



Astra® LoadTest™

Analysis User's Guide
Version 4.5

Online Guide



 Find

Find
Again



Table of Contents

| | |
|---|----------|
| Chapter 1: Understanding Astra LoadTest Analysis | 6 |
| About the Analysis..... | 7 |
| Creating Analysis Sessions | 8 |
| Starting the Analysis..... | 8 |
| Collating Execution Results..... | 9 |
| Analysis Graphs | 10 |
| Applying a Filter to a Graph..... | 12 |
| Grouping and Sorting Results | 17 |
| Working with Analysis Graphs..... | 19 |
| Viewing the Legend..... | 28 |
| Viewing the Data as a Spreadsheet and as Raw Data..... | 31 |
| Configuring Display Options..... | 34 |
| Setting General and Database Options..... | 39 |

 Find

Find
Again



 Top of
Chapter

 Back

| | |
|--|-----------|
| Chapter 2: Viewing Transaction Graphs | 42 |
| About Transaction Graphs..... | 43 |
| Transaction Summary Graph | 44 |
| Transaction Performance Summary Graph..... | 45 |
| Transactions per Second Graph..... | 46 |
| Transaction Response Time - Average Graph | 47 |
| Transaction Response Time - Under Load Graph..... | 49 |
| Transaction Response Time - Distribution Graph | 50 |
| Transaction Response Time - Percentile Graph | 52 |
| Chapter 3: Vuser and Rendezvous Graphs..... | 54 |
| About Vuser and Rendezvous Graphs..... | 54 |
| Vuser Summary Graph..... | 55 |
| Running Vusers Graph..... | 56 |
| Rendezvous Graph..... | 57 |
| Chapter 4: Resource Graphs..... | 58 |
| About Resource Graphs..... | 58 |
| System Resource Graphs | 59 |
| Chapter 5: Web Resource Graphs | 61 |
| About Web Resource Graphs..... | 61 |
| Web Server Resource Graphs | 63 |
| Web Application Server Graphs | 68 |



Find
Again



Top of
Chapter

← Back

| | |
|--|------------|
| Chapter 6: Network Graphs | 87 |
| About Network Monitoring | 87 |
| Understanding Network Monitoring | 88 |
| Network Monitor Graph | 89 |
| Network Monitor Segment Graph | 91 |
| Timeout Count Graph | 92 |
| Verifying the Network as a Bottleneck..... | 94 |
| Chapter 7: Cross Result and Merged Graphs..... | 95 |
| About Cross Result and Merged Graphs..... | 95 |
| Cross Result Graphs | 96 |
| Generating Cross Result Graphs | 99 |
| Merging Graphs..... | 101 |
| Creating a Merged Graph..... | 104 |
| Chapter 8: Understanding Analysis Reports | 107 |
| About Analysis Reports | 107 |
| Viewing Summary Reports | 108 |
| Creating HTML Reports..... | 109 |

 Find

 Find
Again


 Top of
Chapter

 Back

| | |
|--|------------|
| Chapter 9: Managing Results Using TestDirector..... | 111 |
| About Managing Results Using TestDirector..... | 112 |
| Opening a Connection to a TestDirector Project..... | 112 |
| Creating a New Session Using TestDirector | 115 |
| Opening an Existing Session Using TestDirector | 118 |
| Saving Sessions to a TestDirector Project | 120 |
| | |
| Index | 121 |

 Find

Find Again

 Top of Chapter

 Back

Understanding Astra LoadTest Analysis

The Analysis provides graphs and reports to help you analyze the performance of your system. These graphs and reports summarize the scenario execution.

This chapter describes:

- **Creating Analysis Sessions**
- **Starting the Analysis**
- **Collating Execution Results**
- **Analysis Graphs**
- **Applying a Filter to a Graph**
- **Grouping and Sorting Results**
- **Working with Analysis Graphs**
- **Viewing the Legend**
- **Viewing the Data as a Spreadsheet and as Raw Data**
- **Configuring Display Options**
- **Setting General and Database Options**



Find
Again



← Back

About the Analysis

During scenario execution, Vusers generate result data as they perform their transactions. To monitor the scenario performance *during* test execution, use the Online Monitoring tools. To view a summary of the results *after* test execution, you can use one or more of the following tools:

- 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.) The information is also available in the Vuser log.
- 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 the *Astra LoadTest Controller User's Guide*.
- The **Analysis graphs** help you determine system performance and provide information about transactions and Vusers. You can compare multiple graphs by combining results from several scenarios or merging several graphs into one.
- The **Graph Data** and **Raw Data** views display the actual data used to generate the graph in a spreadsheet format. You can copy this data into external spreadsheet applications for further processing.
- The **Report** utilities enable you to view a Summary report and summary HTML report for each graph.



Find Again



Top of Chapter

← Back

Creating Analysis Sessions

When you run a scenario, data is stored in a result file with an *.lrr* extension. The Analysis is the utility that processes the gathered result information and generates graphs and reports.

When you work with the Analysis utility, you work within a *session*. An Analysis session contains at least one set of scenario results (*lrr* file). The Analysis stores the display information and layout settings for the active graphs into a file with an *.lra* extension.

Starting the Analysis

You can open the Analysis through the Astra LoadTest program group as an independent application, or directly from the Controller. To open the new Analysis utility as an independent application, choose **Tools > Analysis** from the Astra LoadTest Program Group.

To open the Analysis directly from the Controller, select **Results > Analyze Results**. This option is only available after running a scenario. The Analysis takes the latest result file from the current scenario, and opens a new session using these results. You can also instruct the Controller to automatically open the Analysis after it completes scenario execution by selecting Auto Load Analysis. To do this, select **Results > Auto Load Analysis**.



Find Again



← Back

When creating a new session, the Analysis prompts you for the result file (*.lrr* extension) that you wish to include in the session. To open an existing session, you specify an Analysis Session file (*.lra* extension).

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. You disable automatic collation by choosing **Results > Auto Collate Results** from the Controller window, to clear the check mark adjacent to the option. To manually collate results, choose **Results > Collate Results**. If your results have not been collated, the Analysis will automatically collate the results before generating the analysis data. For more information about collating results, see *Astra LoadTest Controller User's Guide*.

 Find Find Again Top of Chapter Back

Analysis Graphs

The Analysis graphs are divided into the following categories:

- Transactions
- Vusers
- Rendezvous
- System Resources
- Web Server Resources
- Web Application Server Resources
- Network Monitor

Transaction graphs and reports provide information about transaction performance and response time. For more information, see Chapter 2, [Viewing Transaction Graphs](#).

Vuser graphs display information about Vuser states and other Vuser statistics. For more information, see Chapter 3, [Vuser and Rendezvous Graphs](#).

Rendezvous graphs show the behavior of Vusers that participated in a rendezvous during the scenario. For more information, see Chapter 3, [Vuser and Rendezvous Graphs](#).



System Resource graphs show statistics relating to the system resources that were monitored during the scenario using the Online Monitor. This category also includes graphs for SNMP monitoring. For more information, see Chapter 4, [Resource Graphs](#).

Web Server Resource graphs provide information about the throughput and hits per second for Web Vusers. In addition, these graphs provide resource usage details for Netscape, IIS, and Apache Web servers. For more information see Chapter 5, [Web Resource Graphs](#).

Web Application Server Resource graphs provide information about the resource usage for various Web application servers such as BroadVision, ColdFusion, SilverStream, ASP, WebLogic, and WebSphere. For more information see Chapter 5, [Web Resource Graphs](#).

Network Monitor graphs provide information about the network delays. For more information, see Chapter 6, [Network Graphs](#).

Cross Result graphs let you compare the results from several scenario runs. For example, this feature can be useful for benchmarking hardware to determine how a superior system will improve performance. For more information, see Chapter 7, [Cross Result and Merged Graphs](#).



Find
Again



← Back

Applying a Filter to a Graph

When viewing graphs in an Analysis session, you can specify to display only the desired information. By default, ten transactions are displayed for the entire length of the scenario. You can filter a graph to show fewer transactions for a specific segment of the scenario. For example you can display four transactions beginning from five minutes into the scenario and ending three minutes before the end of the scenario.

The filter conditions differ for each type of graph. The filter conditions also depend on your scenario. For example, if you only had one group or one host machine in your scenario, the Group Name and Host Name filter conditions do not apply. The following table lists the available filter conditions for each graph type.

| Graph | Filter Conditions |
|------------------|---|
| Transactions | Transaction Name, Transaction Response Time, Scenario Elapsed Time, Host Name, Transaction End Status, Group Name, Vuser ID |
| Vusers | Vuser End Status, Scenario Elapsed Time, Host Name, Group Name, Vuser ID, Vuser Status |
| Rendezvous | Number of Released Vusers, Scenario Elapsed Time, Host Name, Group Name, Vuser ID |
| System Resources | Resource Name, Resource Value, Scenario Elapsed Time, Host Name, Group Name, Vuser ID |



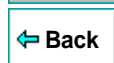
| Graph | Filter Conditions |
|----------------------------------|---|
| Web Server Resources | Resource Name, Resource Value, Scenario Elapsed Time, Host Name, Group Name, Vuser ID |
| Web Application Server Resources | Application Name, Application Value, Scenario Elapsed Time, Host Name, Group Name, Vuser ID |
| Network Monitor | Network Path Name, Network Path Delay, Scenario Elapsed Time, Host Name, Group Name, Vuser ID |

You can also activate global filter conditions that will apply to all graphs in the session. The available global filter conditions are a combination of the filter conditions that apply to the individual graphs.

Note that you can also filter merged graphs. The filter conditions for each graph are displayed on separate tabs.

To set a filter condition for an individual graph:

- 1 Select the graph you want to filter. Click on a graph tab or click on the graph name in the tree view.



- 2 Choose **View > Set Filter/Group By**. The Graph Settings dialog box opens.

Graph Settings

Graph: Transaction Summary

Filter condition:

| | Criteria | Values |
|-----------------------------|----------|-----------------|
| Transaction Name | = | TEnd1 or TInit1 |
| Transaction Response Time | | |
| Scenario Elapsed Time | | |
| Host Name | | |
| Result Name | <> | result7.ltr |
| Transaction End Status Name | | |
| Group Name | | G1 or G2 |
| VuserID | | |

Group By:

Available groups:

- Host Name
- VuserID

Selected groups:

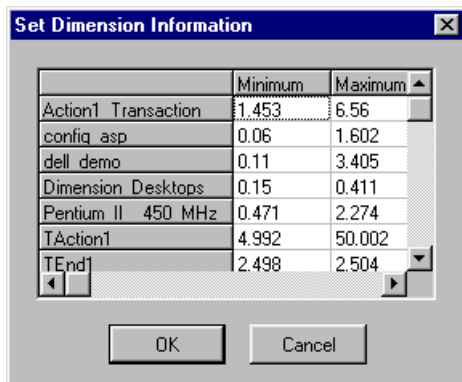
- Result Name
- Group Name

Buttons: OK, Cancel, Help, Set Defaults

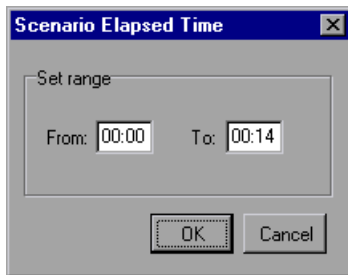
- 3 Click in the **Criteria** box for the condition you want to set. Select either "=" or "<>" (does not equal) from the drop down list.
- 4 Click in the **Values** box of the filter condition you want to set. Select a value from the drop down box. For several filter conditions, a new dialog box opens.



- 5 For Transaction Response Time, the Set Dimension Information dialog box opens. You specify a start and end time (in minutes:seconds format) for each transaction. The time is relative to the beginning of the scenario.



- 6 For Scenario Elapsed Time, the Scenario Elapsed Time dialog box opens. Specify the start and end time for the graph in minutes:seconds format. The time is relative to the start of the scenario.



- 7 For Rendezvous graphs, while setting the **Number of Released Vusers** condition, the Set Dimension Information dialog box opens. Specify a minimum and maximum number of released Vusers.
- 8 For all graphs that measure resources (Web Server, Database Server, etc.), when you set the **Resource Value** condition, the Set Dimension Information dialog box opens displaying a full range of values for each resource. Specify a minimum and maximum value for the resource.
- 9 Click **OK** to accept the settings and close the Graph Settings dialog box.



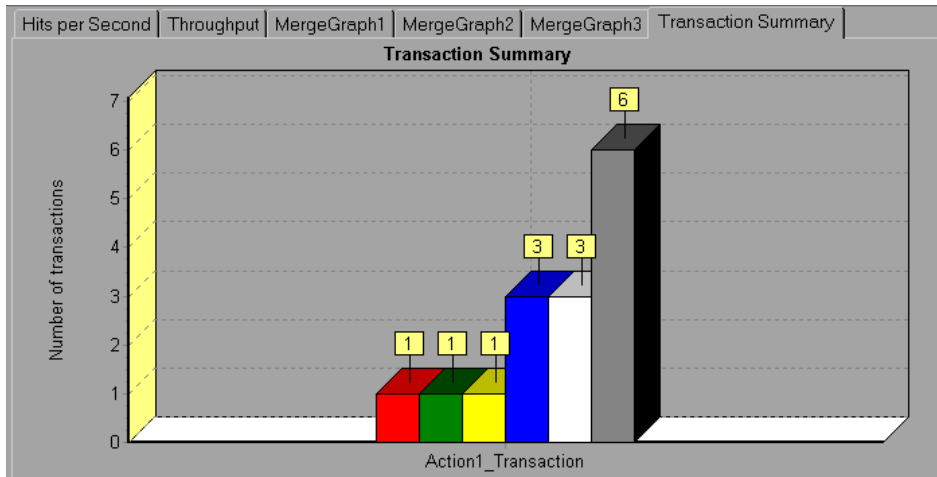
To apply a global filter condition for all graphs displayed in the session, choose **File > Set Global Filter** or click the **Set Global Filter** button, and set the desired filters.



Grouping and Sorting Results

When viewing a graph, you can group the data in several ways. All graphs can be grouped by the Vuser ID. Transactions graphs can be grouped by the Transaction End Status. The Vuser graphs can be grouped by the Scenario Elapsed Time, Vuser End Status, Vuser Status and VuserID.

