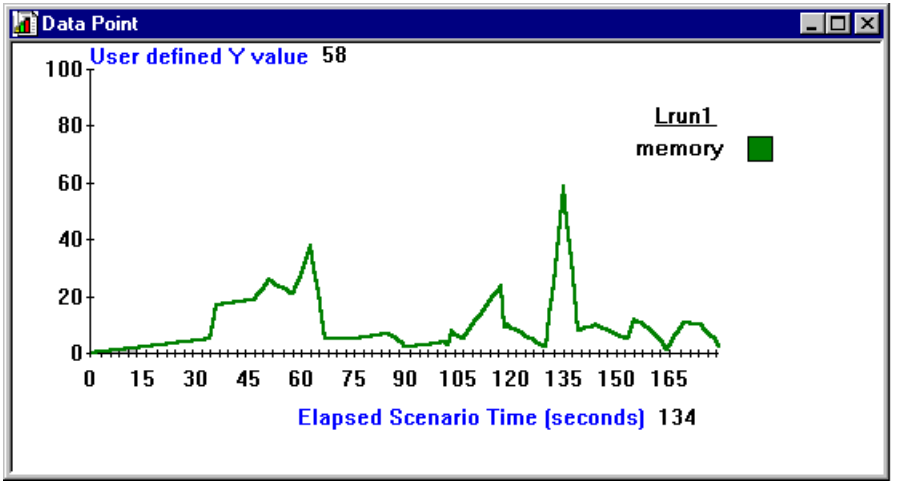
Results: the final transaction status, either Pass or Fail.

Data Point Graph and Report

Astra LoadTest enables you to record your own data for analysis. You instruct Astra LoadTest to record the value of an external function or variable, also known as a *data point*, during the scenario run. Using the gathered data, Astra LoadTest creates a graph and report for the data point.

The data point is set by including an **lr_user_data_point** function in your Vuser script. For more information, refer to the *Online Function Reference*.

The Data Point graph shows the value of the data point during the scenario run. The *x*-axis represents the number of seconds elapsed since the start time of the run. The *y-axis* displays the value of each recorded data point statement.



In the above example, an **lr_user_data_point** statement was placed in a Vuser script, defining a data point called *memory*. The data point is associated with an external function that checks the computer's virtual memory. The above graph indicates that the value of the function was 58 at the 134th second of the scenario. You can compare this graph with other graphs to understand why and when the peaks occurred. The peaks could indicate a rendezvous, or a transaction that uses a great deal of memory.

The Data Point report lists the name of the data point, its value, and the time its value was recorded. The values are displayed for each Group and Vuser.

Sgroup: Group1

		Data Point	Time
Vuser id:	1		
	memory	19.00	13:37:16
	memory	1.00	13:37:20
	memory	9.00	13:37:32
	memory	1.00	13:37:36
	memory	1.00	13:37:40
Vuser id:	2		
	memory	6.00	13:37:05
	memory	8.00	13:37:20
	memory	9.00	13:37:32
	memory	1.00	13:37:36
	memory	1.00	13:37:40

18

Cross-Scenario Analysis

After running a scenario, you can analyze the performance of your client/server system using Astra LoadTest's graphs and reports. Cross-scenario graphs allow you to compare the results of several scenario runs by superimposing their results onto a single graph.

This chapter discusses:

- Cross-Scenario Graph Types
- Generating Cross-Scenario Analysis Graphs
- Performing Cross-Scenario Analysis
- Cross-Scenario Analysis for Transaction Performance
- Cross-Scenario Analysis for Performance Summary
- Cross-Scenario Analysis with Percentile Graphs

About Cross-Scenario Analysis

Cross-scenario analysis enables you to compare the results of multiple scenario runs. Astra LoadTest can superimpose the generated results onto a single cross-scenario graph. On this graph, you can compare the performance of the server during multiple scenarios. Cross-scenario analysis is useful for:

- benchmarking hardware
- testing software versions
- determining system capacity

Suppose that you want to benchmark two hardware configurations. You run the same scenario with both configurations and compare the transaction response times using a single cross-scenario graph.

Suppose that your vendor claims that a new software version is optimized to run quicker than a previous version. You can verify this claim by running the same scenario on both versions of the software, and comparing the scenario results.

You can also use cross-scenario graphs to determine your system's capacity. You run scenarios using different numbers of Vusers running the same script. By analyzing the cross-scenario graphs, you can determine the number of users that cause unacceptable response times.

Cross-Scenario Graph Types

You can perform cross-scenario analysis for the following graph types:

Graph Name	Type
Running Virtual Users	Activity
Rendezvous	Activity
Transactions per Second (Passed)	Activity
Transactions per Second (Failed)	Activity
Percentile	Performance
Performance Under Load	Performance
Transaction Performance	Performance
Transaction Performance Summary	Performance
Transaction Performance Summary by Vuser	Performance
Transaction Distribution	Performance
Connections per Second	Web
Throughput	Web

For information about the Activity graphs, see Chapter 16, “Analyzing Scenario Activity.” For information about the Performance graphs, see Chapter 17, “Analyzing Scenario Performance.” For information about the Web graphs, see Chapter 19, “Web Vuser Graphs.”

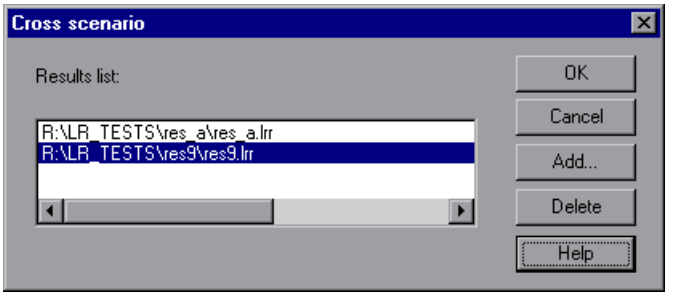
Generating Cross-Scenario Analysis Graphs

You can perform cross-scenario analysis for two or more graphs.

