ProTune®

Console User's Guide

Version 1.0

MERCURY INTERACTIVE

ProTune Console User's Guide, Version 1.0

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

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

PTCONUG1.0/02

Table of Contents

PART I: UNDERSTANDING PROTUNE

PART II: MAPPING A BUSINESS PROCESS

PART III: DESIGNING A TUNING SESSION

PART IV: EXECUTING A TUNING SESSION

PART V: MONITORING A SESSION

PART VI: TUNING YOUR SYSTEM

PART VII: APPENDIXES

Welcome to ProTune

Welcome to ProTune, the proactive solution for optimizing production systems. ProTune's system-wide approach to optimization is a product of Mercury Interactive's expert knowledge and tuning methodologies. ProTune guides you through the tuning process in four easy steps: topology, design, execute, and tune.

Online Resources

ProTune includes the following online tools:

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

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

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

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

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

Support Information presents the locations of Mercury Interactive's Customer Support Web site and home page, and a list of Mercury Interactive's offices around the world.

Mercury Interactive on the Web uses your default Web browser to open Mercury Interactive's home page (http://www.mercuryinteractive.com). This site enables you to browse the knowledge base and add your own articles, post to and search user discussion forums, submit support requests, download patches and updated documentation, and more.

ProTune Documentation Set

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

- install ProTune
- create scripts
- use the ProTune Console
- use the ProTune Analysis

Using the ProTune Documentation Set

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

Installation Guide

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

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

Console User's Guide

The ProTune documentation pack includes one Console user's guide:

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

Analysis User's Guide

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

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

Guide for Creating Scripts

The ProTune documentation pack includes one guide for creating scripts.

The *ProTune Creating Vuser Scripts User's Guide* describes how to create scripts using VuGen. When necessary, supplement this document with the *Online Function Reference* and the *WinRunner User's Guide* for creating GUI scripts.

For information on	**Look here...**
Installing ProTune	*ProTune Installation Guide*
The ProTune testing process	*ProTune Console User's Guide*
Creating scripts	*ProTune Virtual User Generator User's Guide*
Creating and running sessions	*ProTune Console User's Guide*
Analyzing test results	*ProTune Analysis User's Guide*

Typographical Conventions

This book uses the following typographical conventions:

1, 2, 3	Bold numbers indicate steps in a procedure.
➤	Bullets indicate options and features.
>	The greater than sign separates menu levels (for example, **File** > **Open**).
Stone Sans	The **Stone Sans** font indicates names of interface elements on which you perform actions (for example, “Click the **Run** button.”).
Bold	**Bold** text indicates method or function names
Italics	*Italic* text indicates method or function arguments, file names or paths, and book titles.
Helvetica	The Helvetica font is used for examples and text that is to be typed literally.
<>	Angle brackets enclose a part of a file path or URL address that can vary (for example, *<Product installation folder>/bin*).
[ ]	Square brackets enclose optional arguments.
{ }	Curly brackets indicate that one of the enclosed values must be assigned to the current argument.
...	In a line of syntax, an ellipsis indicates that more items of the same format may be included.

Part I

Understanding ProTune

1

Planning a Tuning Session

The main purpose of ProTune is to enable the user to explore his network, detect bottlenecks and assist during the tuning phase. ProTune combines the network topology, predefined tuning sessions and goals to a set of tests that pinpoint problematic components in the client network. In addition, ProTune helps the more advanced user to explore and tune the business processes by providing a simple and organized methodology.

About Planning a Tuning Session

Organizations worldwide are investing millions of dollars annually to keep their complex production applications operating at peak levels. They recognize that the success of these systems depends on delivering the availability, reliability, scalability and security demanded by end users. Production tuning is the act of optimizing and tuning complex production systems, thus creating substantial improvements in system performance.

Production tuning helps businesses optimize and maximize utilization of their current infrastructure. It creates cost savings by reducing unnecessary hardware and software usually purchased to solve performance problems. An efficient production system safeguards applications that generate revenue in B2B and B2C models and helps companies realize the opportunities that their CRM, ERP and other business applications provide. Production tuning can also validate how a company's application runs, providing a third-party perspective of the performance and reliability of the system.

During development, system optimization seems easily attainable through an effective combination of functional testing, load testing and test management. Many organizations, however, are now discovering that their greatest challenges await them once the applications are moved into production.

In a typical application lifecycle, R&D develops an application and then sends it through a quality verification and validation process managed by the QA department. QA engineers engage in several rounds of defect identification and code correction and then deem the application ready for release.

The application is then moved into production. Despite this pre-deployment testing, applications commonly experience performance problems in the production environment. Operations groups must react quickly to these problems, but often do not have the proper tools to help them reproduce these problems and determine their causes. When they are unable to isolate the causes of these problems in the production environment, many operations groups resort to a trial and error approach to tuning. This can lead to unnecessary purchases of additional software licenses and hardware, in the hopes that they will solve the elusive performance problems. While this gives outsiders the impression of capacity planning, it uses a "hammer" approach rather than a precise methodology, and is not a cost-effective or efficient way to improve the performance or reliability of a production system.

Utilizing ProTune, Mercury Interactive's production tuning solution, you can tune each tier and element of a production system's architecture systematically. This allows you to achieve superior results and increased capacity with existing software, hardware and personnel. ProTune provides multi-layer tuning, capacity planning, security validation under load, capacity limit and reliability testing to reach the maximum system performance with the optimum configuration.

Reasons for Testing and Tuning Systems in Production

Why does an application fail in production after being thoroughly tested by QA?

Scaled-Down Version

The QA lab uses a scaled-down version of the production system. An organization's QA lab may consist of one front-end server, one application server and a database—just enough hardware and software to emulate a fully functional application. This system is used to test application logic and scalability, as well as perform acceptance testing. The production system, however, is often very different. It may, for example, have four front-end servers, two applications servers, a middleware server, one database, a back-up disk array, linked to a "back office" pulling some data from a legacy system, and a third-party service. As a result, the application and various server configuration settings that are tuned for maximum performance in the QA lab are unlikely to be the optimal settings for the production environment. Since scalability is never linear, having a CPU server in QA will not extrapolate accurately to the 24 CPU server in production.

Absence of a Network

All network devices and protocols must work together efficiently. Since the network resides in front of the application, any performance problems in the network will impact the end user. Furthermore, if the network serves more than one application, these performance issues will negatively impact every application on the network. In addition, these performance issues will have a cumulative effect. As the network receives more traffic for each application, these inefficiencies will begin to play a larger role in slowing the traffic, causing all applications to perform increasingly slower and forcing more traffic to wait on the network. Application speed is therefore irrelevant if the network has performance or routing issues. Without testing in a networked environment, such issues are often invisible and therefore difficult to detect and isolate.