You can also sort by more than one group—for example by Vuser ID and then Vuser status. The results are displayed in the order in which the groups are listed. You can change the order in which things are grouped by rearranging the list. The following graph shows the Transaction Summary grouped according to the Vuser.



Find

Find Again

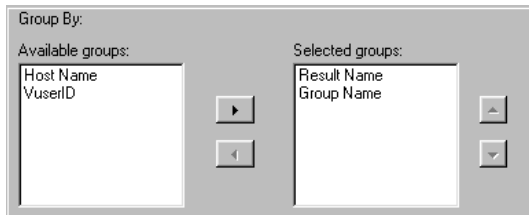


Top of Chapter

Back

To sort the graph data according to groups:

- 1 Select the graph you want to sort. Click on a graph tab or click on the graph name in the tree view.
- 2 Choose **View > Set Filter/Group By**. The Graph Settings dialog opens.
- 3 In the **Available Groups** box, select the group by which you want to sort the results.



- 4 Click the right-facing arrow to move your selection to the **Selected Groups** box.
- 5 To change the order in which the results are grouped, select the group you want to move and click the up or down arrow until the groups are in the desired order.
- 6 To remove an existing grouping, select an item in the **Selected Groups** box and click the left-facing arrow to move it to the **Available Groups** box.
- 7 Click **OK** to close the dialog box.



Working with Analysis Graphs

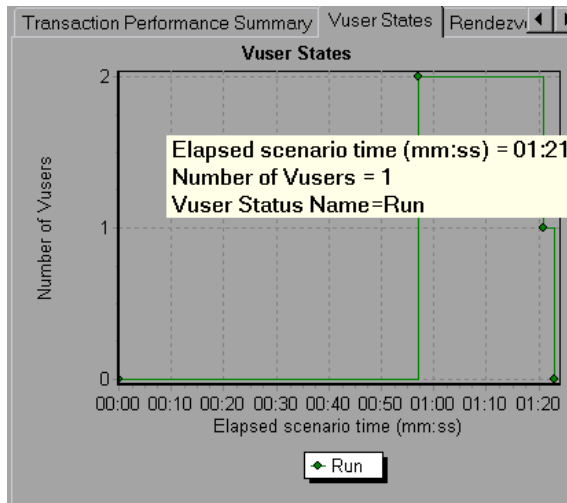
The Analysis contains several utilities that allow you to view the data in the graph in the most effective way:

- **Determining a Point's Coordinates**
- **Performing a Drill Down**
- **Enlarging a Section of a Graph**
- **Changing the Granularity of the X-axis**



Determining a Point's Coordinates

You can determine the coordinates and values at specific points in a graph. Place the cursor over the point you want to evaluate. The Analysis displays the axis values and other grouping information.



Find Again



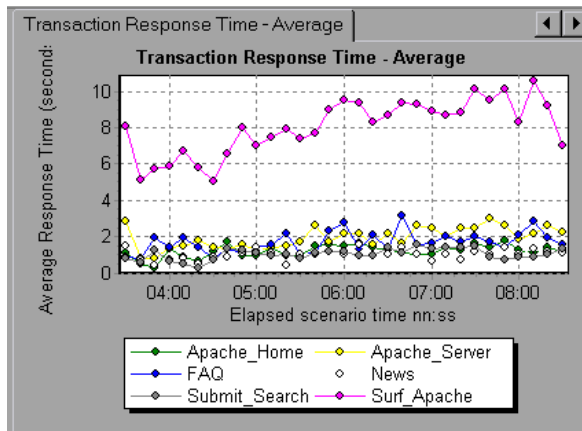
Top of Chapter

← Back

Performing a Drill Down

The Analysis provides another filtering mechanism called *drill down*. Drill down lets you focus on a specific measurement within your graph. You select one measurement within the graph and display it according to desired grouping. The available groupings are Vuser ID, Host, Group Name, Vuser Status, etc. depending on the graph. For example, the **Transaction Response Time - Average** graph shows one line per transaction. To determine the response time for each host, you drill down on one transaction and sort it according to host. The graph displays a separate line for the transaction response time on each host.

In the following example, the graph shows a line for each of the six transactions.



Find

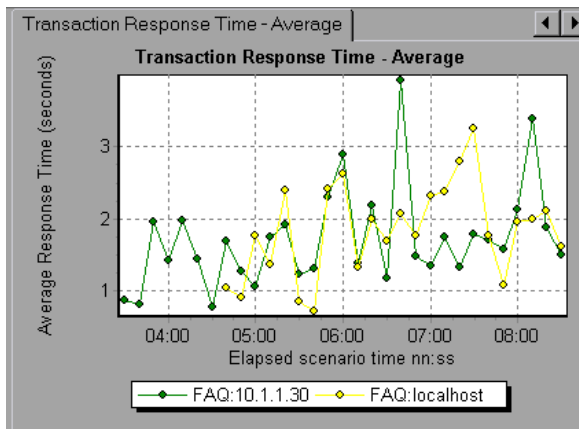
Find Again

◀ ▶

Top of Chapter

← Back

In a drill down for the FAQ transaction per host, the graph displays the response time only for the FAQ transaction, one line per host.



Find

Find Again

Navigation arrows

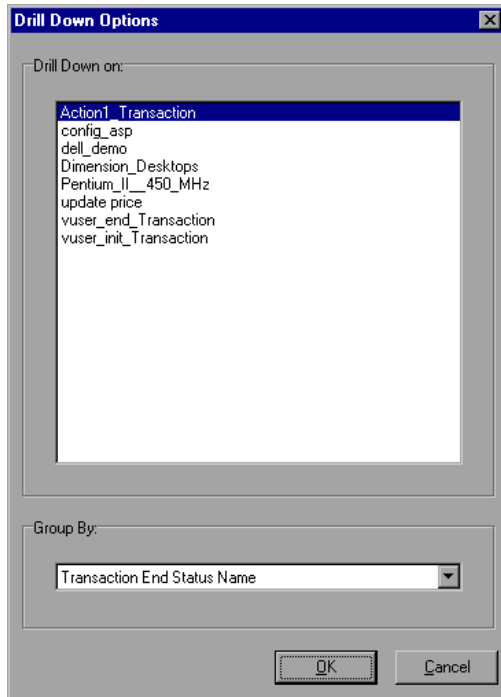
Top of Chapter

Back

You can see from the graph that the response time was longer on one host. If you sort it by Vuser ID, each Vuser is represented by a separate line.

To drill down in a graph:

- 1 Click on a line or bar within the graph. The Drill Down Options dialog box opens listing all of the measurements in the graph.



- 2 Select a measurement for drill down.
- 3 From the **Group By** box, select a group by which to sort.
- 4 Click OK. The Analysis drills down and displays the new graph.
- 5 To undo the last drill down settings, choose **Undo Set Filter/Group By** from the right-click menu.
- 6 To perform additional drill downs, repeat steps 1 to 4.
- 7 To clear all filter and drill down settings, choose **View > Clear Filter/Group By**.

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 zoom-in on a specific period of 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 fifth minutes.

To zoom in on a section of the graph:

- 1 Click inside a graph.
- 2 Move the mouse pointer to the beginning of the section you want to enlarge, but not over a line of the graph.
- 3 Hold down the left mouse button and draw a box around the section to enlarge.
- 4 Release the left mouse button. The section is enlarged.
- 5 To restore the original view, choose **Undo Zoom In** from the right-click menu.



Find
Again



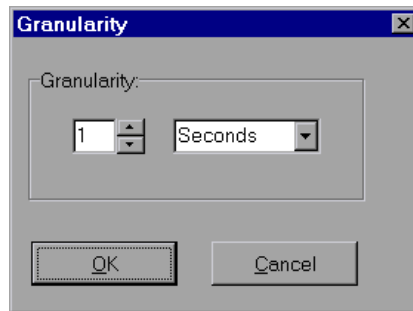
← Back

Changing the Granularity of the X-axis

You can make the graphs easier to read and analyze by changing the granularity (scale) of the x-axis. The maximum granularity is half of the graph's time range. To ensure readability and clarity, the Analysis automatically adjusts the minimum granularity of graphs with ranges of 500 seconds or more.

To change the granularity of a graph:

- 1 Click inside a graph.
- 2 Select **View > Set Granularity**, or click the **Set Granularity** button. The Granularity dialog box opens.



- 3 Enter a new granularity value in milliseconds, minutes, seconds, or hours.
- 4 Click **OK**.

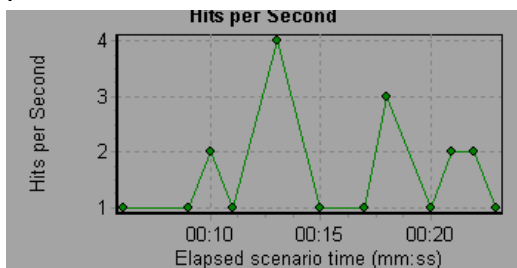


Find
Again

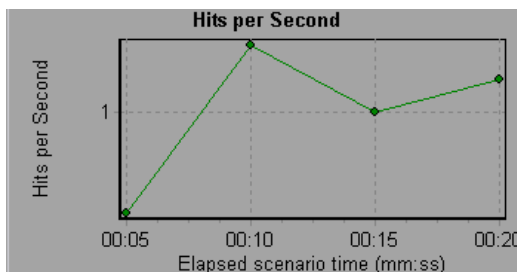


Top of
Chapter

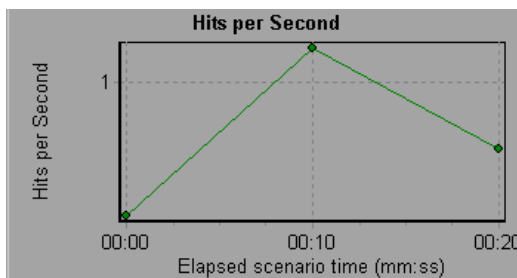
Back



Granularity=1



Granularity=5



Granularity=10

 Find

 Find Again

 Top of Chapter

 Back

In the above example, the Hits per Second 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 5, the y-axis shows the average response time for every five-second period of the scenario

In the above graphs, the same scenario results are displayed in a granularity of 1, 5, and 10. The lower the granularity, the more detailed the results. For example, using a low granularity as in the upper graph, you see the intervals when no transactions were performed. It is useful to use a higher granularity to study the overall Vuser behavior throughout the scenario.

By viewing the same graph with a higher granularity, you can easily see that the overall transaction response time averaged 1 hit per second.



Viewing the Legend

The Legend displays the color, scale, minimum, maximum, average, median, and standard deviation of each measurement appearing in the graph.

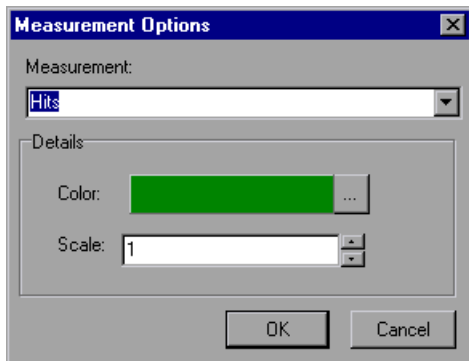
| Legend | Graph Details | User Notes | Graph Data | Raw Data | | | |
|-------------------------------------|---------------|---------------------------|------------|----------|---------|--------|----------------|
| Color | Scale | Measurement | Minimum | Maximum | Average | Median | Std. Deviation |
| <input checked="" type="checkbox"/> | 1 | air_head2_gif_Action1_159 | 0.176 | 0.657 | 0.247 | 0.19 | 0.155 |
| <input checked="" type="checkbox"/> | 1 | beige_gif_Action1_183 | 0.167 | 0.434 | 0.24 | 0.194 | 0.092 |
| <input checked="" type="checkbox"/> | 1 | book_gif_Action1_167 | 0.198 | 0.812 | 0.291 | 0.2 | 0.2 |
| <input checked="" type="checkbox"/> | 1 | bullet1_gif_Action1_234 | 0.11 | 0.245 | 0.164 | 0.17 | 0.04 |
| <input checked="" type="checkbox"/> | 1 | butt_ast_gif_Action1_207 | 0.13 | 0.238 | 0.177 | 0.176 | 0.027 |
| <input checked="" type="checkbox"/> | 1 | Click_here_Action1_107 | 0.187 | 0.827 | 0.288 | 0.201 | 0.22 |

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

- **Show/Hide:** Displays or hides a measurement in the graph.
- **Show only selected:** Displays the highlighted measurement only.
- **Show all:** Displays all the available measurements in the graph.



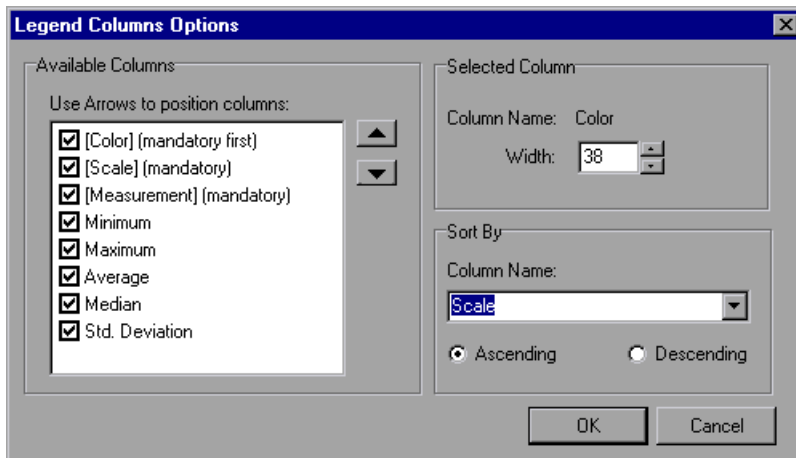
- **Configure measurements:** Opens the Measurement Options dialog box in which you can set the color and scale of the measurement you selected.



- **Animate selected line:** Displays the selected measurement as a flashing line.
- **Sort by [SELECTED] column:** Sorts the measurements according to the selected column, in ascending or descending order.



- **Configure columns:** Opens the Legend Columns Options dialog box in which you can choose the columns you want to view, the width of each column, and the way you want to sort the columns.