To perform cross-scenario analysis:

1 Select **Results** > **Analyze Results**, or click the Analysis button to open the Analysis window.

2 Select **File** > **Cross Scenario Graphs**, or click the **Cross-Scenario** button. The Cross Scenario dialog box opens. You use the dialog box to select the scenarios to compare.

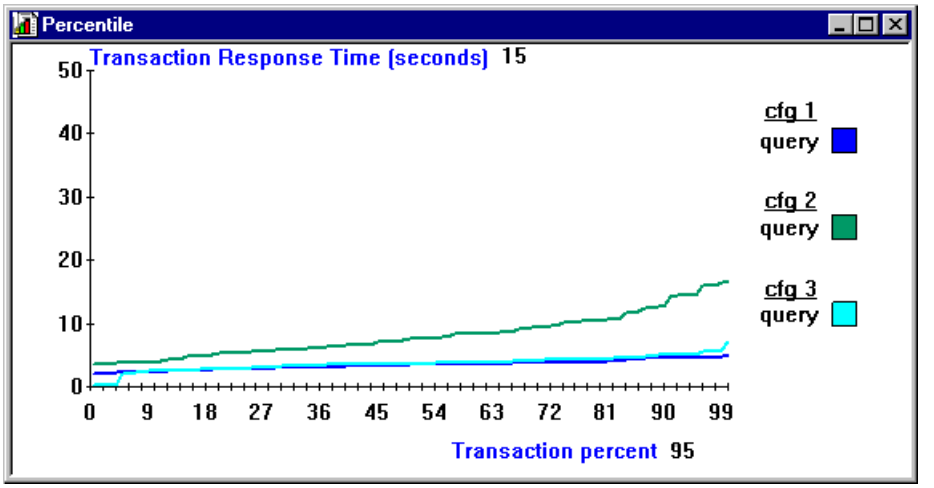


3 Click **Add** to add scenarios to the Results list. The Open dialog box opens.

4 In the Open dialog box, select a results directory and choose a results file (*.lrr*). Click **OK**. The scenario is added to the Results list.

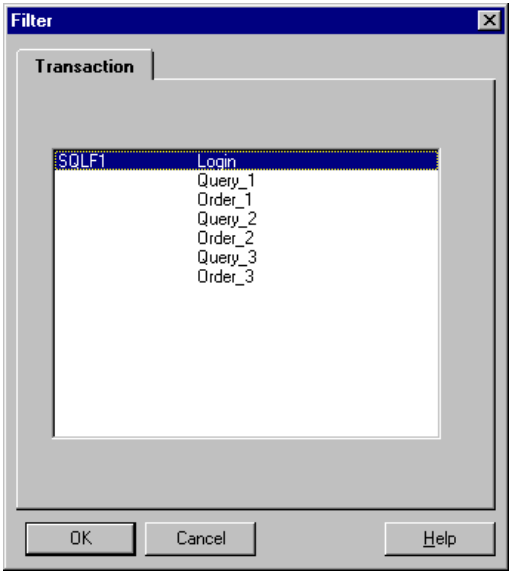
5 Repeat steps three and four until all the scenarios you want to compare appear in the Results list.

6 Click **OK**. The Percentile graph appears. The information for each scenario is plotted in a different color.



7 Select additional graphs from the Graphs menu.

8 Filter the scenarios and transactions you want to view. Choose multiple transactions using the CTRL key.



9 Manipulate the graphs using the analysis tools. For example, you can tile graphs to compare their results, change the granularity, and zoom in on a specific area.

Performing Cross-Scenario Analysis

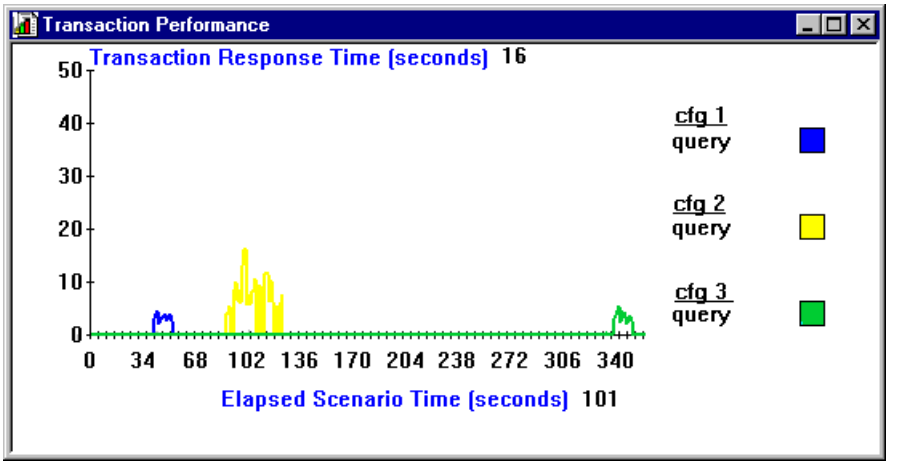
Astra LoadTest cross-scenario analysis graphs allow you to compare results and performance data from various scenarios on a single graph. You can filter graphs to display selected scenarios, transactions, or Vusers. By comparing the scenario results, you can determine which Vusers had acceptable response times and what event caused a bottleneck. If you are benchmarking machines, you can determine if a new server justifies its cost by comparing the transaction response times.