Internet Connectivity

The QA lab does not have the Internet connectivity and accompanying infrastructure. For Web-based applications being accessed via the Internet, items such as proper bandwidth, ISP peering, border routers, LAN, WAN and latency all play a critical role in the way applications work. Yet operations groups often do not factor in response time and latency when creating time-out settings. Since user traffic must travel over the Internet, the network and then the application, Internet issues will have extensive effects on the entire architecture. For those relying on third parties or using the Web service approach, not having this connective to the live provider can provide misleading results.

Lacking Security Infrastructure

Security systems and strategies reside both in the application and the infrastructure. As a result, security devices and strategies not optimally configured, patched, or tuned can lead to performance degradations across the entire system, as well as present serious security risks. Enabling too much security (e.g., strong encryption) can consume more bandwidth and CPU cycles and must therefore be accounted for. Intrusion detection systems (IDS) and denial of service (DoS) systems that intentionally audit incoming and outgoing traffic for attack or malicious signatures pose additional performance overhead. Making predictions about application performance, without validating or tuning the security system, can be disastrous.

Additional Causes of Failure

The situations described above can be compounded by a lack of communication between the QA, operations/production and security teams. Today's applications rely on hundreds of components (hardware, software and protocols) from a myriad of suppliers working in concert to deliver a business solution. As a result, systems that have not been patched, tuned or configured optimally can bring a multi-million dollar system to a crawl.

ProTune's Approach to Solving Performance Problems

ProTune's approach is designed to solve performance problems in the three core areas: the infrastructure, application, and security.

First you use the ProTune scripts to test the network and the Internet infrastructure, since these layers will ultimately affect all other system components. Next, focus on the application, optimizing it to achieve the desired response times for both peak and off-peak usage. As a final step, work with security devices, both software and hardware, to ensure the desired security levels are achieved without negatively impacting application and infrastructure.

Once the system has been optimized in each of these three key areas—infrastructure, application, and security—operations groups can ascertain the system's scalability and reliability. In addition, they can identify problematic components within the entire system, including a faulty server, server memory leakage, limited bandwidth or a misconfigured firewall. By following this methodology, operations groups come away with a clear list of what must be addressed to increase capacity and avoid investing time and money in hardware and/or software that cannot solve the current performance problems.

Understanding the Infrastructure

The *infrastructure* is defined as all the physical elements of the system, including the network, the servers and other devices. You define the infrastructure in a topology diagram within the ProTune Console. In general, the infrastructure is the most complicated piece of the entire system, consisting of many moving parts that rely on a multitude of configuration settings and software. Each device in turn must be configured to work with the entire infrastructure and not interfere with the software.

Misconfigurations originating within the infrastructure can have far-reaching effects. All applications will appear to perform poorly on a misconfigured infrastructure. A load balancer, for example, that is misconfigured will not utilize the servers properly. Yet it also creates less obvious problems, including issues with the firewall due to "unplanned" IP routing, as well as bandwidth problems. In this way, infrastructure bottlenecks are similar to peeling an onion. Operations groups often discover, upon further investigation, that most bottlenecks are obscuring other bottlenecks. Without properly testing and tuning a system, IT groups cannot possibly uncover all of these bottlenecks, which can surface at inopportune times.

ProTune tests each tier of the infrastructure in an iterative fashion to uncover hidden bottlenecks. Working with the step-by-step methodology, you can validate that new bottlenecks are not created as old ones are removed and that application operation and reliability are not negatively impacted.

The following is a list of the elements that you can individually tune using the ProTune Console:

- Web Server
- Application Server
- Mail Server
- FTP Server
- Groupware Server
- Database Server
- Firewall
- Router
- Injector
- DNS Server
- Load Balancer

The process used to test each infrastructure element is through *canned* or prepared scripts that perform typical operations on each element.

Tuning the Load Balancer

This section describes a common performance problem found within the load balancer element of the infrastructure, including the symptoms, causes and corrective actions.

Given the load balancer's strategic location in the infrastructure and extensive routing functions, a faulty load balancer can impact the entire system—both the application and the infrastructure. It is therefore critical to control the path to the load balancer in order to tune it properly. This may require testing in multiple locations.

Symptoms

- End users report sporadic performance or access issue or proxy-related errors
- Users experience a significant number of HTTP errors (e.g., 404, 505, etc.)

- Load-balanced Web servers are not distributing load properly, as observed by CPU
- X% of users experience poor performance while Y% do not experience the same conditions (improper routing of traffic)

Validation

- Design and execute a load test that bypasses the load balancer (goes to one of the Web servers directly) to determine whether the problematic behavior can be reproduced
- Emulate a large number of users engaging in a wide variety of actions, such as browsing, logging on, searching and buying. If the load balancer is working correctly, each of the Web servers should be sharing the load.
- Design and execute a load test that tests the load balancing strategy (Multiple load test farms and different domains are used to test this aspect of the load balancer configuration.)
- Provide a connection test to validate that the load balancer and server can support the proper number of connections and that the load balancer connection table is updated and refreshed

Corrective Actions

The actual corrective actions depend on the results of the validation stage.

- Validate the proper load balancer strategy for full equipment utilization
- Tune and optimize load balancer settings and validate with additional performance tests

Tuning Applications

The application is the ultimate barometer of system performance, since it is where the end user interfaces with the systems. It is therefore critical to tune an application in the production environment, since this reflects real-world conditions. To be most effective, the tuning must anticipate and replicate a wide range of real-user behavior and actions (log in, browse, search, purchase, track, account information, etc.) on a wide variety of connections, user speeds, browsers and other variables.

Most important, the application must be able to serve the target number of users accurately and with acceptable response times. Clearly the system must be able to scale to handle everyday peak traffic, but it must also be able to respond to unanticipated increases in traffic. Also, the system must be scalable at each layer of the application—from the Web servers to the database. Since unusual traffic surges often deviate from the normal traffic patterns, each type of business process must be able to perform independently and simultaneously when the load occurs.

In addition to scalability issues, an application's settings and operations can create conflicts with the hardware on which the application resides. This adds another level of complexity, reinforcing the need for a systematic approach to performance optimization.

Database Servers

The database server is the engine that delivers most of the information and facilitates nearly every complex transaction.

Symptoms

- Complex and transaction-based processes are slow (e.g. login, searching or buying items)
- Database intensive operations are slow; Web pages and independent applets are fast
- High database CPU utilization

Validation

- Test both transactional (database intensive) and simple Web pages to validate the performance disparity
- Force a high number of concurrent connections to the database to test server (e.g., database communication [protocol], connection, connection release and process capacity)
- Create specific tests to generate database-intensive requests and large amounts of information to determine database design and efficiency

Corrective Actions

The actual corrective actions depend on the results of the validation stage.