 Find

Find Again



 Top of Chapter

 Back

Viewing the Data as a Spreadsheet and as Raw Data

The Analysis 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 the Analysis in spreadsheet format. The **Graph Data** tab below the graph, displays the data in a grid format.

| Transaction Name | Minimum | Average | Maximum |
|---------------------------|---------|---------|---------|
| air_head2_gif_Action1_159 | 0.12 | 0.304 | 3.105 |
| beige_gif_Action1_183 | 0.111 | 0.27 | 1.462 |
| book_gif_Action1_167 | 0.19 | 0.373 | 3.314 |
| bullet1_gif_Action1_234 | 0.11 | 0.184 | 0.701 |
| butt_ast_gif_Action1_207 | 0.13 | 0.184 | 0.311 |
| Click_here_Action1_107 | 0.16 | 0.33 | 2.124 |

The first column displays the values of the x-axis. The following columns show the y-axis values for each transaction.

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.



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

- **Copy:** You can copy one or more columns to the clipboard in order to paste it/them into an external spreadsheet program.
- **Save As:** You can save the spreadsheet data to an Excel file. Once you have the data in Excel, you can generate your own customized graphs.

Raw Data View

You can view the actual raw data collected during test execution for the active graph. The Raw Data view is not available for all graphs.

Viewing the raw data can be especially useful for:

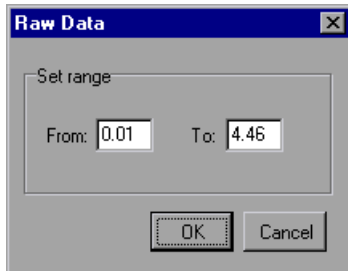
- determining specific details about a peak—for example which Vuser was running the transaction that caused the peak value(s).
- performing a complete export of unprocessed data for your own spreadsheet application.

 Find Find Again Top of Chapter Back

To display a graph's Raw Data view:



- 1 Choose **View > View Raw Data** or click the **Raw Data** button. The Raw Data dialog box opens.



- 2 Specify a time range—the entire graph (default) or a specific range of time—and click **OK**.
- 3 Click the **Raw Data** tab below the graph. The Analysis displays the raw data in a grid directly below the active graph.

| Transaction Name | Transaction Response Time | Scenario Elapsed Time | Transaction End Status Name | UserID |
|---------------------|---------------------------|-----------------------|-----------------------------|--------|
| Dimension_Desktops | 0.25 | 18 | Pass | Vuser1 |
| Pentium_II_450_MHz | 2.274 | 15 | Pass | Vuser1 |
| Action1_Transaction | 6.5 | 13 | Pass | Vuser1 |
| dell_demo | 1.502 | 18 | Pass | Vuser1 |
| Action1_Transaction | 4.106 | 17 | Pass | Vuser1 |
| dell_demo | 0.17 | 13 | Pass | Vuser1 |

- 4 To show a different range, repeat the above procedure.



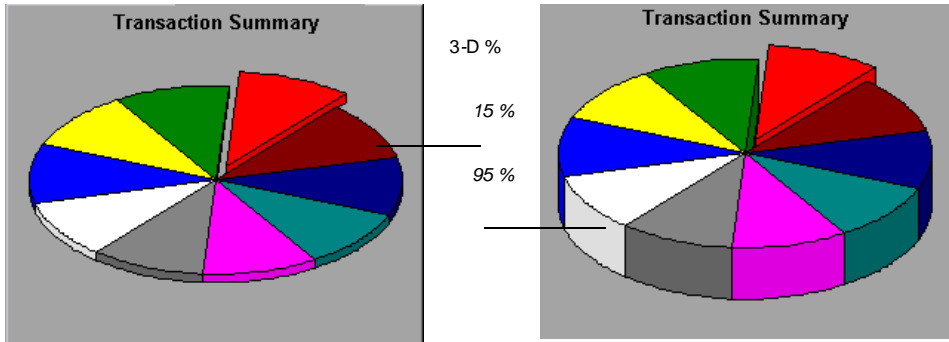
Configuring Display Options

You can configure both standard and advanced display options for each graph. The standard options let you select the type of graph and the time setting. The advanced options allow you to modify the scale and format of each graph.

Standard Display Options

The standard display options let you choose the type of graph to display: line, point, bar, or pie graph. Not all options are available for all graphs.

You can also indicate whether the graph should be displayed with a 3 dimensional look. This applies primarily to bar and pie graphs. You can specify the percent for the 3 dimensional graphs. This percentage indicates the thickness of the bar or pie chart.



 Find

Find Again

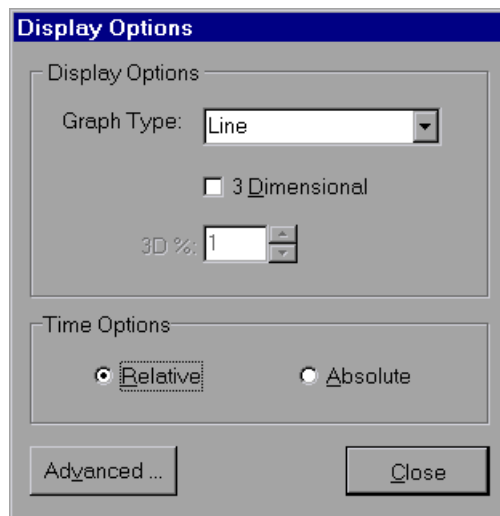


 Top of Chapter

 Back

The standard display options also let you indicate how to plot the results that are time-based: relative to the beginning of the scenario (default), or absolute time, based on the system clock of the machine.

To open the Display Options dialog box, choose **View > Display Options** or click the **Display Options** button.



Choose a graph type, specify a 3 dimensional percent, and/or choose a time option. Click **Close** to accept the settings and close the dialog box.



Advanced Display Options

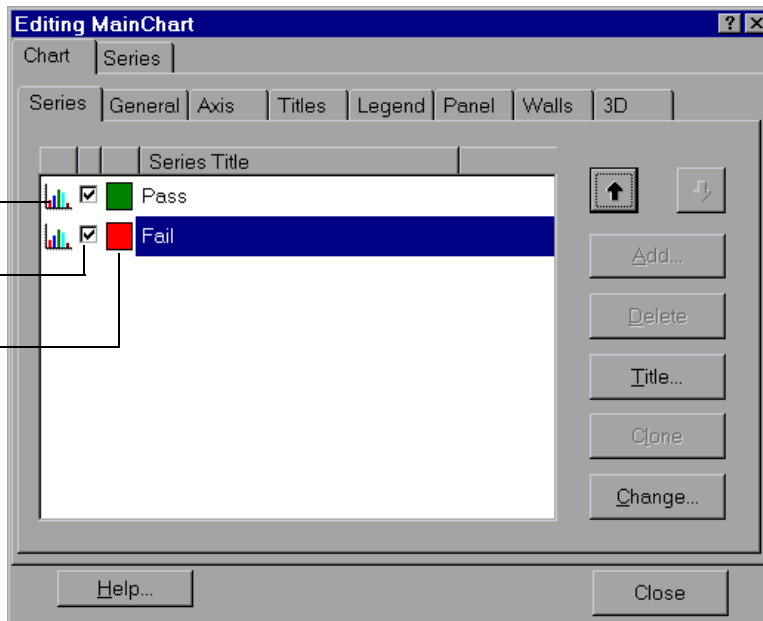
The Advanced options let you configure the look and feel of your graph as well as its title and the format of the data. To set the Advanced options, you must first open the Display Options dialog. Choose **View > Display Options** or click the **Display Options** button. To access the Advanced options, click **Advanced**. The Editing MainChart dialog box opens.



style

show/hide

color



You can customize the graph layout by setting the *Chart* and *Series* preferences. Click the appropriate tab and sub-tab to configure your graph.

Chart Settings

The Chart settings control the look and feel of the entire graph—not the individual points. You set Chart preferences using the following tabs: Series, General, Axis, Titles, Legend, Panel, Walls, and 3D.

Series: displays the graph style, (bar, line, etc.) the hide/show settings, line and fill color, and the title of the series.

General: contains options for print preview, export, margins, scrolling, and magnification.

Axis: indicates which axes to show as well as their scales, titles, ticks, and position.

Titles: allows you to set the title of the graph, its font, background color, border, and alignment.

Legend: includes all legend-related settings, such as position, fonts, and divider lines.

Panel: shows the background panel layout of the graph. You can modify its color, set a gradient option, or specify a background image.



Find
Again



← Back

Walls: lets you set colors for the walls of 3-dimensional graphs.

3D: contains the 3-dimensional settings, offset, magnification, and rotation angle for the active graph.

Series Settings

The Series settings control the appearance of the individual points plotted in the graph. You set Series preferences using the following tabs: Format, Point, General, and Marks.

Format: allows you to set the border color, line color, pattern and invert property for the lines or bars in your graph.

Point: displays the point properties. Points appear at various points within your line graph. You can set the size, color and shape of these points.

General: contains the type of cursor, the format of the axis values, and show/hide settings for the horizontal and vertical axis.

Marks: allows you to display the value for each point in the graph and configure the format of those marks.



Find
Again



Top of
Chapter

 Back

Setting General and Database Options

You use the *General* and *Database* Options to regulate the size of the database containing your test results and optimize the processing time of your Analysis session.

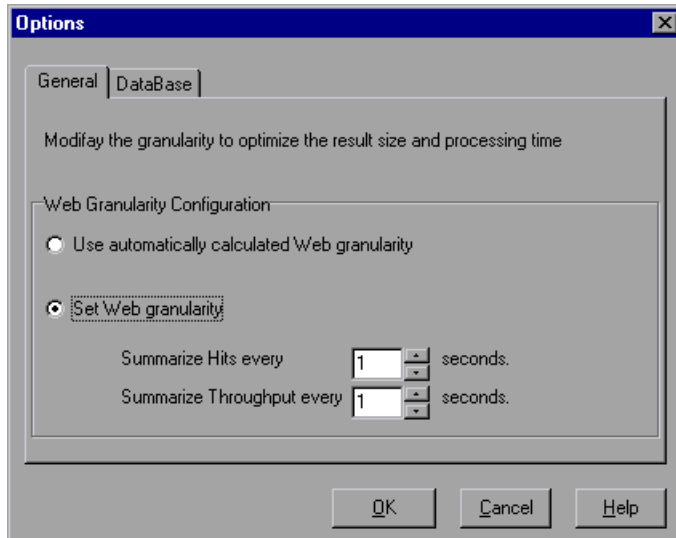
Setting General Options

User data from long scenario runs (lasting two hours or more) create large databases. To decrease the size of the database and thereby decrease processing time, you can increase the granularity for the two primary Web-related measurements that account for over half of the data in the database—Hits per Second and Throughput.



To configure Web granularity:

- 1 Select **Tools > Options**. The Options dialog box opens, displaying the General tab.



- 2 To allow the Analysis to determine the best granularity for displaying the results, select **Use automatically calculated Web granularity** (the default option).



- 3 To specify a custom granularity for the Hits per Second and Throughput measurements, select **Set Web granularity** and select the number of seconds you want the Analysis to wait before it summarizes the Hits and Throughput.

By default, the Analysis summarizes the Hits per Second and Throughput measurements every 5 seconds. To reduce the size of the database, increase the granularity. To focus on more detailed results, decrease the granularity.

Setting Database Options

When you configure and set up your Analysis session, the database containing the results may become fragmented. As a result, it will use excessive disk space. To repair and compress your results and optimize the database, select **Tools > Options** and click the **Database** tab. Click the **Compact Database Now** button.

Note: The long scenarios (duration of two hours or more) will require more time for compacting.



Viewing Transaction Graphs

After running a scenario, you can analyze the transactions that were executed during the test using one or more of the following graphs:

- **Transaction Summary Graph**
- **Transaction Performance Summary Graph**
- **Transactions per Second Graph**
- **Transaction Response Time - Average Graph**
- **Transaction Response Time - Under Load Graph**
- **Transaction Response Time - Distribution Graph**
- **Transaction Response Time - Percentile Graph**



About Transaction Graphs

During scenario execution, Vusers generate data as they perform transactions. The Analysis enables you to generate graphs that show the transaction performance and status throughout script execution.

You can use additional Analysis tools such as merging and crossing results to understand your transaction performance graphs. You can also sort the graph information by transactions. For more information about working with the Analysis, see Chapter 1, [Understanding Astra LoadTest Analysis](#).



Find
Again



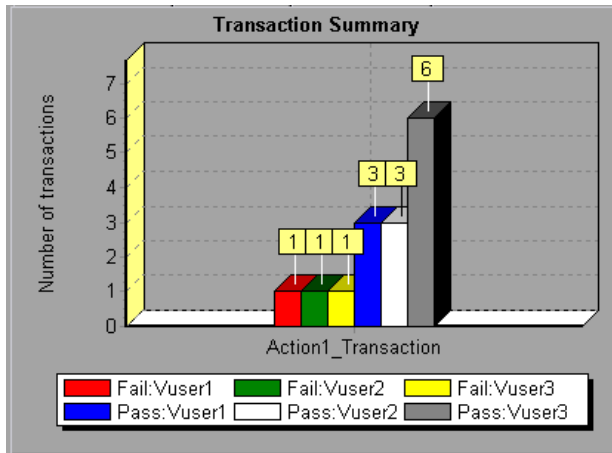

Top of
Chapter

 Back

Transaction Summary Graph

The Transaction Summary graph summarizes the number of transactions in the scenario that failed, passed, aborted and ended in error.

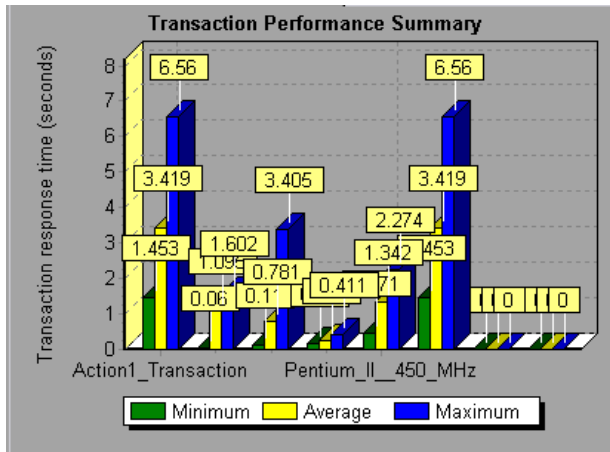
The x-axis specifies the name of the transaction. The y-axis shows the total number of transactions performed during the scenario run.

[Find](#)[Find Again](#)[Top of Chapter](#)[Back](#)

Transaction Performance Summary Graph

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.