The following sections illustrate the use of cross-scenario analysis using three cross-scenario graphs:

- Transaction Performance
- Transaction Performance Summary
- Percentile

Cross-Scenario Analysis for Transaction Performance

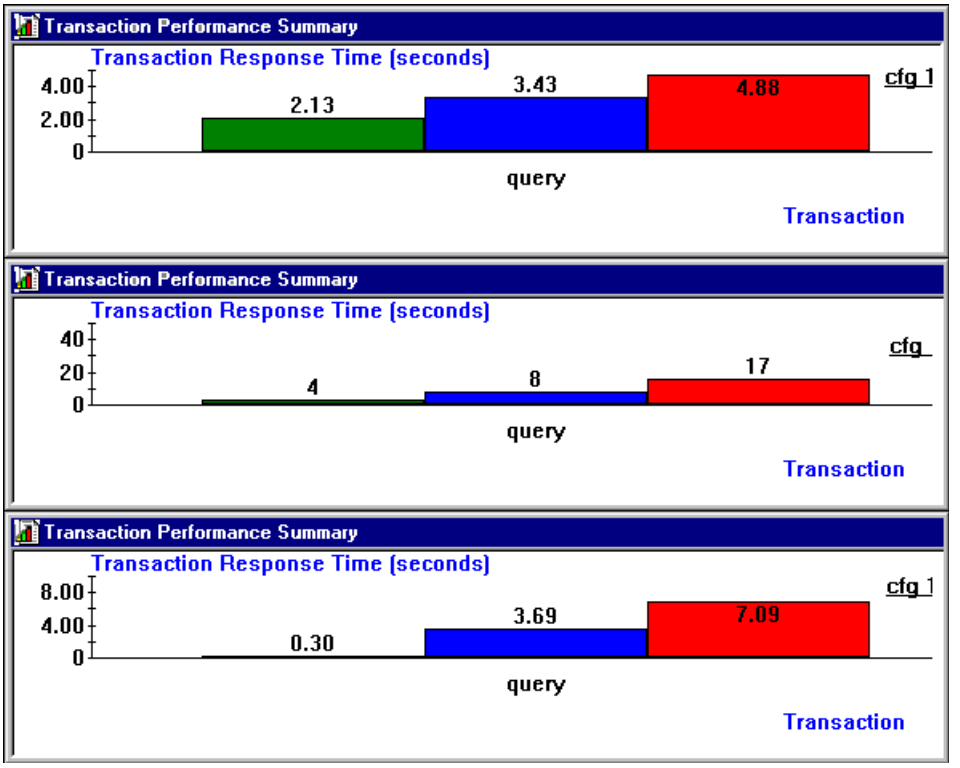
The Transaction Performance graph displays the average time taken to perform transactions during each second of the scenario run. Cross-scenario graphs are useful in comparing user behavior after a configuration change such as additional memory or CPUs. You run an identical scenario with the same number of Vusers for each configuration. Afterwards you compare the transaction results using cross-scenario analysis.



In the above example, the operator ran the same scenario three times, using three different configurations. The transaction response time was substantially longer in the second configuration. The transaction time for the second configuration was slightly better than the third. The graph also shows that the loading time in the third configuration was greater than the previous configurations.

Cross-Scenario Analysis for Performance Summary

Using the Transaction Response Time cross-scenario graphs, you can determine and compare the minimum, maximum, and average transaction times for your scenarios. These graphs only display statistics for a single transaction. To compare multiple transactions, regenerate the Transaction Response Time graph several times and select **View** > **Filter** to select a transaction to display. Select **Window** > **Tile** to align the axis of the graphs. Note that the scale of the *y-axis* may differ between graphs.



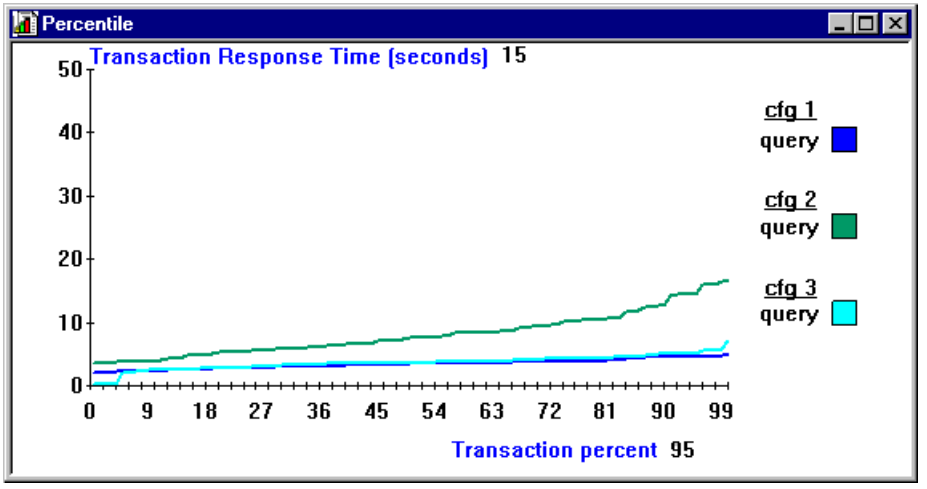
In the above graphs, the average transaction response time ranged from 3.43 seconds in the first configuration, to eight seconds in the third configuration. To determine the Vuser distribution—for example how many Vusers performed a transaction in the average time of 3.69 seconds, use the Transaction Distribution graph.

These graphs enable you to see that although the minimum transaction time was lower in the third configuration, the average response time was lower in the second configuration.

Cross-Scenario Analysis with Percentile Graphs

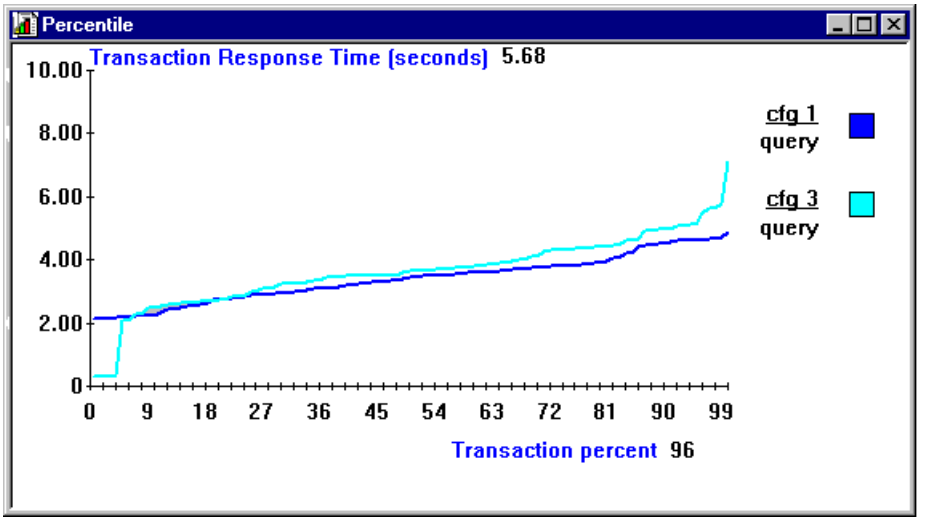
The Percentile graph analyzes the percentage of transactions that were performed within a given time range. Use the Percentile cross-scenario graph to compare the transaction response times between scenario runs.