- Improve the database indexing and remove outdated statistics
- Compact and de-fragment the database
- Improve the caching algorithms, based on real traffic conditions
- Increase the number of connections between the server and database, and configure the memory accordingly
- Consider redesigning the database if the application and database have conflicts with protocols, I/O or lock duration
- Optimize the SQL
- Share resource management/general database configuration

Web Server

The Web server is critical for directing user requests to the correct resource, whether that is the application server, cache server or even the database.

Symptoms

- Only unusually low numbers of users can access the site before performance degrades for all processes
- High incidence of 404 errors
- Application-related processes are faster than Web-page delivery
- Large volumes of users degrade performance for all types of transactions; network latency is low;
- hits/second and throughput rates are low
- Server CPU or memory resources are low
- Server connection resources get tied up

Validation

- Test standard Web page delivery and application requests to determine whether one performs in a dramatically different manner than the other, and that application servers are being equally loaded
- Determine whether common pages are cached correctly when larger numbers of users browse popular areas

- Create session with many users to validate that the number of connections can be sustained and released to the application server, database and cache servers/accelerators
- Test with large number of sustained users accessing the site to determine memory (amount, allocation and leaks) and CPU usage problems

Corrective Actions

The actual corrective actions depend on the results of the validation stage.

- Configure servers to load application servers equally Improve and tune caching algorithm and connection settings
- Add or reallocate physical memory
- Fix server issues with memory leaks and/or connection capability

Tuning Security Elements

Many organizations have security systems in place, but they have limited means to verify that it is working properly. Occasionally, only when the sites are attacked, do they realize that their security systems were not working.

A misconfigured security system, unlike other components, does not provide any obvious symptoms, such as a 404 error message. Nonetheless, a single misconfigured component—an IP, router, VPN access—can render the security system completely useless and leave the site vulnerable to attack.

Security devices directly impact application performance. Improperly configured security devices not only can present security breaches, but they also can significantly degrade application and infrastructure performance.

In addition to misconfiguration, security systems can be compromised by high volumes of traffic. As network traffic patterns change and load increases, open ports are exposed. New sources of vulnerability surface as additional servers and resources come online to help manage the load. In addition, devices such as firewalls and IDS systems, can create bottlenecks under heavy traffic and then fail completely.

Most hackers will try to breach security during high traffic time for two reasons—to disrupt business and to sneak in unnoticed. Therefore, it is critical to run security tests while the site is under peak user loads.

ProTune provides a way to tune the security system to maximize security while maintaining the performance of the entire production system. It launches scripts that emulate real users conducting typical business processes on the Web site. The following sections describe the tuning of HTTPS and firewalls, along with their symptoms and recommended corrective actions.

HTTPS/SSL

HTTPS and SSL help ensure the security of transactions, but they also add complexity to transactions and require additional resources (bandwidth and CPU power) in order to function properly.

Symptoms

- Slow performance with low user levels
- Web server CPUs are working at maximum capacity with low user levels
- Web pages and applications are both slow, but database operations are fast
- Dramatic increase in response time for a low number of users

Validation

- Set up hundreds or thousands of concurrent users to determine connection limit, release and renew
- Contrast Web and application speed with a non-secure “search” function without full-page refresh
- Test that only valid objects and transactions have SSL
- Set up persistent connection to test release and memory leak possibility

Corrective actions

The actual corrective actions depend on the results of the on results of the validation stage.

- Set SSL /HTTPS only on critical transactions and material

- Make sure Web server CPU can handle load (encryption/decryption)
- Possible need for SSL accelerators
- Terminate open connections after a determined amount of idle time
- Fine-tune application to stop memory leaks

Firewalls

Firewalls are intended to keep out hackers, as well as block the entry of malicious packets. Because the firewall must interact with every data packet, a misconfigured firewall can contribute to several performance issues.

Symptoms

- Very poor performance with high volumes of users
- A high number of dropped connections and time outs
- Slow response time, but Web and application servers have low utilization and few connections

Validation

- Temporarily disable firewall (for range of IPs) and look for any performance increases during its absence
- Verify that network traffic metrics show significant delay on the firewall segment
- Create standard business process mix, look for slow response time. Web and application servers should have low utilization and few connections.
- Validate the same number of attempted transaction matches with Web server connection counts
- Test bandwidth and throughput though the firewall using a low number of concurrent users accessing pages quickly

Corrective Actions

- Optimize firewall configuration settings then retest
- Possibly add additional firewalls or upgrade hardware

2

The ProTune Testing Process

ProTune is the proactive solution for optimizing production systems. ProTune’s system-wide approach to optimization is a product of Mercury Interactive’s expert knowledge and tuning methodologies. ProTune guides you through the tuning process in a few easy steps: topology, design, execute, analyze, and tune.

Creating a Topology

You begin your tuning session with ProTune’s topology mapping tool. After creating your system topology, you define monitors, alerts, and responses that both measure the state of your systems and establish safe performance thresholds. If an event triggers an alert, the effected component flashes on ProTune’s topology map, to immediately pinpoint the problem. By doing so, ProTune helps ensure the stability of your production system as you tune.

Designing a Tuning Session

Next comes the design phase. Based upon your topology, ProTune provides a recommended menu of steps that will comprise the foundation of your tuning session. ProTune’s recommended scripts not only save you time, but also provide structure and expert guidance on how to begin your tuning sessions. To expand the functionality of ProTune’s recommended scripts, you can also create custom scripts for your application. Custom scripts, combined with the scripts included in ProTune, give you the power of Mercury Interactive's tuning expertise while maintaining the flexibility you need for your unique business needs.

Executing a Tuning Session

Once you finish designing, you can begin to tune. During the tuning session, ProTune's monitors and analysis display your performance in real time. ProTune begins with a baseline measurement of your system, and then moves on to more rigorous tuning exercises that focus on isolating troublesome areas within your infrastructure, application, and security systems.

Analyzing Your System

During the session execution, ProTune records your application's performance under different loads. You use ProTune Analysis graphs and reports to analyze the application's performance. By analyzing your system as a whole, while focusing on specific components, ProTune allows you to take corrective action and unleash your system's hidden potential.

Fine-Tuning your System

Once ProTune helps to identify the problem areas, you use the integrated tuners to make changes to a variety of system components from a single convenient interface. After fine-tuning, ProTune helps you to run your tuning session again, validate results, and graphically display the resulting performance improvements.

Part II

Mapping a Business Process

3

Creating a Topology

The first step in mapping the business process is defining a topology of the servers used in the business process. This chapter describes how to create a topology.

This chapter discusses:

- Building a Topology Diagram
- Setting the Component Properties
- Selecting Monitors
- Automatically Assigning Monitors and Alerts

About Creating a Topology

The first step in creating a test for tuning your session is defining a topology. The topology refers to the architecture of the system. You indicate the servers and their types, such as database, Web, or application. You also define how the servers are connected. The topology also includes other hardware such as routers, firewalls, and load generator machines.