Find

Find Again



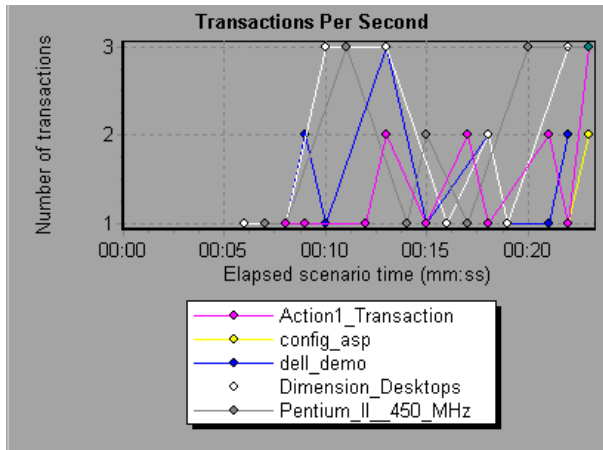
Top of Chapter

Back

Transactions per Second Graph

The Transactions per Second graph displays the number of 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 Response Time - Average 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) from the beginning of the scenario run. The y-axis represents the number of transactions successfully performed during the scenario run.



Find

Find Again



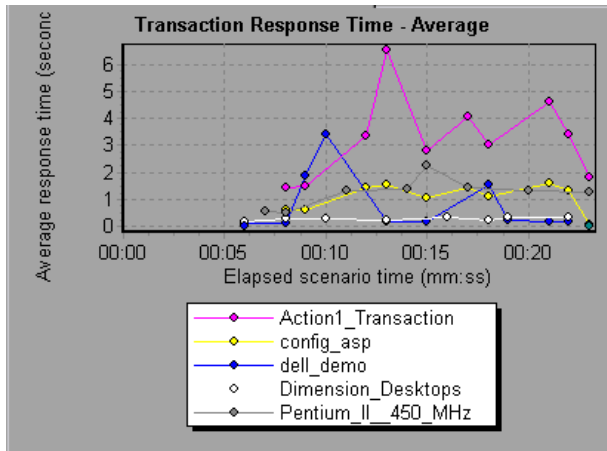
Top of Chapter

Back

Transaction Response Time - Average Graph

The Transaction Response Time – Average 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) from the beginning of the scenario run. The y-axis represents the average time (in seconds) taken to perform each transaction.



This graph is displayed differently for each granularity. 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.

Find

Find Again



Top of Chapter

Back

For example, using a low granularity, you may see intervals when no transactions were performed. However, by viewing the same graph with a higher granularity, you will see the graph for the overall transaction response time. For more information on setting the granularity, see Chapter 1, [Understanding Astra LoadTest Analysis](#).

You can compare the Transaction Response Time – Average graph to the Vuser States graph to see how the number of running Vusers affects the transaction performance time.

For example, if the Transaction Response Time – Average graph shows that performance time gradually improved, you can compare it to the Vuser States graph to see whether the performance time improved due to a decrease in the Vuser load.



Find Again



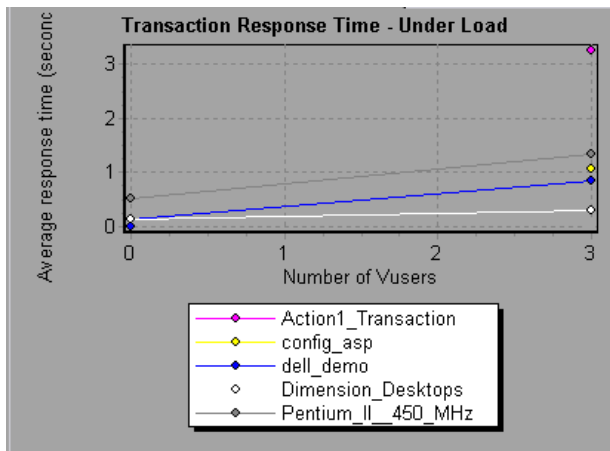
Top of Chapter

← Back

Transaction Response Time - Under Load Graph

The Transaction Response Time - 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 the average transaction time in seconds.

[Find](#)[Find Again](#)[Top of Chapter](#)[Back](#)

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 zeros.

For example, if you have a scenario with a load of between 50 and 70 Vusers and an average performance of 10 seconds, yet the graph shows zero, then the Vusers did not stabilize for 5 seconds.

Transaction Response Time - Distribution Graph

The Transaction Response Time - Distribution graph displays the distribution of the time taken to perform a transaction. If you compare it to the Transaction Performance Summary graph, you can see how the average performance was calculated.

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.



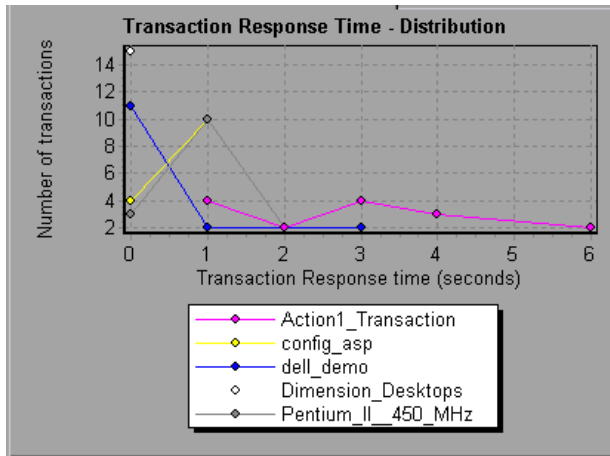
Find
Again



Top of
Chapter

← Back

In the following graph, 10 Vusers executed the *config_asp* transaction in the first second of the scenario.



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.

Transaction Response Time - Percentile Graph

The Transaction Response Time - 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.



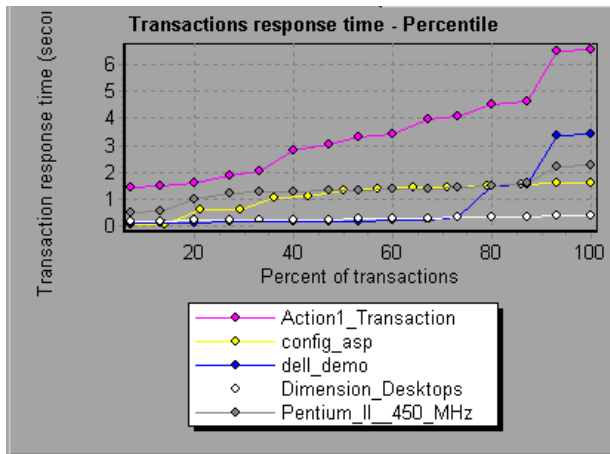
Find Again



Top of Chapter

← Back

In the following graph, approximately 90 per cent of the *Action1_Transaction* transactions had a response time of 5 seconds or less.



Find

Find Again



Top of Chapter

Back

It is recommended to compare the Percentile graph to a graph indicating average response time such as the Transaction Response Time - Average graph. A high response time for several transactions may raise the overall average. However, if the transactions with a high response time occurred less than five percent of the times, that factor may be insignificant.

Vuser and Rendezvous Graphs

After running a scenario, you can check the behavior of the Vusers that participated in the scenario using the following Vuser and Rendezvous graphs:

- **Vuser Summary Graph**
- **Running Vusers Graph**
- **Rendezvous Graph**

About Vuser and Rendezvous Graphs

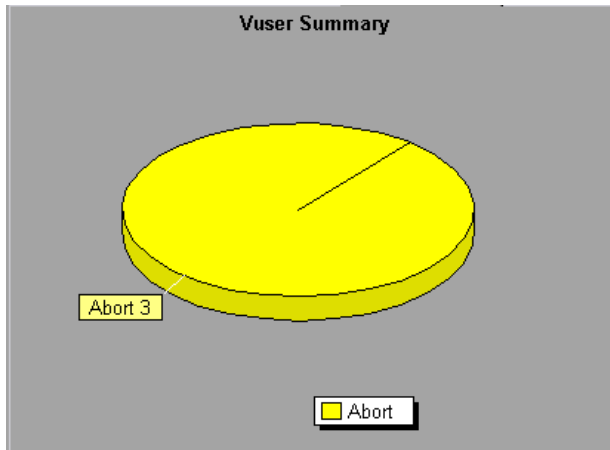
During scenario execution, Vusers generate data as they perform transactions. The Vuser and Rendezvous graphs let you determine the overall behavior of Vusers during the scenario. They display the Vuser states, the number of Vusers that completed the script, and rendezvous statistics. Use these graph in conjunction with Transaction graphs to determine the effect of the number of Vusers on transaction response time.



Vuser Summary Graph

The Vuser Summary graph displays a summary of Vuser performance. It lets you view the number of Vusers that successfully completed the scenario run relative to those that did not.

This graph may be viewed as a pie, line, point, or bar graph.



Find

Find Again

◀ ▶

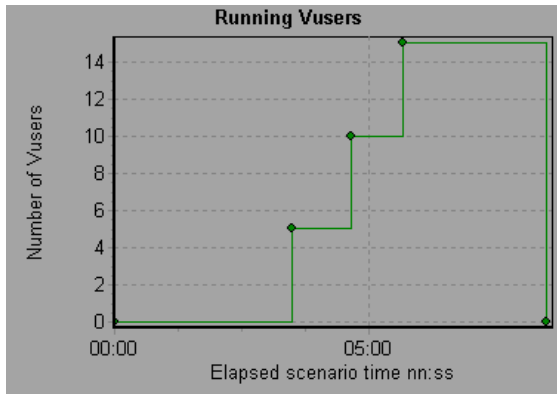
Top of Chapter

← Back

Running Vusers Graph

The Running Vuser States graph displays the number of Vusers that executed Vuser scripts and their status during each second of the test. This graph is useful for determining the Vuser load on your server at any given moment. By default, this graph only shows the Vusers with a *Run* status. To view another Vuser status, set the filter conditions to the desired status. For more information, see Chapter 1, [Understanding Astra LoadTest Analysis](#).

The x-axis represents the elapsed time (in seconds) from the beginning of the scenario run. The y-axis represents the number of Vusers in the scenario.



Find Again



Top of Chapter

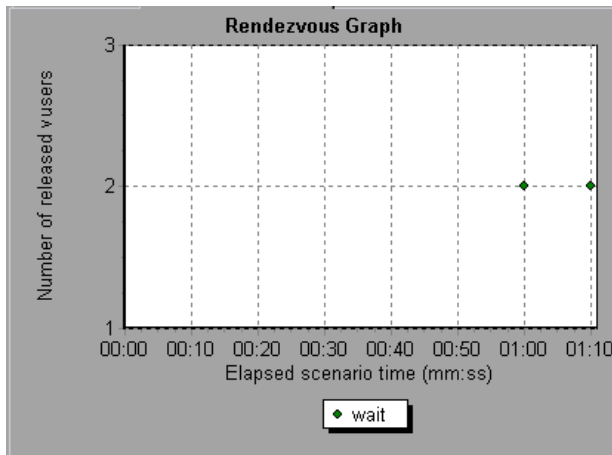
Back

Rendezvous Graph

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 Response Time - Average 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 beginning of the scenario. 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.



Find Again



Top of Chapter

← Back

Resource Graphs

After running a scenario, you can check the various resources that were monitored during the scenario using the:

- **System Resource Graphs**

About Resource Graphs

During scenario execution, Astra LoadTest collects data from the Vusers and online monitors. You configure the online monitors before running the scenario by indicating which resources to measure. You can view a summary of the statistics collected for each resource after the scenario using the Analysis tool. Each resource is represented by a separate line in the appropriate graph. For information about setting the online monitors, see the *Astra LoadTest Controller User's Guide*.

The following Resource Graphs are available:

- System Resource Graphs
- Web Server Resource Graphs
- Web Application Server Graphs

This chapter describes System Resource graphs. For information on Web and Web Application graphs, see Chapter 5, **Web Resource Graphs**.



Find
Again



← Back

System Resource Graphs

System Resource graphs display the system resource usage measured by the online monitors during the scenario run. These graphs require that you specify the resources you want to measure *before* running the scenario. For more information, see the section on online monitors in the *Astra LoadTest Controller User's Guide*. System Resource graphs are available for:

- Server Resources
- SNMP

The **Server Resource** graph shows the NT resources measured during the scenario. The NT measurements correspond to the built-in counters available from the NT Performance Monitor.



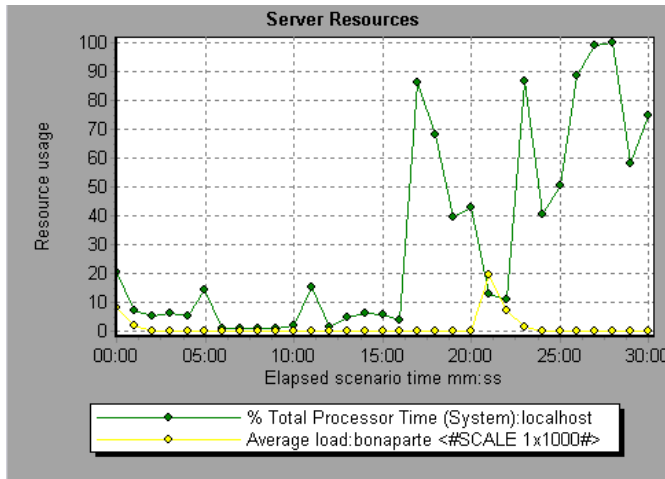
Find
Again




Top of
Chapter

 Back

This graph helps you determine the impact of Vuser load on the various system resources. The x-axis represents the elapsed time. The y-axis represents the resource usage.



Find

Find Again



Top of Chapter

Back

The **SNMP** graph shows statistics for machines using the Simple Network Management Protocol (SNMP).

Web Resource Graphs

After a scenario run, you can use graphs to analyze Web server and Web application server performance.

This chapter describes:

- **Web Server Resource Graphs**
- **Web Application Server Graphs**

About Web Resource Graphs

Web Vuser Graphs provide you with information about your Web server or Web application server. These graphs display information about hits, throughput, and other resources specific to your server.

There are two types of Web Vuser graphs: Web server graphs and Web application server graphs. Web server graphs show general performance information for the Web server. Web Application Server graphs provide specific information about application servers such as BroadVision and MS Active Server Pages.