For example, to compare transaction response time for several hardware configurations, you run the same scenario for each configuration. Run the cross-scenario analysis to display the transaction response times.



The above graph indicates that a large percentage of the transactions had a better response time in the third configuration than in the second configuration.

If we filter the graph to display only the first and third configurations, the *y-axis* scale shows that the response time for the first configuration is slightly better than the third.



19

Web Vuser Graphs

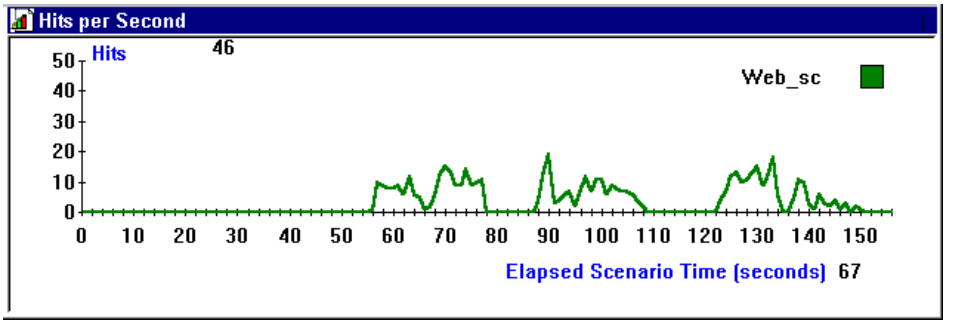
After a scenario run, you can use graphs to analyze web server performance. Web Vuser graphs can help you analyze the behavior and performance of your web site. They display information about server load, transaction performance time, hits, and throughput.

This chapter describes:

- Hits per Second Graph
- Throughput Graph
- Comparing Web Graphs

Hits per Second Graph

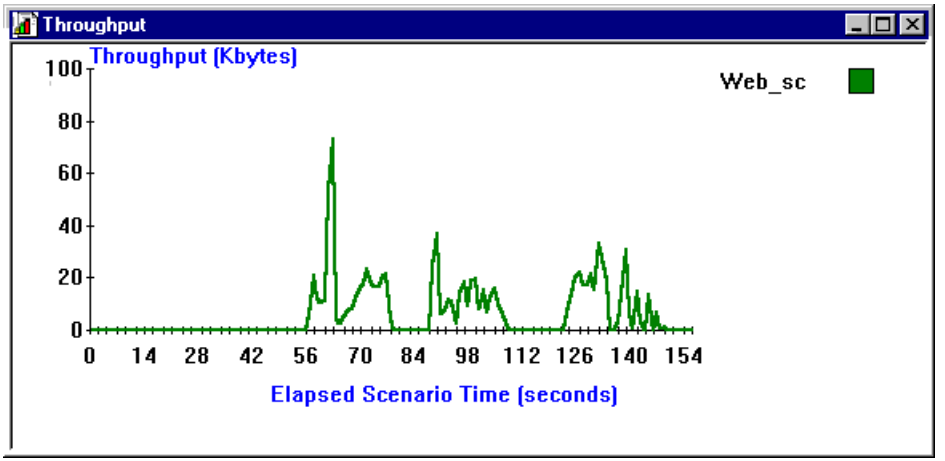
The Hits per Second graph shows the number of hits made on the web server by Vusers during each second of the scenario run. This graph helps you evaluate the amount of load Vusers generate, in terms of the number of hits. You can compare this graph to the Transaction Performance graph to see how the number of hits affects transaction performance.



The *x-axis* represents the number of seconds elapsed since the start of the scenario run. The *y-axis* represents the number of hits on the server. For example, the graph above shows how the number of hits decreased gradually throughout the scenario. The most hits per second took place during the 90th second.

Throughput Graph

The Throughput graph shows the amount of throughput on the server during each second of the scenario run. Throughput is measured in kilobytes and represents the amount of data that the Vusers received from the server at any given second. This graph helps you to evaluate the amount of load Vusers generate, in terms of server throughput. You can compare this graph to the Transaction Performance graph to see how the throughput affects transaction performance.



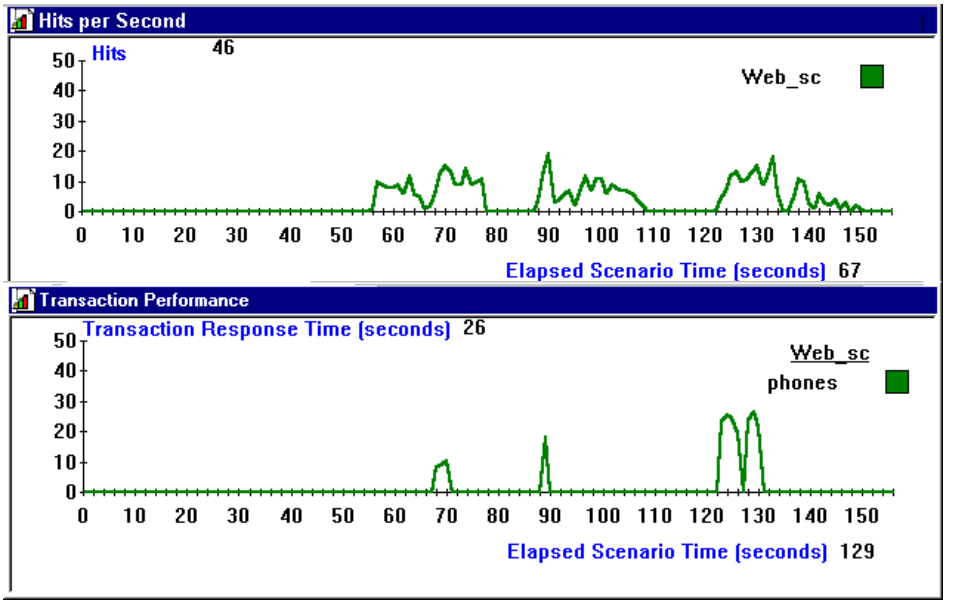
The *x-axis* represents the number of seconds elapsed since the start of the scenario run. The *y-axis* represents the throughput of the server, in kilobytes. For example, in the graph below, you can see that the highest throughput was about 75,000 bytes, during the 60th second of the scenario.