ProTune provides several topology templates. The following table describes them:

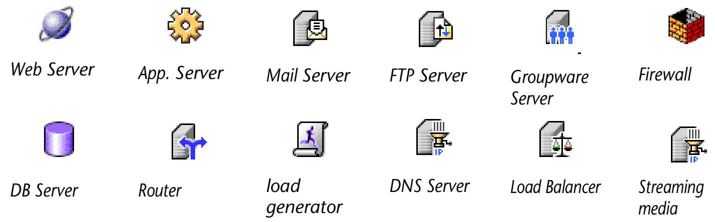
	Web Servers	App. Servers	DB Servers	Router	Firewall	load generator
Single App. Server	-	1	-	1	-	1
Single Database Server	-	-	1	1	-	1
Single Web Server	1	-	-	1	-	1
3-Tier FWLdBWebAppDb	3	3	1	1	2	1
3-Tier LdBWebAppDb	3	3	1	3	-	1
3-Tier Two System Arch	2	2	2	-	4	2
Three Tier WebAppDb	1	1	1	3	-	1
2-Tier WebAppServer	1	1	-	2	-	1
2-Tier WebDB	1	-	1	2	-	1

You can use one of these templates or modify an existing one. You can also create your own topology from an empty template. When you add items to the topology, ProTune arranges the elements in the most common layout. You can rearrange this layout or delete any unnecessary components.

Building a Topology Diagram

Before building a topology diagram, make sure you have a clear understanding of your system architecture. An incorrect representation of your system will result in inaccurate results.

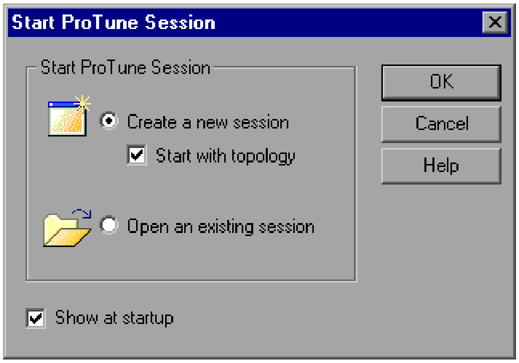
Using ProTune's design window, you can drag objects from the design palette into the topology diagram. The design palette contains the following objects:



You can use an existing topology, base your topology on a template, or create a new topology from scratch. ProTune allows you to modify templates and existing topologies to suit your specific needs.

To open the System Topology window:

1 Invoke ProTune Console by choosing **Start** > **Programs** > **ProTune** > **Console**. ProTune displays the Start ProTune Session window:


Start ProTune Session

Start ProTune Session

● Create a new session

☑ Start with topology

○ Open an existing session

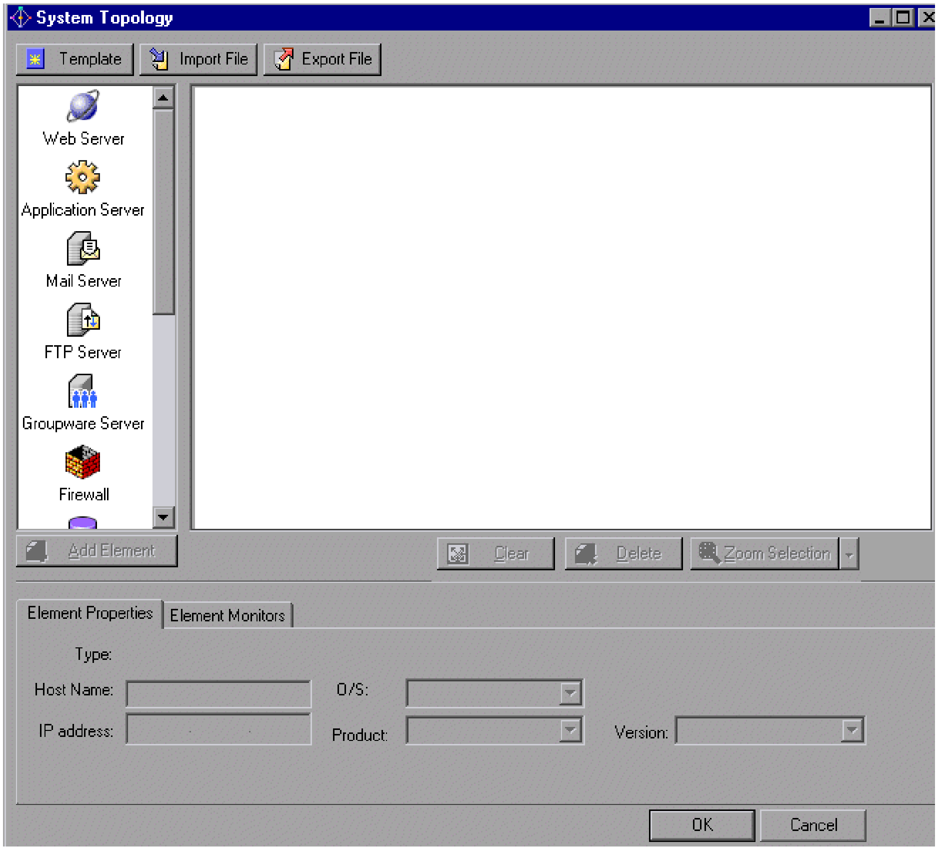
☑ Show at startup

You can create a new session or open an existing one. If you choose to create a new session, you can also decide whether you want the System Topology window to be displayed immediately.

2 Click **OK**.

3 If the Console is running but the System Topology window is not displayed, click the **Topology** button or choose **Tools** > **System Topology** to open it.

The System Topology window is displayed:



Note: You can also open the System Topology window by double-clicking the white space in the Execute View's topology pane. See "Running a Session," on page 119.