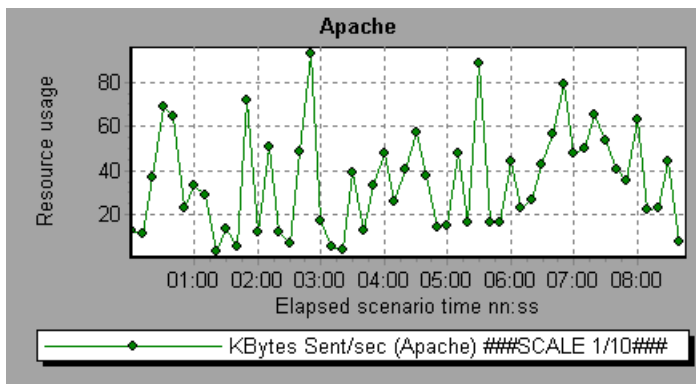
Find
Again



← Back

Note that in order to obtain data for Web Vuser graphs (except for **Hits per Second** and **Throughput**), you need to activate the online monitor for the server before running the scenario. When you set up the online monitor for a server, you indicate which statistics and measurements to monitor. For example, on the Apache server you could measure *Busy Servers*, *Idle Servers*, and *CPU usage*. For more information, see the *Astra LoadTest Controller User's Guide*.

In order to display all the measurements on a single graph, the Analysis may scale them. The legend below the graph indicates the scale factor for each resource. To obtain the true value, multiply the scale factor by the displayed value. For example, in the following graph the actual value of *KBytes Sent per second* in the second minute is 1; 10 multiplied by the scale factor of 1/10.



 Find

Find
Again



 Top of
Chapter

 Back

Web Server Resource Graphs

The following *Web Server Resource* graphs are available:

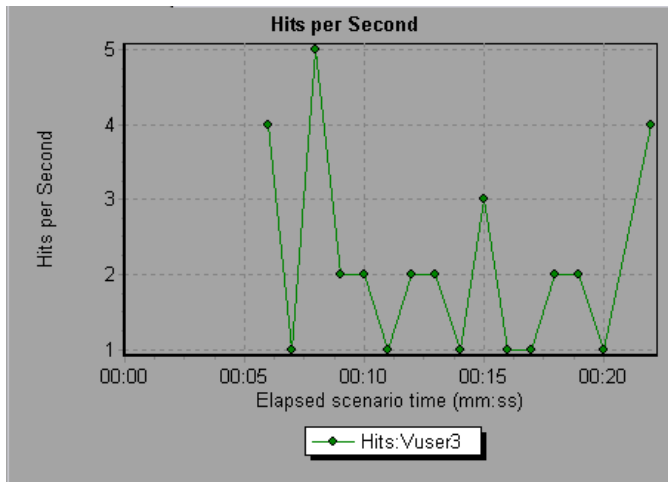
- Hits Per Second
- Throughput
- Apache
- Microsoft Information Internet Service (IIS)
- Netscape Enterprise Server

Note that the **Apache**, **Microsoft IIS**, and **Netscape Server** graphs require that you specify which resources you want to measure before running the scenario. For more information, see the *Astra LoadTest Controller User's Guide*.



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.

[Find](#)[Find Again](#)[Top of Chapter](#)[Back](#)

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 that the most hits per second took place during the eighth 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.



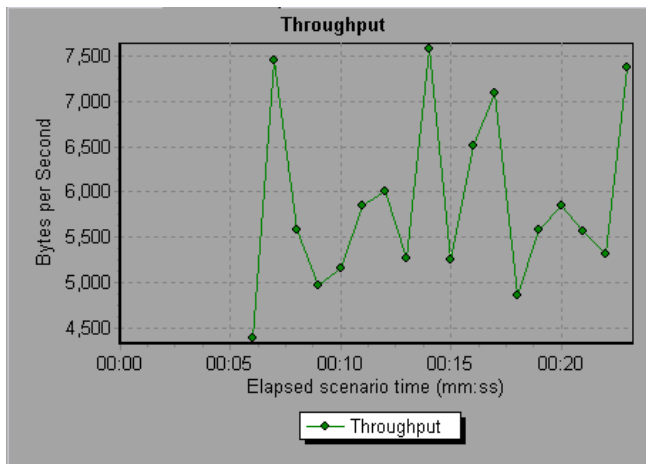
Find
Again



Top of
Chapter

← Back

The following graph shows the highest throughput to be 7,500 bytes during the sixth second of the scenario.



 Find

Find Again

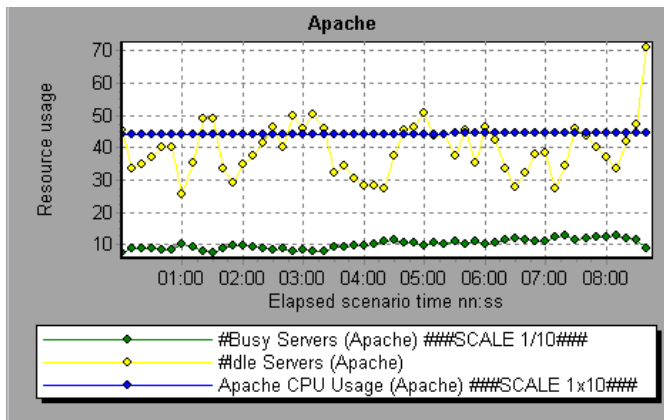


 Top of Chapter

 Back

Apache Server Graph

The Apache Server graph shows server statistics as a function of the elapsed scenario time.



In the above graph, the CPU usage remained steady throughout the scenario. At the end of the scenario, the number of idle servers increased. The number of busy servers remained steady at 1 throughout the scenario implying that the User only accessed one Apache server. Note that the scale factor for the *Busy Servers* resource is 1/10 and the scale factor for *CPU usage* is 10.

Find

Find Again



Top of Chapter

Back

Web Application Server Graphs

The following *Web Application Server Resource* graphs are available:

- BroadVision
- ColdFusion
- Microsoft Active Server Pages
- SilverStream
- WebLogic
- WebSphere

Note that the above graphs require that you specify the resources you want to measure before running the scenario.

When you open a Web Application Server Resource graph, you can filter it to only show the relevant application. When you need to analyze other applications, you can change the filter conditions and display the desired resources.

Each Web Application Server Resource graph measures different resources. These are detailed in the following sections.



Find
Again



Top of
Chapter

← Back

BroadVision

The BroadVision monitor supplies performance statistics for all the servers/services available within the application server.

The following table describes all the Servers/Services that can be monitored:

| Server | Multiple Instances | Description |
|--------------|--------------------|--|
| adm_srv | No | One-To-One user administration server. There must be one. |
| alert_srv | No | Alert server handles direct IDL function calls to the Alert system. |
| bvconf_srv | No | One-To-One configuration management server. There must be one. |
| cmsdb | Yes | Visitor management database server. |
| cntdb | Yes | Content database server. |
| deliv_smtp_d | Yes | Notification delivery server for e-mail type messages. Each instance of this server must have its own ID, numbered sequentially starting with "1". |
| deliv_comp_d | No | Notification delivery completion processor. |



Find Again



Top of Chapter

← Back

| | | |
|-------------|-----|--|
| extdbacc | Yes | External database accessor. You need at least one for each external data source. |
| genericdb | No | Generic database accessor handles content query requests from applications, when specifically called from the application. This is also used by the One-To-One Command Center. |
| hostmgr | Yes | Defines a host manager process for each machine that participates in One-To-One, but doesn't run any One-To-One servers. For example, you need a hostmgr on a machine that runs only servers. You don't need a separate hostmgr on a machine that already has one of the servers in this list. |
| g1_ofbe_srv | No | Order fulfillment back-end server. |
| g1_ofdb | Yes | Order fulfillment database server. |
| g1_om_srv | No | Order management server. |
| pmtassign_d | No | The payment archiving daemon routes payment records to the archives by periodically checking the invoices table, looking for records with completed payment transactions, and then moving those records into an archive table. |
| pmthdlr_d | Yes | For each payment processing method, you need one or more authorization daemons to periodically acquire the authorization when a request is made. |

 **Find**
 **Find Again**

 **Top of Chapter**
 **Back**

| | | |
|--------------|-----|--|
| pmtsettle_d | Yes | Payment settlement daemon periodically checks the database for orders of the associated payment processing method that need to be settled, and then authorizes the transactions. |
| sched_poll_d | No | Notification schedule poller scans the database tables to determine when a notification must be run. |
| sched_srv | Yes | Notification schedule server runs the scripts that generate the visitor notification messages. |



Find Again



Top of Chapter

← Back

Performance Counters

Performance counters for each server/service are divided into logical groups according to the service type.

The following section describes all the available counters under each group. Please note that for some services the number of counters for the same group can be different.

Counter groups:

- BV_DB_STAT
- BV_SRV_CTRL
- BV_SRV_STAT
- NS_STAT
- BV_CACHE_STAT
- JS_SCRIPT_CTRL
- JS_SCRIPT_STAT



Find
Again



Top of
Chapter



Back

BV_DB_STAT

The database accessor processes have additional statistics available from the BV_DB_STAT memory block. These statistics provide information about database accesses, including the count of selects, updates, inserts, deletes, and stored procedure executions.

- DELETE - Count of deletes executions
- INSERT - Count of inserts executions
- SELECT - Count of selects executions
- SPROC - Count of stored procedure executions.
- UPDATE - Count of updates executions

BV_SRV_CTRL

- SHUTDOWN



Find Again



Top of Chapter

← Back

NS_STAT

The NS process displays the namespace for the current One-To-One environment, and optionally can update objects in a name space.

- Bind
- List
- New
- Rebind
- Rsvl
- Unbind

BV_SRV_STAT

The display for Interaction Manager processes includes information about the current count of sessions, connections, idle sessions, threads in use, and count of CGI requests processed.

- **HOST** - Host machine running the process.
- **ID** - Instance of the process (of which multiple can be configured in the bv1to1.conf file), or engine ID of the Interaction Manager.
- **CGI** - Current count of CGI requests processed.
- **CONN** - Current count of connections.



- **CPU** - CPU percentage consumed by this process. If a process is using most of the CPU time, consider moving it to another host, or creating an additional process, possibly running on another machine. Both of these specifications are done in the `bv1to1.conf` file. The CPU % reported is against a single processor. If a server is taking up a whole CPU on a 4 processor machine, this statistic will report 100%, while the Windows NT Task Manager will report 25%. The value reported by this statistic is consistent with "% Processor Time" on the Windows NT Performance Monitor.
- **GROUP** - Process group (which is defined in the `bv1to1.conf` file), or Interaction Manager application name.
- **STIME** - Start time of server. The start times should be relatively close. Later times might be an indication that a server crashed and was automatically restarted.
- **IDL** - Total count of IDL requests received, not including those to the monitor.
- **IdIQ**
- **JOB**
- **LWP** - Number of light-weight processes (threads).
- **RSS** - Resident memory size of server process (in Kilobytes).
- **STIME** - System start time.
- **SESS** - Current count of sessions.
- **SYS** - Accumulated system mode CPU time (seconds).

 Find Find Again Top of Chapter Back

- **THR** - Current count of threads.
- **USR** - Accumulated user mode CPU time (seconds).
- **VSZ** - Virtual memory size of server process (in kilobytes). If a process is growing in size, it probably has a memory leak. If it is an Interaction Manager process, the culprit is most likely a component or dynamic object (though Interaction Manager servers do grow and shrink from garbage collection during normal use).

BV_CACHE_STAT

Monitors the request cache status.

The available counters for each request are:

- **CNT- Request_Name-HIT** - Count of requests found in the cache.
- **CNT- Request_Name-MAX** - Maximum size of the cache in bytes
- **CNT- Request_Name-SWAP** - Count of items that got swapped out of the cache.
- **CNT- Request_Name-MISS** - Count of requests that were not in the cache.
- **CNT- Request_Name-SIZE** - Count of items currently in the cache.

 Find Find Again Top of Chapter Back

Cache Metrics

Cache metrics are available for the following items:

- AD
- **ALERTSCHED** - Notification schedules are defined in the BV_ALERTSCHED and BV_MSGSCHED tables. They are defined by the One-To-One Command Center user or by an application.
- CATEGORY_CONTENT
- **DISCUSSION** - The One-To-One discussion groups provide moderated system of messages and threads of messages aligned to a particular topic. Use the Discussion group interfaces for creating, retrieving and deleting individual messages in a discussion group. To create, delete, or retrieve discussion groups, use the generic content management API. The BV_DiscussionDB object provides access to the threads and messages in the discussion group database.
- **EXT_FIN_PRODUCT**
- **EDITORIAL** - Using the Editorials content module, you can point cast and community cast personalized editorial content, and sell published text on your One-To-One site. You can solicit editorial content, such as investment reports and weekly columns, from outside authors and publishers, and create your own articles, reviews, reports, and other informative media. In addition to text, you can use images, sounds, music, and video presentations as editorial content.



Find
Again



← Back

- **INCENTIVE** - Contains sales incentives
- **MSGSCHEd** - Contains the specifications of visitor-message jobs. Notification schedules are defined in the BV_ALERTSCHEd and BV_MSGSCHEd tables. They are defined by the One-To-One Command Center user or by an application.
- **MSGSCRIPT** - Contains the descriptions of the JavaScripts that generate visitor messages and alert messages. Contains the descriptions of the JavaScripts that generate targeted messages and alert messages. Use the Command Center to add message script information to this table by selecting the Visitor Messages module in the Notifications group. For more information, see the Command Center User's Guide.
- **PRODUCT** - BV_PRODUCT contains information about the products that a visitor can purchase.
- **QUERY** - BV_QUERY contains queries.
- **SCRIPT** - BV_SCRIPT contains page scripts.
- **SECURITIES**



- **TEMPLATE** - The Templates content module enables you to store in the content database any BroadVision page templates used on your One-To-One site. Combining BroadVision page templates with BroadVision dynamic objects in the One-To-One Design Center application is one way for site developers to create One-To-One Web sites. If your developers use these page templates, you can use the Command Center to enter and manage them in your content database. If your site doesn't use BroadVision page template, you will not use this content module.