Comparing Web Graphs

You can improve your scenario performance by analyzing a specific transaction to locate the bottlenecks. By comparing several graphs, you can view the maximum transaction time and determine why the transaction time reached its maximum at that point.

To view and compare several graphs simultaneously, generate the graphs within the analysis window and select **Window** > **Tile**.

In the following example, the maximum transaction response time was 26 seconds. This occurred when the number of connections was at its peak. This graph clearly indicates that the response time is influenced by the number of hits.



You can compare other graphs and report results to analyze the scenario and determine additional bottlenecks.

Part VI

Appendixes

A

Troubleshooting the Controller

Astra LoadTest enables you to test entire client/server networks. If one of the components of the client/server system is not configured properly, Astra LoadTest scenarios will not run.

This appendix discusses the most common Astra LoadTest problems:

- Failure to Communicate with a Host
- Failure to Connect to the AUT Database
- Failure to Access Files
- Failed Vusers or Transactions
- Updating the IDAPI Settings
- Increasing the Number of Vusers on an NT Server
- Working with Firewall Limitations

About Troubleshooting

Astra LoadTest relies heavily upon communication between host machines on a network. If communication is not established properly, the Controller will be unable to send commands to remote hosts 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 hosts as a stand-alone:

➤ Test all types of Vuser scripts on Windows platforms by running them from VuGen.

For more information on running Vuser scripts in stand-alone mode, refer to the appropriate guide for creating Vuser scripts.

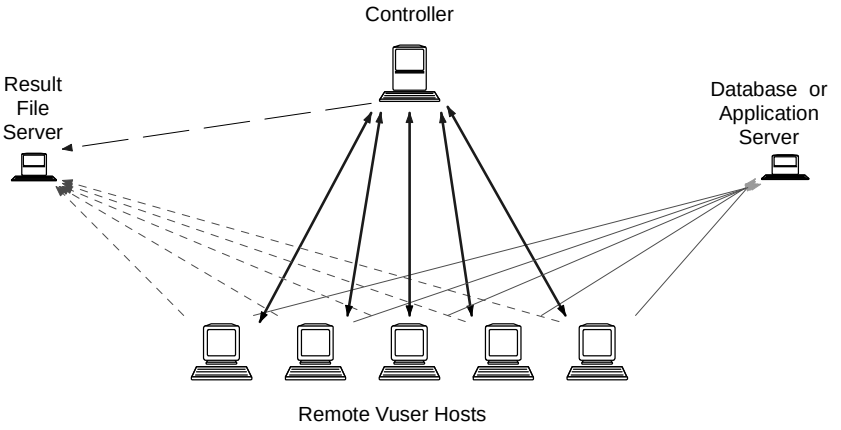
Astra LoadTest Communications

Most communication problems can be solved if you understand your Astra LoadTest 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 Astra LoadTest. There are three servers: The Astra LoadTest Controller, the application or 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 hosts, 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 application server in both directions (send/receive). In order for a client host

to connect to the server host, it must be able to resolve the server host name.



If any of the connections are broken, the scenario will fail.

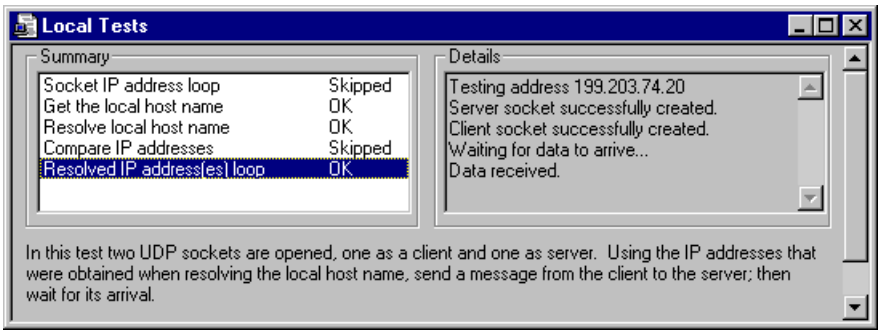
Failure to Communicate with a Host

The most common communication error is the failure of the Controller host to connect with a remote host. Check the following items:

- TCP/IP setup
- TCP/IP connectivity
- Remote Command Launcher
- Host connections

Checking TCP/IP Setup

The first step in checking your configuration is to verify your machine's TCP/IP setup. Astra LoadTest includes a utility called Hostinfo (*hostinfo.exe*), located under Astra LoadTest's bin directory. This utility provides information about the current host—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 host 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 host name is the same as the actual IP address of this host.

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 host 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 host machines. Use a ping utility or type PING *hostname* from the DOS command line to verify communication with a remote machine. Make sure that the remote host and Controller machines can ping each other by IP addresses and hostnames.

Remote Command Launcher

All remote hosts running on a Windows platform, use a Astra LoadTest Agent to accept commands from Astra LoadTest. The Agent is started through the Remote Command Launcher placed in you StartUp group during installation. An active Remote Command Launcher is indicated by an icon (on the task bar in Windows NT 4.0). If the Remote Command Launcher is not active, invoke it from the Astra LoadTest program group. If you upgraded from an earlier version of Astra LoadTest, check that all remote hosts have an updated version of the Remote Command Launcher.