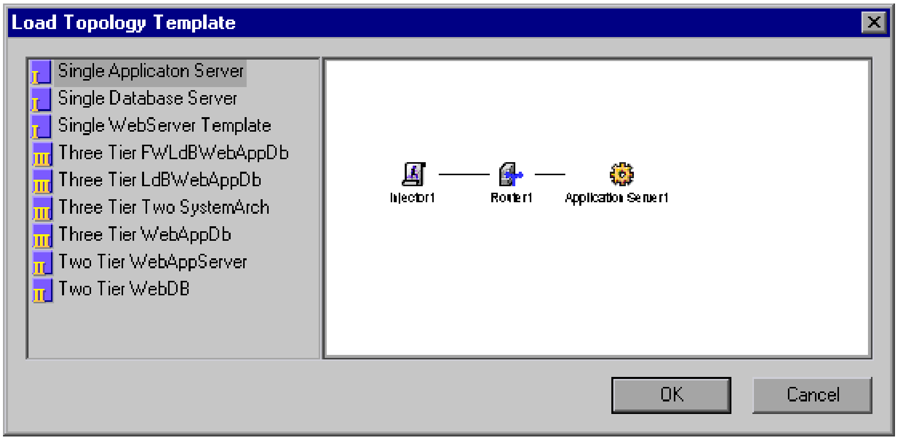
To import an existing topology diagram:

1 Click **Import File** and browse to the desired topology (a file with a *.tpl* extension).

2 Click **Open**. ProTune displays your topology.

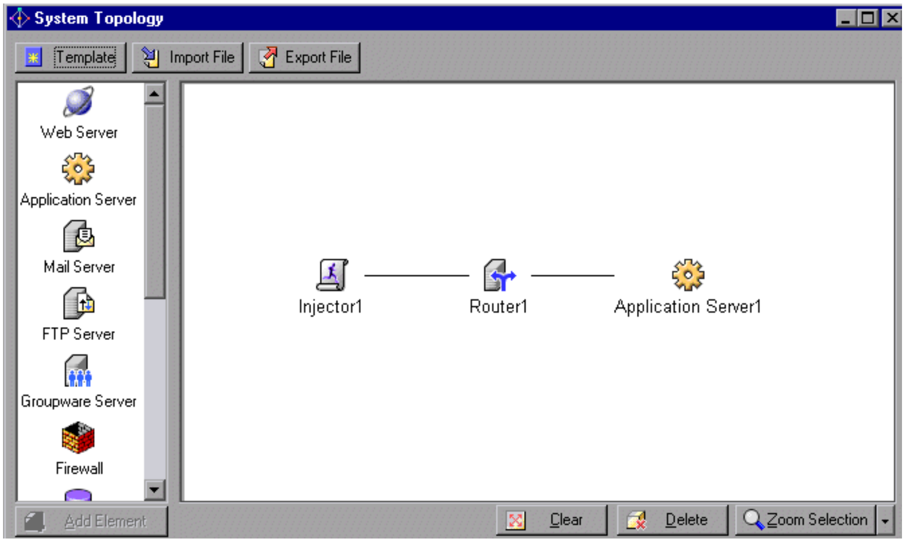
To create a topology diagram from a template:

1 Click **Template** to open the Load Topology Template dialog box.



2 Select a template and click **OK**.

The topology diagram you selected appears in the right pane, as in the following example:

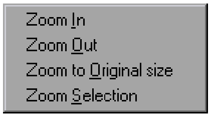


To create or modify a topology:

1. To add a component to a topology, select the component in the left pane and click **Add Element**, or drag the component from the left pane into the right pane.
2. To change a component's position, click the component and drag it to its new position in the topology.
3. To connect components in the topology, drag a line from one component to the other.
4. To delete a component, select the component by clicking it in the diagram, and click **Delete**.
5. To delete all the components from the topology diagram, click **Clear**.

To zoom in on a component:

1. Select the component and click **Zoom Selection**.
2. Click the arrow to the right of the **Zoom Selection** button to view all zoom options:



3. Click **Zoom In** to increase the size of the topology diagram.
4. Click **Zoom Out** to reduce the size of the topology diagram.
5. Click **Zoom to Original Size** to restore the original size of the diagram.

To save the topology diagram for use in another session:

1. Click **Export File**.
2. Browse to the desired location, specify a filename with a *.tpl* extension, and click **Save**.

Setting the Component Properties

After you create a topology diagram, you specify the properties for each component. ProTune builds a tuning session based on these settings.

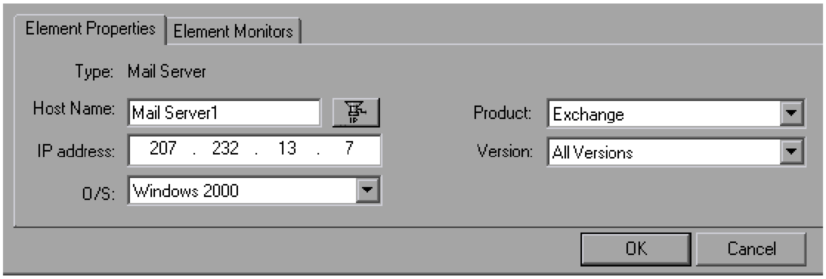
You can specify the following properties for each component:

- host name
- IP address
- operating system
- product
- version.

To specify a component's properties:

1 Select the component whose properties you want to specify.

2 Click the **Element Properties** tab.



Type: Mail Server

Host Name: Mail Server1

IP address: 207 . 232 . 13 . 7

O/S: Windows 2000

Product: Exchange

Version: All Versions

3 Specify the component's name in the **Host Name** box, or accept the default name.

4 ProTune resolves the component's IP address from the host name. If you want to view the IP address, click the **Resolve IP** button adjacent to the **IP address** box. ProTune displays the component's IP address in the **IP address** box.

5 Select your component's operating system from the **O/S** list.

6 Select the product or vendor for the selected component from the **Product** list.

7 Select the application's version from the **Version** list.

Selecting Monitors

After you define each component's properties, you select the measurements that you want to monitor for the component.

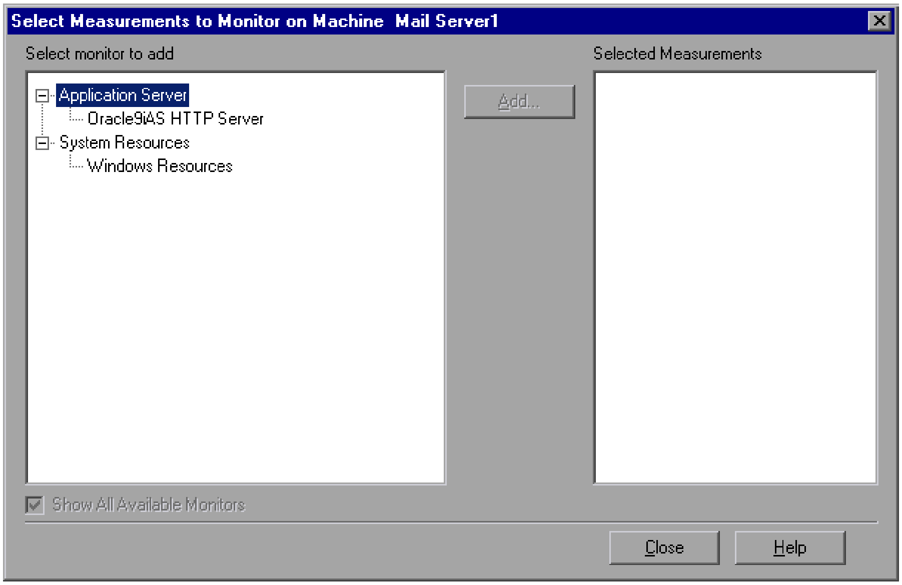
Note: You need to define the component properties before selecting the monitors, since ProTune connects to the components to determine which monitors are available.