JS_SCRIPT_CTRL

- CACHE
- DUMP
- FLUSH
- METER
- TRACE

 FindFind
Again Top of
Chapter Back

JS_SCRIPT_STAT

- ALLOC
- ERROR
- FAIL
- JSPERR
- RELEASE
- STOP
- SUCC
- SYNTAX

ColdFusion

The following measurements are available for Allaire's ColdFusion server:

- **% Total Processor Time (System)** - The % Total Processor Time is the average percentage of time that all the processors on the system are busy executing non-idle threads. On a multi-processor system, if all processors are always busy this is 100%, if all processors are 50% busy this is 50% and if 1/4th of the processors are 100% busy this is 25%. It can be viewed as the fraction of the time spent doing useful work. Each processor is assigned an Idle thread in the Idle process which consumes those unproductive processor cycles not used by any other threads.



Find
Again



Top of
Chapter

← Back

- **Avg. Queue Time (msec)** - This is a running average of the amount of time, in milliseconds, requests spent waiting in the CF input queue before CF began to process that request.
- **Avg Req Time (msec)** - This is a running average of the total amount of time, in milliseconds, it took CF to process a request. In addition to general page processing time, this value includes both queue time and database processing time.
- **Bytes Out/sec** - This is the number of bytes returned by the ColdFusion Server per second.
- **Page Hits/sec** - The number of web pages processed per second by the ColdFusion Server.
- **Queued Requests** - This is the number of requests currently waiting to be processed by the ColdFusion Server.
- **Running Requests** - This is the number of requests currently being actively processed by the ColdFusion Server.



Find

Find
Again



Top of
Chapter

← Back

MS Active Server Pages (ASP)

The following measurements are available for MS Active Server Pages (ASP):

- **% Total Processor Time (System)** - The % Total Processor Time is the average percentage of time that all the processors on the system are busy executing non-idle threads. On a multi-processor system, if all processors are always busy this is 100%, if all processors are 50% busy this is 50% and if 1/4th of the processors are 100% busy this is 25%. It can be viewed as the fraction of the time spent doing useful work. Each processor is assigned an Idle thread in the Idle process which consumes those unproductive processor cycles not used by any other threads.
- **Request Bytes Out Total** - The total size, in bytes, of responses sent to clients. This does not include standard HTTP response headers.
- **Requests Executing** - The number of requests currently executing.
- **Requests Queued** - The number of service requests on the queue.
- **Requests/sec** - The number of requests executed per second.
- **Transactions Aborted** - The number of transactions aborted.
- **Transactions Pending** - Number of transactions in progress.

 Find Find Again Top of Chapter Back

SilverStream

The following measurements are available for the SilverStream server:

- **#Idle Sessions** - Number of sessions in the Idle state.
- **Avg.Request processing time** - Average request processing time.
- **Bytes Sent/sec** - The rate at which data bytes are sent from the Web server.
- **Current load on Web Server** - Percentage of load utilized by SilverStream web server, scaled at a factor of 25.
- **Hits/sec** - The HTTP request rate.
- **Total sessions** - The total number of sessions.
- **Free memory** - The total amount of memory in the Java Virtual Machine currently available for future allocated objects.
- **Total memory** - The total amount of memory in the Java Virtual Machine.
- **Memory Garbage Collection Count** - The total number of times the JAVA Garbage Collector has run since the server has started.
- **Free threads** - The current number of threads not associated with a client connection and available for immediate use.
- **Idle threads** - The number of threads associated with a client connection, but not currently handling a user request.
- **Total threads** - The total number of client threads allocated.

 Find Find Again Top of Chapter Back

WebLogic

The following measurements are available for the WebLogic server:

- Server Table

Lists all WebLogic Servers that are being monitored by the agent. A server must be contacted or be reported as a member of a cluster at least once before it will appear in this table. Servers are only reported as a member of a cluster when they are actively participating in the cluster, or shortly thereafter.

ServerState - State of the WebLogic Server, as inferred by the SNMP agent. Up implies the agent can contact the server; down implies the agent cannot contact the server.

ServerLoginEnable - This value is true if client logins are enabled on the server.

ServerMaxHeapSpace - Maximum heap size for this server, in KB.

ServerHeapUsedPct - Percentage of heap space currently in use in the server.

ServerQueueLength - Current length of the server execute queue.

SeverQueueThroughput - Current throughput of execute queue, expressed as number of requests processed per second

ServerNumEJBDeployment - Total number of EJB deployment units known to the server.



ServerNumEJBBeansDeployed - Total number of EJB beans actively deployed on the server.

- **Listen Table**

The set of protocol, ip address, port combinations on which servers are listening. There will be multiple entries for each server: one for each (protocol,ipAddr,port) combination.

If clustering is used, then the clustering related MIB objects will assume a higher priority.

ListenPort - Port number

ListenAdminOK - True if admin requests are allowed on this (protocol,ipAddr,port), else false.

ListenState - listening if the (protocol, ipAddr, port) is enabled on the server; notListening if it is not. The server may be listening but not accepting new clients if its serverLoginEnable state is false. In this case existing clients will continue to function but new ones will not.

- **ClassPath Table**

Table of classpath elements for Java, WebLogic Server and servlets. There are multiple entries in this table for each server; there may be multiple entries for each path on a server.

If clustering is used, then the clustering related MIB objects will assume a higher priority.



Find Again



Top of Chapter

← Back

CPTYPE - The type of CP element: Java, weblogic, servlet. A Java CPTYPE means the cpElement is one of the elements in the normal Java classpath. A weblogic CPTYPE means the cpElement is one of the elements in weblogic.class.path. A servlet CPTYPE means the cpElement is one of the elements in the dynamic servlet classpath.

CPINDEX - Position of element within its path. The index starts at 1.

WebSphere Graph

The following measurements are available for the WebSphere server:

- Run Time Resources
 - BeanData
 - BeanObjectPool
 - OrbThreadPool
 - DBConnectionMgr
 - TransactionData
- ServletEngine
- Sessions



Find
Again



 Top of
Chapter

 Back

Network Graphs

You can use Network graphs to determine whether your network is causing a delay in the scenario. You can also determine the problematic network node.

This chapter describes:

- **Network Monitor Graph**
- **Network Monitor Segment Graph**
- **Timeout Count Graph**
- **Verifying the Network as a Bottleneck**

About Network Monitoring

Network configuration is a primary factor in the performance of applications and web systems. A poorly designed network can slow client activity to unacceptable levels.

In an application, there are many network segments. A single network segment with poor performance, can affect the entire system.

Network graphs let you locate the network related problem so that it can be fixed.



Find
Again

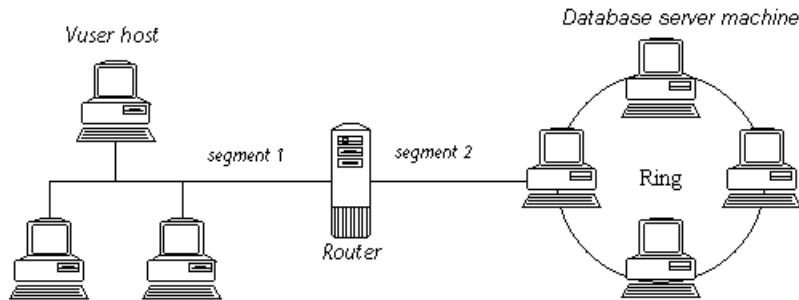


 Top of
Chapter

 Back

Understanding Network Monitoring

The following diagram shows a typical network. In order to go from the database server machine to the Vuser machine, data must travel over several segments.



You can use analyze the Network Monitor graphs to help you improve your system. The available graphs are:

- **Network Monitor Graph** - to help you determine whether the network is the bottleneck
- **Network Monitor Segment Graph** - to help you locate the problematic segment
- **Timeout Count Graph** - displays the number of timeouts that occurred during the scenario run.



Find Again



Top of Chapter

← Back

In order for the Analysis to generate Network Monitor graphs, you must activate the Network monitor before executing the scenario. In the Network Monitor settings, you specify the path you want to monitor. For information about setting up the Network monitor, see the *Astra LoadTest Controller User's Guide*.

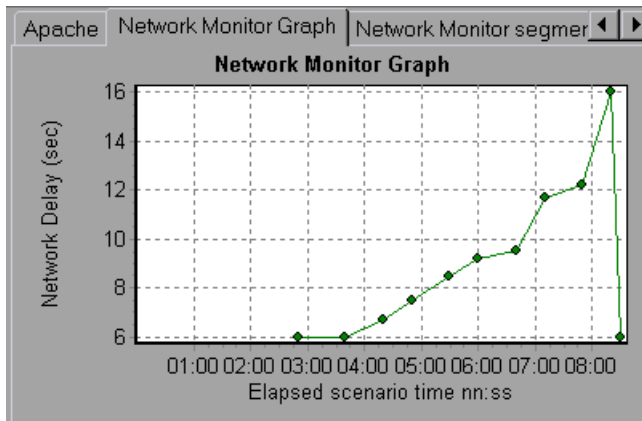
Network Monitor Graph

The **Network Monitor** graph shows the delays for the complete path between the source and destination machines (for example the database server and Vuser host). The graph maps the delay as a function of the elapsed scenario time.

Each path defined in the Controller is represented by a separate line with a different color in the graph.



The following graph shows the network delay as a function of the elapsed scenario time. It shows a delay of 16 seconds in the 8th minute of the scenario.



Find

Find Again



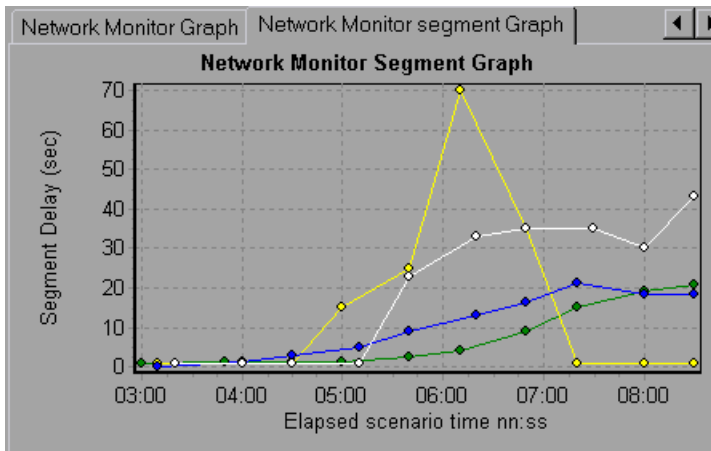
Top of Chapter

Back

Network Monitor Segment Graph

The **Network Monitor Segment** graph shows the delay for each segment of the path according to the elapsed scenario time. Each segment is displayed as a separate line with a different color.

In the following example, four segments are shown. The graph indicates that one segment caused a delay of 70 seconds in the sixth minute.

[Find](#)[Find Again](#)[Top of Chapter](#)[Back](#)

Timeout Count Graph

The Timeout Count graph shows when network timeouts occurred during the scenario. When Astra LoadTest sends a packet of information and the packet does not return before the timeout period, it is indicated in the Timeout Count graph.

You set the Packet Roundtrip Timeout value in the Controller's Monitoring Options. The default value is 4000 milliseconds. For more information, see the *Astra LoadTest Controller User's Guide*.

In the Timeout Count graph, each segment is displayed by a different color line. You can apply a filter to the graph to display only specific segments, or those segments with a significant delay. For information on filtering, see Chapter 1, [Understanding Astra LoadTest Analysis](#).



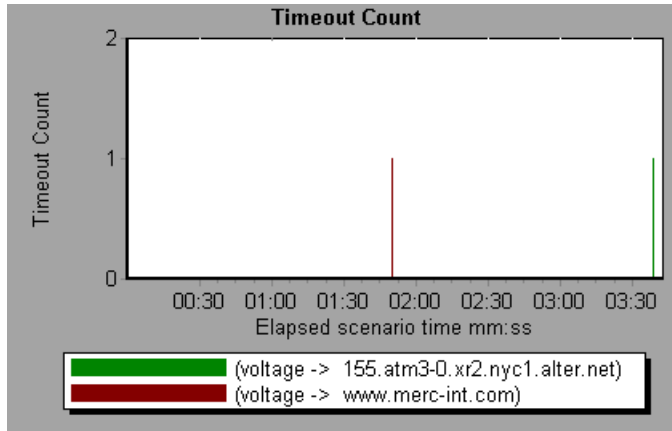
Find
Again




Top of
Chapter

 Back

In the following example, timeouts occurred twice during the scenario.

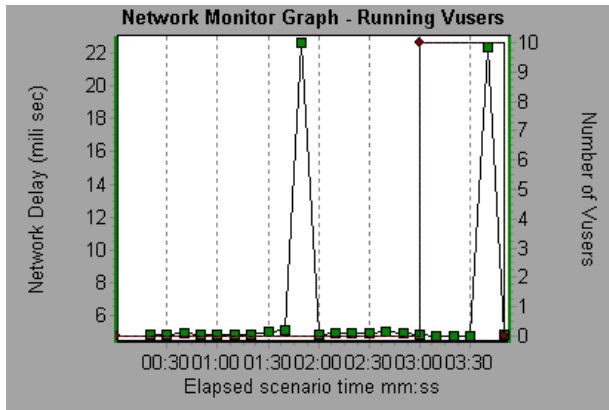


You can compare this graph to the Network Monitor Segment graph to determine how each timeout affected the delay for that path.

Verifying the Network as a Bottleneck

You can use merge various graphs to determine if the network is a bottleneck. For example, using the **Network Monitor** and **Running Vusers** graphs, you can determine how the number of Vusers effects the network delay. The **Network Monitor** graph indicates the network delay during the scenario run. The **Running Vusers** graph shows the number of running Vusers.

In the following merged graph, the network delays are compared to the running Vusers. The graph shows that when all 10 Vusers were running, a a network delay of 22 milliseconds occurred, implying that the network may be overloaded.



Find

Find Again



Top of Chapter

Back

Cross Result and Merged Graphs

The Analysis utility lets you compare results and graphs to determine the source of a problem.

This chapter discusses:

- **Cross Result Graphs**
- **Generating Cross Result Graphs**
- **Merging Graphs**
- **Creating a Merged Graph**

About Cross Result and Merged Graphs

Comparing results is essential for determining bottlenecks and problems. You use Cross Result graphs to compare the results of multiple scenario runs. You create Merged graphs to compare results from different graphs within the same scenario run.