Host Connections

To verify the host connectivity, connect to each one of the remote hosts from the Controller's Host window. Select the host(s) and click the Connect button. The status changes to *Connecting*.

If the Connection fails, the status changes to *Fail* 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 WIN95, WINNT, 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.

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 *hostname* from the DOS command line.

SQL Utilities: Use a simple utility such as ISQL or SQLPLUS to logon to the database server and perform several basic operations.

Failure to Access Files

A Astra LoadTest scenario will fail if the result path or Vuser script is inaccessible to one or more of the participating hosts. Check the following items:

➤ Vuser Script

➤ Result Path

Vuser Script: Make sure that the Vuser script is accessible to all hosts participating in the scenario through path translation and permissions. View or run the Vuser script as a stand-alone on each of the participating hosts.

Result Path: Make sure that the result path is accessible to all hosts 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

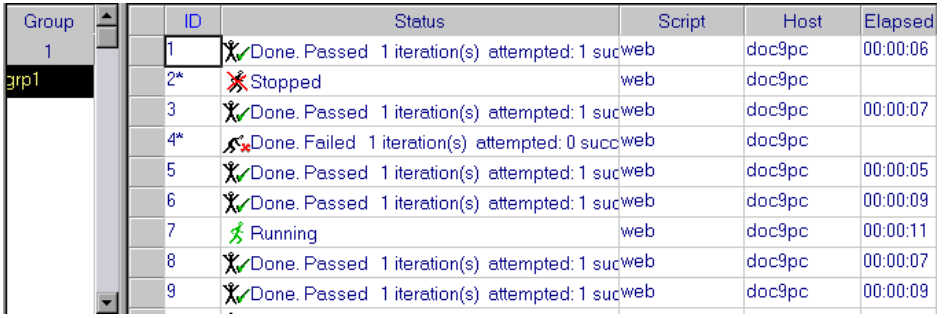
Astra LoadTest Vusers or transactions may fail for a variety of reasons relating to the network, database, or actual script. You can find information about scenario runs from the following sources:

- Vusers view
- Output window
- Output file
- Analysis reports and graphs

Vusers View

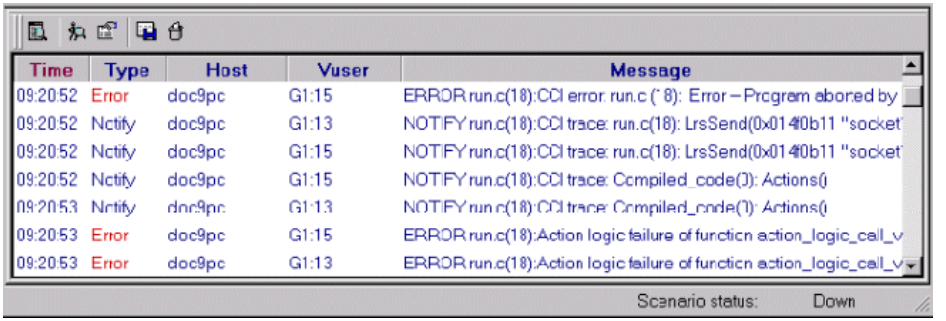
You view the Vusers window from the Astra LoadTest Controller. The Status column indicates the status of the Vuser during and after the scenario run. During the scenario run, the column will show a PENDING, INITIALIZING, READY, RUNNING, RENDEZVOUS, or PAUSED status. If a Vuser fails and does not complete the script execution, Astra LoadTest displays an error status. If a Vuser completes the script some of the transactions may have failed. Astra LoadTest 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 10, "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: error, warning, 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 11, "Viewing Vusers During Execution."

Output File

For non-GUI Vusers, 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 dumped to the output file, depends on the output file settings. In VuGen's runtime 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

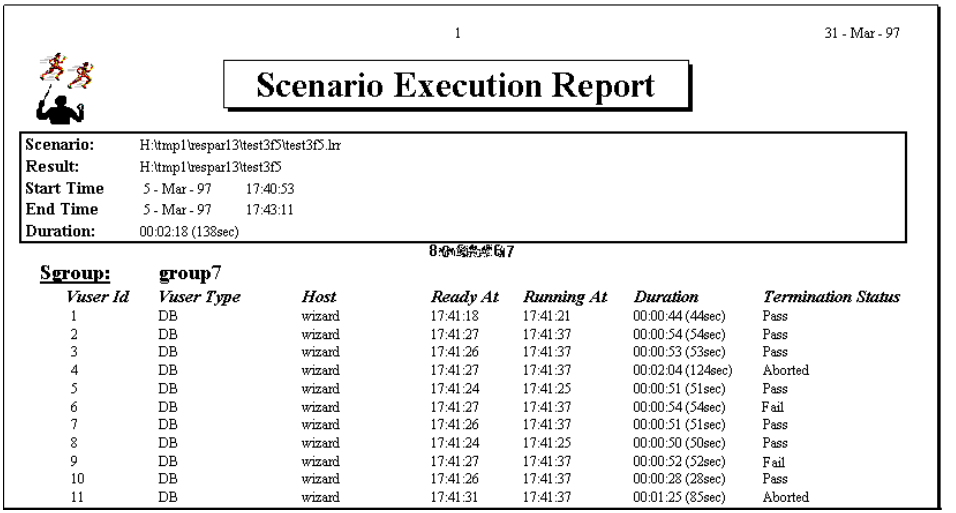
introduces extra overhead. For more information about configuring run-time settings, refer to the *Virtual User Recorder* guide.