To select a measurement to monitor:

1. In the topology diagram, select the component whose properties you want to monitor.
2. Click the **Element Monitors** tab and then click **Add**. The **Select measurements to monitor dialog** box opens.

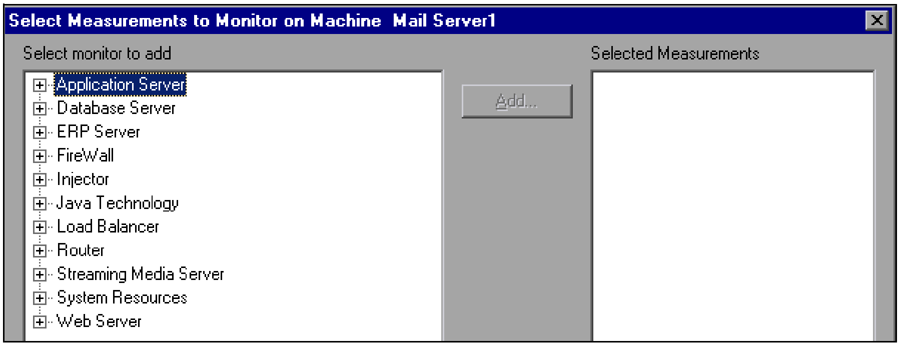
Note: To select monitors and measurements when you are not in the System Topology window, click the **Monitors** button on the main toolbar. See "Choosing Monitors and Measurements," on page 164.

If the ProTune knowledge base contains a list of monitors for your element, ProTune displays only those monitors available for the component in its specified operating system, product and version, as in the following example:



Select Measurements to Monitor on Machine Mail Server1

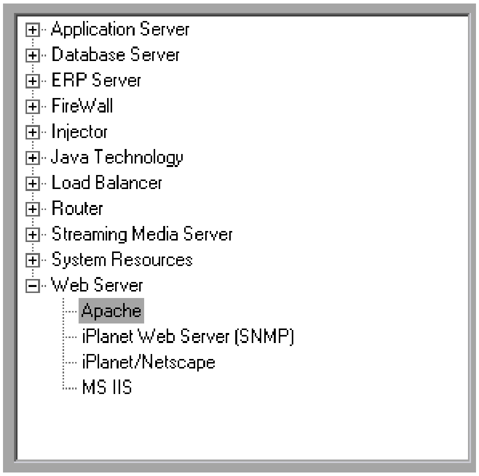
If the knowledge base does not contain a list of monitors for the specified element (possibly because you haven't specified all the element's properties), ProTune automatically displays all its monitors:



Select Measurements to Monitor on Machine Mail Server1

You can also display all the monitors in the knowledge base by clicking the **Show All Available Monitors** box.

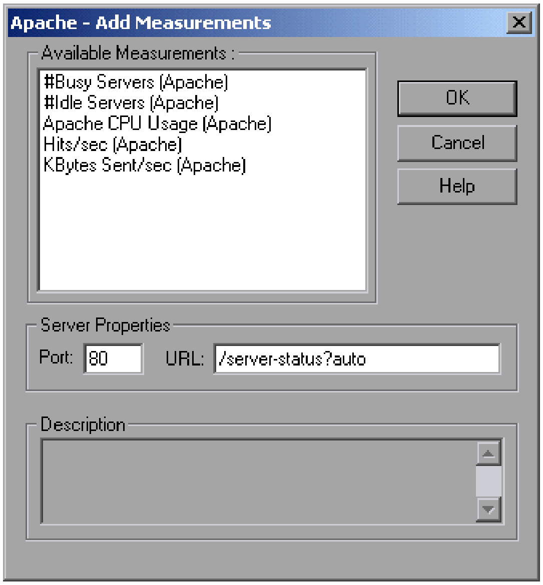
The full list of available monitors is displayed as a list of component types. If the monitor you need is not visible, expand the appropriate component type to display the list of components in your category. For example, clicking Web Server shows you the list of Web Server applications whose measurements you can monitor:



- Application Server
- Database Server
- ERP Server
- FireWall
- Injector
- Java Technology
- Load Balancer
- Router
- Streaming Media Server
- System Resources
- Web Server
  - Apache
  - iPlanet Web Server (SNMP)
  - iPlanet/Netscape
  - MS IIS

3 Click a component to select it and then click **Add**, or double-click the component. For most component types, ProTune opens a dialog box with a list of the component's available measurements. For example, if you choose

Apache, ProTune displays the following list of Apache measurements that you can monitor:



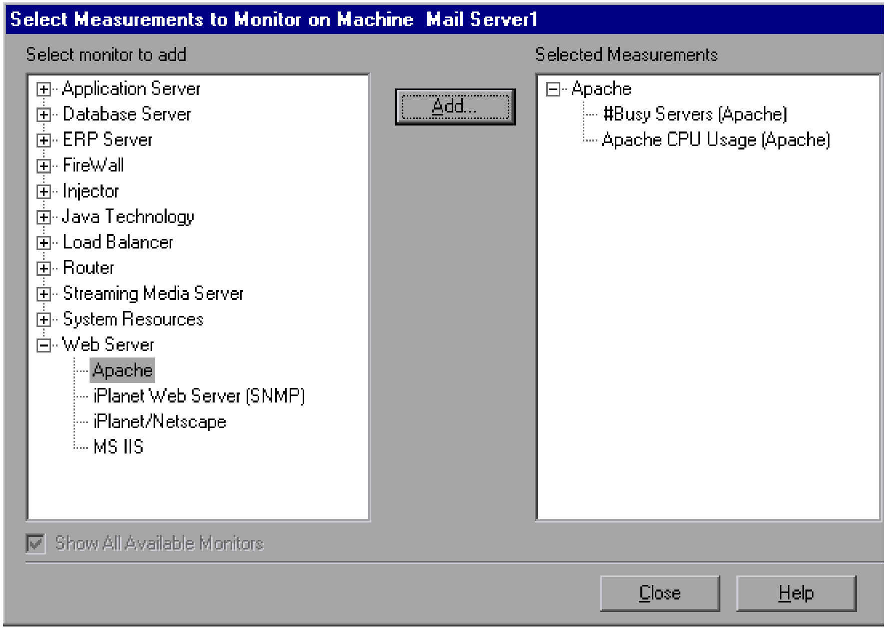
#Busy Servers (Apache)
#Idle Servers (Apache)
Apache CPU Usage (Apache)
Hits/sec (Apache)
KBytes Sent/sec (Apache)

Port: 80 URL: /server-status?auto

Tip: For information on how to specify the measurements specific to each component, see the appropriate chapter in the section "Monitoring a Session," on page 159.