Cross Result Graphs

Cross Result graphs are useful for:

- benchmarking hardware
- testing software versions
- determining system capacity

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



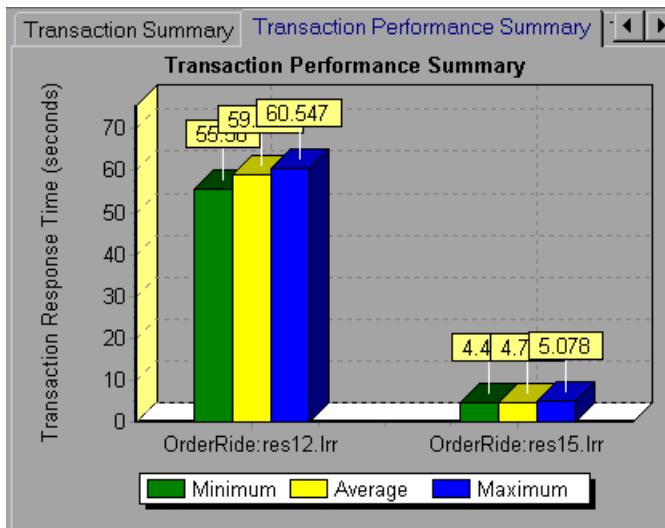
Find
Again



Top of
Chapter

← Back

In the following example, two scenario runs are compared by crossing their results, *res12*, and *res15*. The same script was executed twice—first with 100 Vusers and then with 50 Vusers. In the first run, the average transaction time was approximately 59 seconds. In the second run, the average time was 4.7 seconds. It is apparent that the system works much slower with a greater load.



Find Again



Top of Chapter

Back

The Cross Result graphs have an additional filter and group by category: *Result Name*. The above graph is filtered to the *OrderRide* transaction for results *res12*, and *res15*, grouped by *Result Name*. If the graph was not grouped by the result name, the results from the two scenario runs would not be displayed separately.

The filter conditions for the above graph are:

| Filter condition: | | |
|-----------------------------|----------|----------------------|
| | Criteria | Values |
| Transaction Name | = | OrderRide |
| Transaction Response Time | | |
| Scenario Elapsed Time | | |
| Result Name | = | res12.lrr, res15.lrr |
| Transaction End Status Name | | |
| UserID | | |

The **Group By** settings for the above graph are:

| Group By: | |
|---|---|
| Available groups: | Selected groups: |
| <div style="border: 1px solid black; padding: 5px;"> Transaction End Status Name UserID </div> | <div style="border: 1px solid black; padding: 5px;"> Result Name </div> |
| <input type="button" value="▶"/> <input type="button" value="◀"/> | <input type="button" value="▲"/> <input type="button" value="▼"/> |

 **Find**

Find Again





Top of Chapter

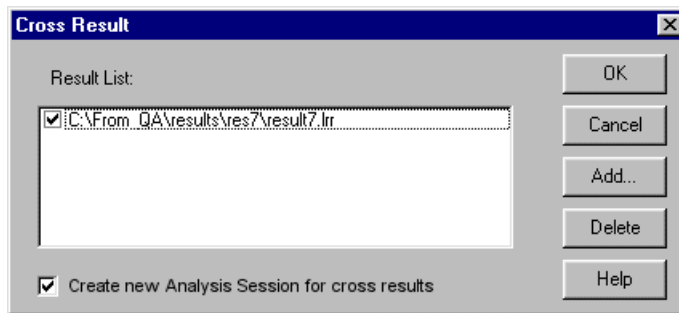
 **Back**

Generating Cross Result Graphs

You can create a Cross Result graph for two or more result sets.

To generate a Cross Result graph:

- 1 Choose **File > Cross With Result**. The Cross Results dialog box opens.



- 2 Click **Add** to add an additional result set to the **Result List**. The Select Result Files for Cross Results dialog box opens.
- 3 Locate a results directory and select its result file (*.lrr*). Click **OK**. The scenario is added to the Result List.
- 4 Repeat steps 2 and 3 until all the scenarios you want to compare are in the Result List.



- 5 When you generate a Cross Result graph, by default it is saved as a new Analysis session. To save it in an existing session, clear the **Create New Analysis Session for Cross Result** check box.
- 6 Click **OK**. The Analysis processes the result data and asks for a confirmation to open the default graphs.

After you generate a Cross Result graph, you can filter it to display specific scenarios and transactions. You can also manipulate the graph by changing the granularity, zoom, and scale. For more information, see Chapter 1, [Understanding Astra LoadTest Analysis](#).



Merging Graphs

The Analysis lets you merge the results of two graphs from the same scenario into a single graph. The merging allows you to compare several different measurements at once. For example, you can make a merged graph to display the network delay and number of running Vusers, as a function of the elapsed time.

In order to merge graphs, their x-axis must be the same measurement. For example, you can merge Web Throughput and Hits per second, because the common x-axis is Scenario Elapsed Time. The drop-down list only shows the active graphs with an x-axis common with the current graph. The Analysis provides three types of merging:

- Overlay
- Tile
- Correlate

Overlay: Superimpose the contents of two graphs that share a common x-axis. The left y-axis on the merged graph shows the current graph's values. The right y-axis shows the values of the graph that was merged. There is no limit to the number of graphs that you can overlay. When you overlay two graphs, the y-axis for each graph is displayed separately to the right and left of the graph. When you overlay more than two graphs, the Analysis displays a single y-axis, scaling the different measurements accordingly.



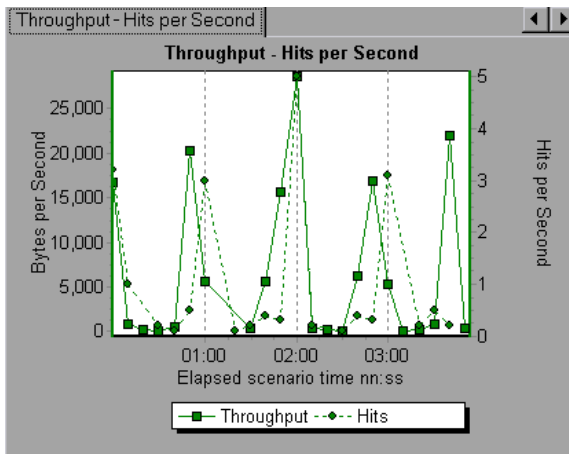
Find
Again



Top of
Chapter

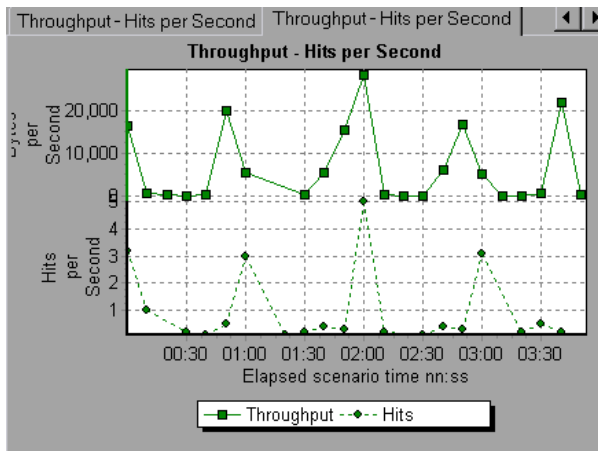
← Back

In the following example the Throughput and Hits per Second graph are overlaid with one another.



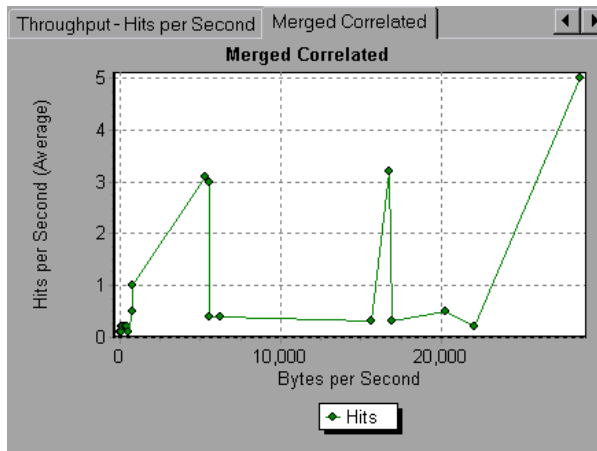
- Find
- Find Again
-
- Top of Chapter
- Back

Title: View contents of two graphs that share a common x-axis in a tiled layout, one above the other. In the following example the Throughput and Hits per Second graph are tiled one above the other.



Correlate: Plot the y-axis of two graphs against each other. The active graph's y-axis becomes the x-axis of the merged graph. The y-axis of the graph that was merged, becomes the merged graph's y-axis.

In the following example, the Throughput and Hits per Second graph are correlated with one another. The x-axis is Bytes per Second (the Throughput measurement) and the y-axis shows the Hits per Second.



Creating a Merged Graph

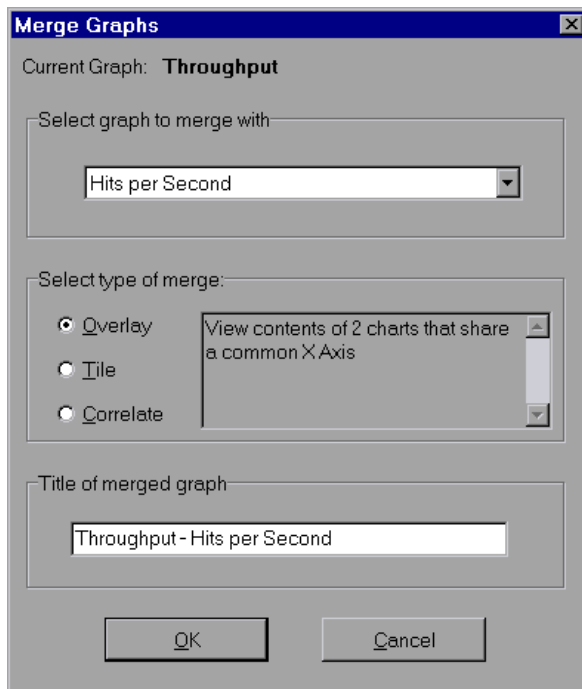
You can merge all graphs with a common x-axis.

To create a merged graph:

- 1 Select a graph in the tree view or click on its tab to make it active.



- 2 Choose **View > Merge Graphs** or click the **Merge Graphs** button. The Merge Graphs dialog box opens and displays the name of the active graph.



- 3 Select a graph with which you want to merge your active graph. Only the graphs with a common x-axis to the active graph, are available.
- 4 Select the type of merge: Overlay, Tile, or Correlate.
- 5 Specify a title for the merged graph. By default, the Analysis combines the titles of the two graphs being merged.
- 6 Click **OK**.
- 7 Filter the graph just as you would filter any ordinary graph.



Understanding Analysis Reports

After running a scenario, you can use the Analysis reports to analyze the performance of your application.

This chapter describes:

- **Viewing Summary Reports**
- **Creating HTML Reports**

About Analysis Reports

After running a scenario, you can view reports that summarize your system's performance. The Analysis provides the following reporting tools:

- Summary report
- HTML reports

The Summary report provides general information about the scenario run. You can view the Summary report at any time from the Analysis window.

You can instruct the Analysis to create an HTML report. The Analysis creates an HTML report for each one of the open graphs.



Find
Again



← Back

Viewing Summary Reports

The Summary report provides general information about scenario execution. This report is always available from the tree view or as a tab in the Analysis window. It lists statistics about the scenario run and provides links to the following graphs: Running Vusers, Throughput (Web), Hits Per Second (Web), and Transaction Summary. At the bottom of the page, the Summary report displays a table containing the scenario's run-time data.

The screenshot shows a software window titled "Analysis Summary" with a period of "24/05/2000 15:20". The window contains the following information:

- Scenario Name:** R:\LR_TESTS\Lrun7.lrs
- Results in session:** R:\LR_TESTS\asp_test1\res12\res12.lrr.
- Duration:** 03 minutes and 57 seconds.

Below this is a "Statistics Summary" section with the following data:

- Maximum Running Vusers:** 10
- Total Throughput:** 146,655
- Throughput (bytes/second):** Average: 6,666 Max: 28,593
- Total Hits:** 19
- Hits per Second:** Average: 1 Max: 5

Next is a "Transaction Summary" section showing:

- Transactions:** Total passed: 200 Total failed: 0

At the bottom, a table displays transaction performance data:

| Transaction Name | Minimum | Average | Max |
|-------------------------|---------|---------|------|
| confirm_asp_Action1_184 | 0.01 | 0.017 | 0.03 |

Find

Find Again

◀ ▶

Top of Chapter

← Back

Creating HTML Reports

The Analysis lets you create HTML reports for your scenario run. It creates a separate report for each one of the open graphs and a Summary report. The Summary report is identical to the Summary report that you access from the Analysis window. The report also provides a link to an Excel file containing the graph data.

Analysis Summary Period: 24/05/2000 15:00

Scenario Name: R:\LR_TESTS\Lrun7.lrs
Results in session: R:\LR_TESTS\asp_test1\res12\res12.lrr.
Duration: 03 minutes and 57 seconds.

Statistics Summary

- **Maximum Running Users:** 10
- **Total Throughput:** 146,655
- **Throughput (bytes/second):** Average: 6,666 Max: 28,593
- **Total Hits:** 19
- **Hits per Second:** Average: 1 Max: 5

Transaction Summary

- **Transactions:** Total passed: 200 Total failed: 0

| Transaction Name | Minimum | Average | Ma |
|------------------|---------|---------|----|
| | | | |

Find


Find Again

◀ ▶

Top of Chapter

← Back

To create HTML reports:

- 1 Open all graphs that you want to be included in the report.
- 2  Choose **Reports > HTML Report** or click the **Create HTML Report** button. The Save dialog box opens.
- 3 Specify a path and file name for the HTML report and click **OK**. The Analysis saves a Summary report in the selected folder and the rest of the graphs in a folder with the same name. When you create an HTML report, the Analysis opens your default browser and displays the Summary report.
- 4 To view an HTML report for one of the graphs, click on its link in the left frame.
- 5 To copy the HTML reports to another location, be sure to copy the filename and the folder with the same name. For example, if you named your HTML report *test1*, copy *test1.html* and the folder *test1* to the desired location.

 FindFind
Again Top of
Chapter Back

Managing Results Using TestDirector

The integration of the Analysis with TestDirector lets you manage results and sessions using TestDirector projects. TestDirector helps you organize and manage all scripts, scenarios, and results using a repository.