The following example is a segment of a log file generated during the recording of a database application in Brief mode.

```
Start vuser_init.
r:\lr_tests\odbc_f3\jtestf\init.c (16): lrd_open_connection: User="scott",
Server="hammer"r:\jtestf\init.c (24):
r:\lr_tests\odbc_f3\jtestf\init.c (44): lrd_stmt: SELECT * FROM DUAL
r:\lr_tests\odbc_f3\jtestf\init.c (47): lrd_fetch: 1 row(s) fetched
r:\lr_tests\odbc_f3\jtestf\init.c (81): lrd_stmt: commit
End vuser_init. rc = 0
Running Vuser...
Start Actions.
Transaction insert_row started.
r:\lr_tests\odbc_f3\jtestf\run.c (10): lrd_stmt: insert into test2 values (’Jim’,102)
Transaction insert_row ended successfully.
At rendezvous: query_rnd.
Transaction query started.
r:\lr_tests\odbc_f3\jtestf\run.c (22):  lrd_stmt: select name, id from test2
r:\lr_tests\odbc_f3\jtestf\run.c (26):  lrd_fetch: 1170 row(s) fetched
Transaction query ended successfully.
```

Analysis Reports and Graphs

You can generate graphs and reports to view information about the scenario run. For example, the Scenario Execution report lists information about the termination status of each Vuser: PASS, FAIL, or ABORTED.



1 31 - Mar - 97

Scenario Execution Report

Scenario:	H:\tmp1\respar13\test3f5\test3f5.lrr
Result:	H:\tmp1\respar13\test3f5
Start Time	5 - Mar - 97 17:40:53
End Time	5 - Mar - 97 17:43:11
Duration:	00:02:18 (138sec)

Sgroup: group7

Vuser Id	*Vuser Type*	*Host*	*Ready At*	*Running At*	*Duration*	*Termination Status*
1	DB	wizard	17:41:18	17:41:21	00:00:44 (44sec)	Pass
2	DB	wizard	17:41:27	17:41:37	00:00:54 (54sec)	Pass
3	DB	wizard	17:41:26	17:41:37	00:00:53 (53sec)	Pass
4	DB	wizard	17:41:27	17:41:37	00:02:04 (124sec)	Aborted
5	DB	wizard	17:41:24	17:41:25	00:00:51 (51sec)	Pass
6	DB	wizard	17:41:27	17:41:37	00:00:54 (54sec)	Fail
7	DB	wizard	17:41:26	17:41:37	00:00:51 (51sec)	Pass
8	DB	wizard	17:41:24	17:41:25	00:00:50 (50sec)	Pass
9	DB	wizard	17:41:27	17:41:37	00:00:52 (52sec)	Fail
10	DB	wizard	17:41:26	17:41:37	00:00:28 (28sec)	Pass
11	DB	wizard	17:41:31	17:41:37	00:01:25 (85sec)	Aborted

The transaction related graphs and reports only list information for successful transactions. For information on failed transactions, use the Transaction per Second (Failed) graph, the Failed Transaction report, or the Failed Vuser report. For more information on the available graphs and reports, see Chapter 14, "Understanding Astra LoadTest Analysis."

Updating the IDAPI Settings

Astra LoadTest uses the Borland Database Engine (BDE) during performance analysis. When you start the Analysis tool, Astra LoadTest checks that the BDE–IDAPI settings on your computer are correct. To do so, Astra LoadTest verifies that the locations are set for the required DLLs and the *.cfg* file. The locations are set in the IDAPI section of your *win.ini* file (16-bit operating systems) or in the Registry (32-bit operating systems).

The IDAPI settings may become corrupted. For example, this may occur if you uninstall another application that uses BDE. You can update the IDAPI settings from the Analysis window.

To update your IDAPI settings:

1 Select **Analysis** > **Analysis** or click the **Analysis** button. The Analysis window opens.

2 Select **File** > **IDAPI Setting**. A message is displayed warning you that the current IDAPI settings will be overwritten.

3 Click **OK**. Astra LoadTest updates the IDAPI settings.

Increasing the Number of Vusers on an NT Server

Under the normal settings of an NT server, 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 NT operating system, modify the NT 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 an NT server using 2 Pentium PRO 200 MHz with 1 GB RAM.

Since each Vuser in this setup used approximately 2MB memory, other Vusers may require more memory.

We were able to load over 2500 Vusers when we ran the NT terminal server as the Operating System and changed the above registry setting.

The above registry changes enable us to run more threads allowing us to run more Vusers on the machine. This implies that we are not bound by the NT operating system, but only 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.

Astra LoadTest uses *rcmdl* for launching the Agents (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 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. A firewall in this case also poses a problem: the Vusers will not be able to run the client program at all.

B

Configuring Multiple IP Addresses

When you run a scenario, the Vusers on each host machine use the host's IP address. You can define multiple IP addresses on a host machine to emulate a real-life situation where users sit on different machines.

This appendix describes:

- Adding IP Addresses to a Host
- Using the IP Wizard
- Updating the Routing Table

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 host machine have the same IP address, server and router optimizations do not reflect real-life situations.

Astra LoadTest'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 Host

Astra LoadTest includes an IP Wizard program that you run on each host machine to create multiple IP addresses. You add new IP addresses to a machine once and use the addresses for all scenarios.

The following procedure summarizes how to add new IP addresses to a host:

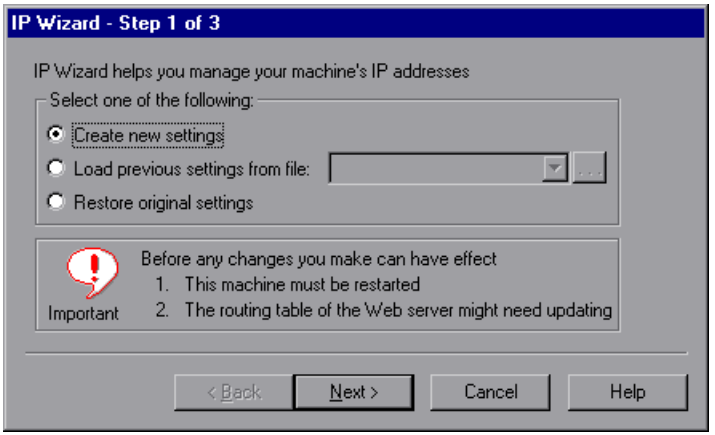
1. Run the IP Wizard on the host machine to add a specified number of IP addresses.
2. Re-start the host machine.
3. Update the server's routing table with the new addresses, if necessary.
4. Enable this feature from the Controller. Refer to "Enabling Multiple IP Addressing from the Controller", on page 208.