---

4 Select the measurements that you want to monitor, and click **OK**. ProTune adds the measurements to the Selected Measurements pane:



Select Measurements to Monitor on Machine Mail Server1

Select monitor to add

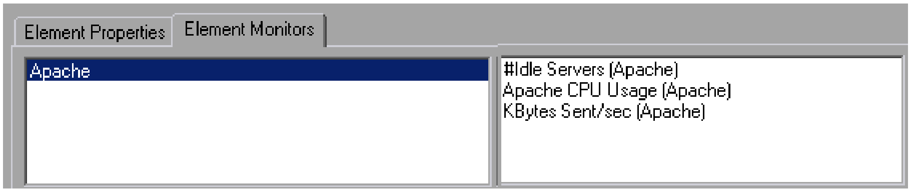
- Application Server
- Database Server
- ERP Server
- FireWall
- Injector
- Java Technology
- Load Balancer
- Router
- Streaming Media Server
- System Resources
- Web Server
  - Apache
  - iPlanet Web Server (SNMP)
  - iPlanet/Netscape
  - MS IIS

Selected Measurements

- Apache
  - #Busy Servers (Apache)
  - Apache CPU Usage (Apache)

☑ Show All Available Monitors

5 Repeat steps 1 through 4 for all the components that you want to monitor, and then click **Close**. ProTune displays the newly added monitors in the Element Monitors section.



Element Monitors

Apache

#Idle Servers (Apache)
Apache CPU Usage (Apache)
KBytes Sent/sec (Apache)

Automatically Assigning Monitors and Alerts

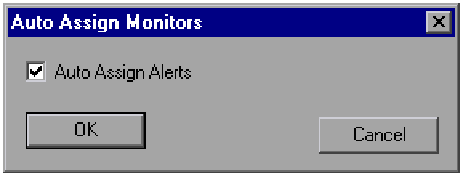
ProTune's Auto Assign feature assigns monitors automatically to a selected component, based on the component's type. This allows you to skip the task of choosing monitors for your components. For example, if your component is an Apache Web Server, the Auto Assign feature assigns to your component all the monitors for all the Apache measurements.

Tip: For details of ProTune's monitors, see "Monitoring a Session," on page 159.

In addition to automatically assigning monitors, you can also automatically assign alerts. This assigns a default set of alerts for various measurements, based on definitions in ProTune's knowledge base. For information on alerts, see "Defining Alerts," on page 79.

To automatically assign monitors and alerts to a component:

1 In the topology diagram, select the component whose properties you want to monitor, and click **Auto Assign Monitors**. Alternatively, you can right click the component and click **Auto Assign Monitors**. ProTune displays the Auto Assign Monitors dialog box:



☑ Auto Assign Alerts

2 To automatically assign alerts, check the **Auto Assign Alerts** box.

3 Click **OK**. ProTune assigns the monitors to the selected components, and displays them in the Element Monitors section of the Topology window. If you chose to assign alerts, you can view the assigned alerts by clicking the **Alerts Definition** button.

Part III

Designing a Tuning Session

4

Adding Session Steps

After creating a topology, you add steps to your session. Each step reflects a specific business process, or part of a whole business process. This chapter describes how to create and manage session steps.

This chapter discusses:

- Getting Started with Designing a Session
- Understanding Session Steps
- Adding Session Steps
- Managing Steps
- Managing Scripts
- Setting an Initial Load (Manual Profiles)
- Configuring Script Details
- Using Relative Paths for Scripts

About Adding Session Steps

For your tuning session, you create test steps. Each step performs a specific business process. ProTune provides a set of *canned* or prepared scripts. These scripts include tests for the Web Infrastructure in the following areas:

- Infrastructure
- Network Servers
- Application Servers

You can use the canned scripts or write your own and incorporate them in the session steps.

You select one or more scripts for each session step. You can also indicate upon which servers to execute these steps. When selecting multiple scripts, you can specify whether the scripts should all be part of one session step or each script should be contained in a separate step. Since ProTune tests single steps, combining scripts in a step causes them to be tested together.

The arguments for each script and their default values are displayed in the right pane of the Add Step window. You can specify a value other than the default.

After setting up the steps for your session, you can indicate the number of Vusers you want to emulate. Vusers are virtual users that emulate real users. You can also choose a goal for each session step.

Getting Started with Designing a Session

Designing your tuning session consists of the following steps:

Add Test Steps: Add steps that emulate a business process to the session, for each server. For more information, see "Adding Session Steps," on page 42.

Manage Test Steps: Manage the steps that you created by arranging their order, adding scripts, disabling them temporarily, or deleting them. For more information, see "Managing Steps," on page 45.

Define a Schedule Profile: Specify how and when the step should run, and whether it should be goal oriented. For more information, see "Scheduling Session Steps," on page 95.

Assign Scripts: Once you have defined a goal, you can use the built-in script or assign additional scripts to emulate other aspects of the business process. For more information, see "Managing Scripts," on page 46.

Set Alerts: You set alerts for your session. ProTune issues an alert for the element that reaches the specified threshold. For more information, see Chapter 7, "Defining Alerts."

Set the Initial Load: Indicate the number of Vusers you want to emulate in your session. For more information, see "Setting an Initial Load (Manual Profiles)," on page 50.

Configure the Script: Configure the values for your script. For more information, see Chapter 4, "Configuring Script Details."

Schedule the Step: Set up a schedule for your session steps in order to automate them and simulate the true environment. For more information, see Chapter 8, "Scheduling Session Steps.".

Set the Run-Time Settings: Set the run-time settings for each script. These relate primarily to the pacing of the script, the think time, and the logging options. For information on configuring the run-time settings, refer to the *ProTune Creating Virtual User Scripts* guide.

Understanding Session Steps

ProTune's built-in scripts help you test the Connection capacity or rate goals. You can create session steps in the following areas:

- Infrastructure
- Network Servers
- Application Servers

Infrastructure

The Infrastructure steps relate to the infrastructure of your network architecture. The available sub-steps are:

DNS Request Rate: Checks the rate at which host names are resolved on the Domain Name Server. The script title as it appears in the script list is *dns_rqst_rate*. The supported server is the DNS server. **Note:** The DNS protocol is not supported by UNIX.

Network Servers

The Network test steps relate to the TCP/IP connection of your servers. The available sub-steps are:

TCP Connection Capacity: Sustains simultaneous TCP connections with the target device. The script title as it appears in the script list is *tcp_conn_cpty*. The supported servers are Web servers, load balancers and database servers.

SSL Connection Capacity: Sustains simultaneous SSL connections with the target device. The script title as it appears in the script list is *tcp_ssl_conn_cpty*. The supported servers are Web servers, load balancers and database servers.