This chapter describes:

- **Opening a Connection to a TestDirector Project**
- **Creating a New Session Using TestDirector**
- **Opening an Existing Session Using TestDirector**
- **Saving Sessions to a TestDirector Project**

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



Find
Again



Top of
Chapter

← Back

About Managing Results Using TestDirector

The Analysis works together with TestDirector, Mercury Interactive's test management tool. You can store both sessions and results in a TestDirector repository and organize them into unique groups. You can also retrieve results from the TestDirector repository, and save the Analysis sessions with your specific graph settings, locally.

You can instruct the Analysis to automatically connect to the TestDirector server upon startup.

Opening a Connection to a TestDirector Project

Before you can work with a TestDirector project, you must open a connection to the server hosting the project. After connecting to the TestDirector server, you still have complete access to the regular file system.



Find Again



Top of Chapter


← Back

To open a connection to a TestDirector project:



- 1 Choose **Tools > TestDirector Connection** from the Analysis window.

Connection to TestDirector

Server Connection

Server:  Connect

Project Connection

Project:   Connect

User Name:

Password:

Allow Connection to Local Database

Reconnect on startup

Save password for reconnection on startup



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

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

Connection to TestDirector

Server Connection

Server:

Project Connection

Project:

User Name:

Password:

Allow Connection to Local Database

Reconnect on startup

Save password for reconnection on startup

- 4 Provide the user login and password.
- 5 Select the **Reconnect on Startup** check box to automatically open the connection to the TestDirector server and project when you start the Analysis.



- 6 If you select Reconnect on Startup, you can save the specified password to automate the login process. Select the Save Password for Reconnection on Startup check box.
- 7 Click **Connect**. A message appears indicating that you are connecting to a project.
- 8 Click **Close** to close the Connection to TestDirector dialog box.

Creating a New Session Using TestDirector

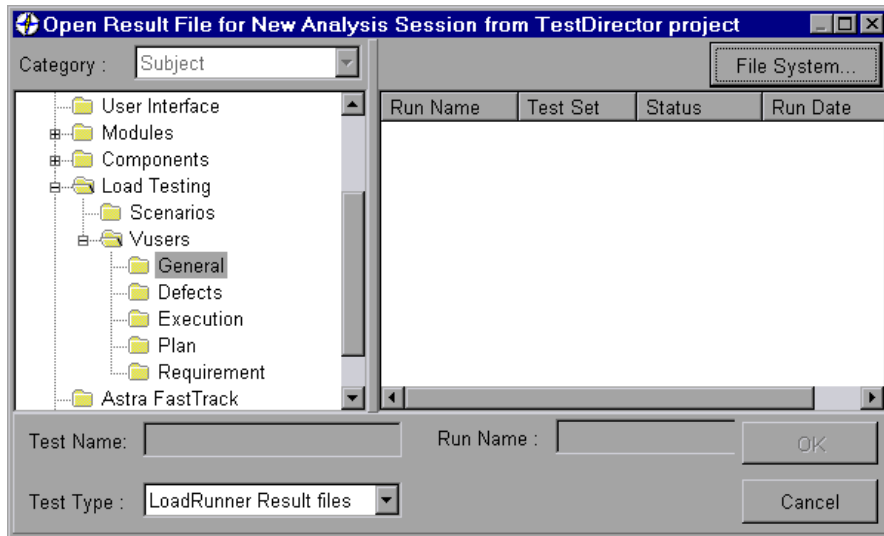
You can create a new Analysis session using a result file (.lrr extension) located in the TestDirector repository or file system.

To create a new session using results from the TestDirector repository:

- 1 Connect to the TestDirector server (see [Opening a Connection to a TestDirector Project](#) on page 112.)



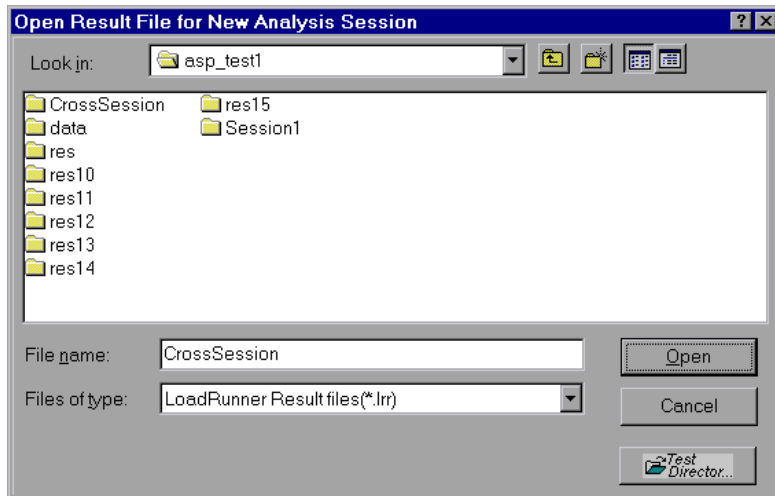
- 2 Choose **File > New** or click the **New File** button. The Open Result File for New Analysis Session from TestDirector Project dialog box opens.



- 3 Select a result file from the project.
- 4 Click **OK**. The Analysis loads the results.



- 5 To select a session file from the standard file system, click the **File System** button in the upper right corner of the dialog box. The Open Result File for New Session dialog box opens.



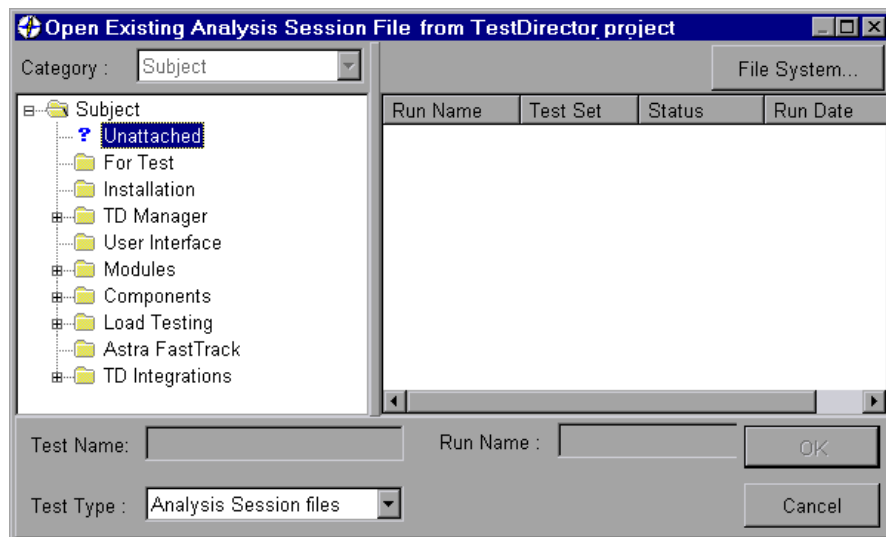
To return to the TestDirector dialog box, click the **TestDirector** button in the bottom right of the dialog box.

Opening an Existing Session Using TestDirector

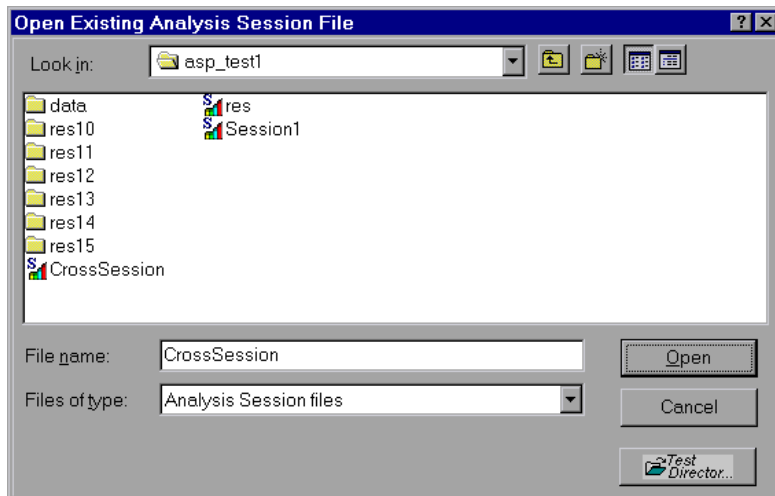
You can open an existing Analysis session through the standard file system or the TestDirector repository.

To open a session from the TestDirector repository:

- 1 Open a connection to the TestDirector server (see [Opening a Connection to a TestDirector Project](#) on page 112.)
- 2 Choose **File > Open** or click the **File Open** button. The Open Existing Analysis Session File from TestDirector Project dialog box opens.



- 3 Select a session from the project.
- 4 Click **OK**. The Analysis loads the scenario. The name of the session appears in the banner. The session's graphs are displayed in the tree-view.
- 5 To select a session file from the standard file system, click the **File System** button in the upper right corner of the dialog box. The Open Existing Analysis Session File dialog box opens.



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

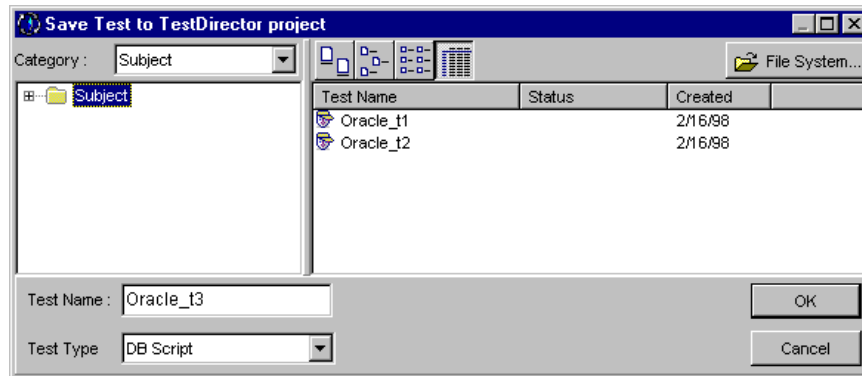


Saving Sessions to a TestDirector Project

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

To save a session to the TestDirector project:

- 1 Connect to the TestDirector server (see [Opening a Connection to a TestDirector Project](#) on page 112.)
- 2 Choose **File > Save As**. The Save to TestDirector Project dialog box opens.



- 3 Type a session name, select a folder and click **OK**.
- 4 To save the session to disk, click **File System** and save the file as you would on a standard file system.

Find

Find Again



Top of Chapter

Back

Index

- A**
 - advanced display settings
 - chart [37](#)
 - series [38](#)
 - Analysis
 - overview [6–41](#)
 - sessions [8](#)
 - Apache graph [67](#)
 - Astra LoadTest Analysis [7, 107](#)
- C**
 - chart settings [37](#)
 - collating results [9](#)
 - connecting to a TestDirector server [112](#)
 - coordinates of a point [20](#)
 - Cross Result graphs [95–106](#)
- D**
 - display options
 - advanced [36](#)
 - standard [34](#)
 - drill down [21](#)
- E**
 - enlarging graphs [24](#)
 - Excel file
 - exporting to [32](#)
 - viewing [109](#)
- F**
 - filtering graphs [12](#)
- G**
 - granularity [25](#)

 Find

Find
Again



 Top of
Chapter

 Back

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

graph

- Apache [67](#)
- Average Response Time [47](#)
- Hits per Second [64](#)
- Network Monitor [89](#)
- Network Monitor Segment [91](#)
- Rendezvous [57](#)
- Response Time-Under Load [49](#)
- Running Vusers [56](#)
- SNMP [60](#)
- Throughput [65](#)
- Throughput Graph [65](#)
- Timeout Count [92](#)
- Transaction Distribution [50](#)
- Transaction Performance Summ. [45](#)
- Transaction Summary [44](#)
- TransactionPercentile [52](#)
- Transactions per second [46](#)
- Vuser Summary [55](#)

graph types

- Network [87–94](#)
- Rendezvous [57](#)
- System Resource [58–60](#)
- Transaction [42–53](#)
- Vuser [54–55](#)
- Web Application Server [68](#)
- Web Resources [61–86](#)
- Web Server [63](#)

graphs, working with

- background [37](#)
- crossing results [95–106](#)
- display options [34](#)
- merging [101](#)
- overlying, superimposing [101](#)

H

- Hits per Second graph [64](#)
- HTML reports [109](#)

L

- legend [28](#)
- legend preferences [37](#)

M

- marks in a graph [38](#)
- Merging graphs [101](#)

N

- Network graphs [87–94](#)
- Network Monitor graph [89](#)
- Network Monitor Segment graph [91](#)



A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

O

output window [7](#)
 overlay graphs [101](#)

R

raw data [31](#)
 Rendezvous graph [57](#)
 reports [107–110](#)
 HTML [109](#)
 summary [108](#)
 Running Vusers State graph [56](#)

S

scale factor [62](#)
 scale of graph [25](#)
 sessions [8](#)
 SNMP graph [60](#)
 spreadsheet view [31](#)
 Summary report [108](#)
 superimposing graphs [101](#)

T

TestDirector
 connecting to a server [112](#)
 integration [111–120](#)
 opening a new session [115](#)
 opening an existing session [118](#)
 saving sessions to a project [120](#)
 three-dimensional properties [38](#)
 Throughput graph [65](#)
 Timeout Count graph [92](#)
 tool options
 setting database options [41](#)
 setting general options [39](#)
 Transaction graphs [42–53](#)
 Transaction Performance Summ. graph [45](#)
 Transaction Response Time graphs [47–53](#)
 Average [47](#)
 Distribution [50](#)
 Percentile [52](#)
 Under Load [49](#)
 Transaction Summary graph [44](#)
 Transactions per Second graph [46](#)



A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

V

- Vuser graphs [54–55](#)
- Vuser Summary graph [55](#)

W

- Web Application Server graphs [68](#)
 - Broadvision graph [69](#)
 - ColdFusion graph [80](#)
 - Microsoft Active Server Pages (ASP) graph [82](#)
 - SilverStream graph [83](#)
 - WebLogic graph [84](#)
 - WebSphere graph [86](#)
- Web Resource graphs [61–86](#)
- Web Server Resource graphs [63](#)

X

- x-axis interval [25](#)

Z

- zoom [24](#)

 FindFind
Again Top of
Chapter Back

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

Astra LoadTest Analysis User's Guide, Version 4.5

© Copyright 2000 by Mercury Interactive Corporation

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

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

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

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

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

Mercury Interactive Corporation
1325 Borregas Avenue
Sunnyvale, CA 94089
Tel. (408) 822-5200 (800) TEST-911
Fax. (408) 822-5300

ALTANUG4.5/01