Using the IP Wizard

The IP Wizard resides on each host machine. You run this process one time to create and save new IP addresses. 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 are the default used by the IP Wizard.

To add new IP addresses to a host machine:

1 Invoke the IP Wizard from the **Tools** submenu of the Astra LoadTest program group.

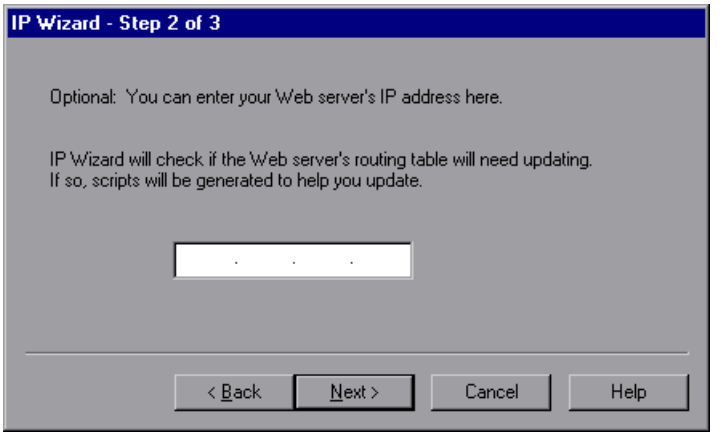


2 If you have an existing file with IP address settings, select **Load settings from file** and choose the file.

3 If you are defining new settings, select **New Settings**.

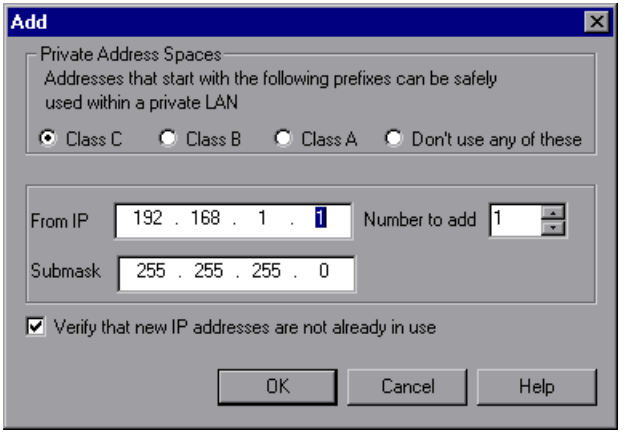
4 Click **Next** to proceed to the next step. If you have more than one network card, choose the card to use for IP addresses and click **Next**.

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 host.



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 208 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.



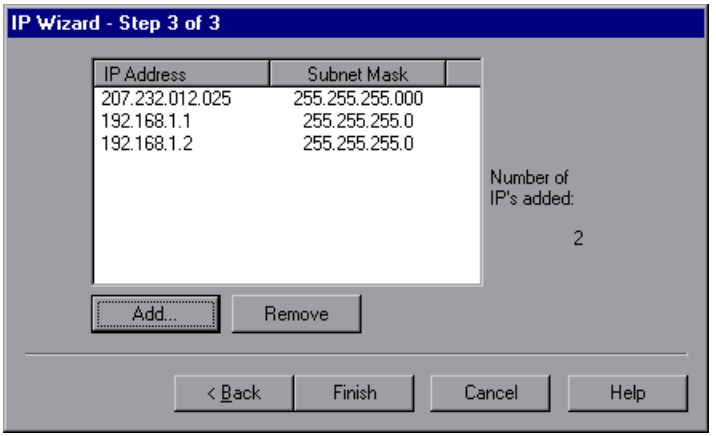
IP addresses include two components, a netid and hostid. The submask determines where the netid portion of the address stops and where the hostid begins.

7 Select a class that represents the correct submask for the machine's IP addresses.

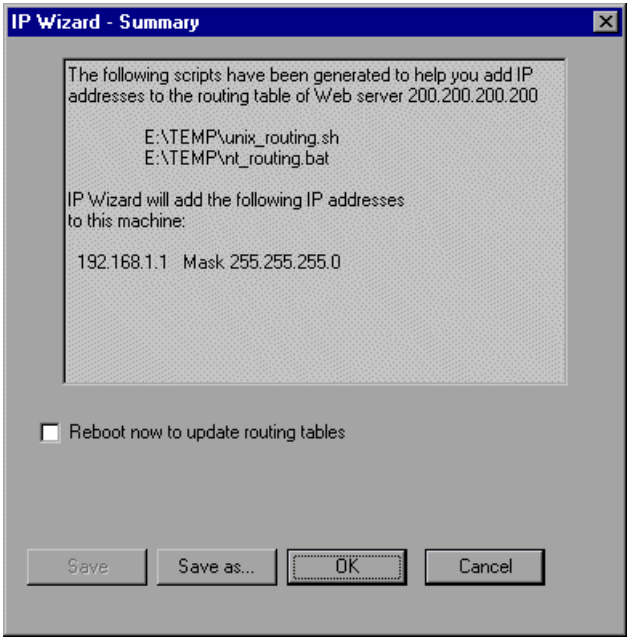
8 Specify the number of addresses to create. Select **Verify that new IP addresses are not already in use** to instruct the IP Wizard to check the new addresses. If some addresses are in use, the IP Wizard will only add the addresses not in use.

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** dialog box is dispalyed.



11 Note the address of the .bat file and check **Reboot now to update routing tables** to initialize the NT device drivers with the new addresses.

12 Click **OK**.

13 Edit the .bat file by inserting your IP address instead of CLIENT.IP.

14 Update the Web Server routing table.

Updating the Routing Table

Now that the client machine has new IP addresses, the server needs the addresses in its routing table, so 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.

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 Scenario Window, select **Scenario>Enable IP Spoofer**.

Index

D

E

F

G

H

Q

R

S

T

U

V

W

Host Resolution Functions Copyright Agreement

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.merc-int.com
Customer Support: web.merc-int.com