HTTP Connection Rate: Generates HTTP requests against a target URL, without keep-alive enabled. Therefore, each new request must establish a new connection with the Web server. The script title as it appears in the script list is *http_conn_rate_nokeepalive*. The supported servers are Web servers and load balancers.

HTTP Request Rate: Generates HTTP requests against a target URL, with keep-alive enabled. Therefore, each new request reuses the same connection with the Web server. Note: the Web server must configured to support keep-alive connections in order for this script to work properly. The script title as it appears in the script list is *http_rqst_rate_keepalive*. The supported servers are Web servers and load balancers.

HTTP Downstream Bandwidth: Generates downstream data transmission by a continuous download of a large file from the Web server via HTTP protocol. The HTTP requests are set with keep-alive enabled. Therefore each new request re-uses the same connection with the web server. The script title as it appears in the script list is *http_downstream_bandwidth*. The supported servers are Web servers and load balancers.

FTP Connection Capacity: Sustains simultaneous FTP connections with the file server. The script title as it appears in the script list is *ftp_conn_cpty*. The supported servers are FTP servers, Web servers and load balancers.

FTP Get File Rate: Generates FTP GET requests to download a specific file from an FTP server. The script title as it appears in the script list is *ftp_get_rate*. The supported servers are FTP servers, Web servers and load balancers.

FTP Put File Rate: Generates FTP PUT requests to upload a specific file to an FTP server. The script title as it appears in the script list is *ftp_put_rate*. The supported servers are FTP servers, Web servers and load balancers.

Application Servers

The Application Server steps test various Application and Database servers. The available sub-steps are:

SMTP Session Capacity: Sustains simultaneous SMTP connections with the mail server. The script title as it appears in the script list is *smtp_conn_cpty*. If a connection is closed by the server, it is re-established in order to sustain the same number of sessions. The supported servers are Mail Servers and Application Servers.

SMTP Send Mail: Submits email messages to a mail server via the SMTP protocol. The script title as it appears in the script list is *smtp_send_mail*. The supported servers are Mail Servers and Application Servers.

MAPI Session Capacity: Sustains simultaneous MAPI sessions with MS Exchange Server. These sessions are created only once. If the session terminates, it is not re-established. The script title as it appears in the script list is *mapi_conn_cpty*. The supported servers are Mail Servers and Application Servers.

MAPI Send Mail: Sends an email with a file attachment to the specified recipient, using the given MS Exchange profile. The script title as it appears in the script list is *mapi_send_mail*. The supported servers are Mail Servers and Application Servers.

POP3 Connection Capacity: Creates simultaneous POP3 session connections with the mail server. The purpose of this script is to test the maximum number of connections that the specified mail server allows. If not enough connections are allowed, new attempts to connect will be rejected and performance will suffer. The script title as it appears in the script list is *pop3_conn_cpty*. The supported servers are Mail Servers and Application Servers.

POP3 Retrieve Mail: Puts stress on the infrastructure involved in retrieving mail messages on a POP3 server. The script title as it appears in the script list is *pop3_retrieve_mail*. The supported servers are Mail Servers and Application Servers.

IMAP Session Capacity: Sustains simultaneous IMAP sessions with the mail server. The script title as it appears in the script list is *imap_conn_cpty*. The supported servers are Mail Servers and Application Servers.

IMAP Search Mail: Searches for mail in the specified folder using the given search criteria. The script title as it appears in the script list is *imap_search_mail*. The supported servers are Mail Servers and Application Servers.

IMAP Store Mail: Stores a mail message in the specified mail folder using the IMAP protocol. The *msg_file* parameter indicates the full path of the file that contains the mail message. The file *imapnote1.dat* is included as an example mail file. You can modify the example or replace it to suit your needs. The script title as it appears in the script list is *imap_store_mail_file*. The supported servers are Mail Servers and Application Servers.

Real Connection Capacity: Sustains simultaneous connections through the RTSP protocol with the Real server. The script title as it appears in the script list is *rtsp_conn_cpty*. The supported servers are Web Servers and Application Servers.

RealPlayer Play Media: Plays a media stream through the RTSP protocol. The script title as it appears in the script list is *rtsp_play_media*. The supported servers are Web Servers and Application Servers.

MMS Play Media: Plays a media stream via the Microsoft Media Stream (MMS) protocol. The script title as it appears in the script list is *mms_play_media*. The supported servers are Web Servers and Application Servers.

ADO-DB Connection Rate: Opens a new connection to the SQL server, executes the specified query on the server and closes the connection. The script title as it appears in the script list is *adodb_open_sql_close_rate*. The supported servers are Database Servers.

ADO-DB SQL Query Rate: Executes the specified SQL statement on the server. The connection is created only once at the script initialization stage. The script title as it appears in the script list is *adodb_sql_query_rat*e. The supported servers are Database Servers.

ODBC SQL Connection Rate: Executes the specified SQL statement on the database server. A new connection is opened in each iteration. The script title as it appears in the script list is *odbc_sql_connection_rate*. The supported servers are Database Servers.

ODBC SQL Query Rate: Executes the specified SQL statement on the database server using the same connection. The connection is created only once at the script initialization stage. The script title as it appears in the script list is *odbc_sql_query_rate*. The supported servers are Database Servers.

DDOS SYN Attack: The SYN denial of service attack produces intensive load on the network and on the victim host's CPU. It tries to open a new connection to the victim host and immediately close the connection. The appropriate behavior for a TCP/IP stack of load balancers or firewalls is detection of attack conditions and filtering. The script title as it appears in the script list is *DDOS_SYN_ATTACK*. The supported servers are Web Servers. **Note:** Can be run only on machines running Windows 2000.

DDOS FIN Attack: The FIN denial of service attack produces intensive load on the network and the victim host's CPU. It tries to close "never opened connections." The correct behavior for a TCP/IP stack of load balancers or firewalls is to ignore the request and drop the packet from the network. Unfortunately, most TCP/IP stacks implement incorrect behavior, and respond by issuing an ACK. The unnecessary network traffic abuses the CPU and the network. The script title as it appears in the script list is *DDOS_FIN_ATTACK*. The supported servers are Web Servers. **Note:** Can be run only on machines running Windows 2000.

DDOS RST Attack: The RST denial of service attack produces intensive load on the network and the victim host's CPU. It tries to reset "never opened connections." The correct behavior for a TCP/IP stack of load balancers or firewalls is to ignore the request and drop the packet from the network. Unfortunately, most TCP/IP stacks implement incorrect behavior, and respond by issuing an ACK. The unnecessary network traffic abuses the CPU and the network. The script title as it appears in the script list is

DDOS_RST_ATTACK. The supported servers are Web Servers. **Note:** Can be run only on machines running Windows 2000.

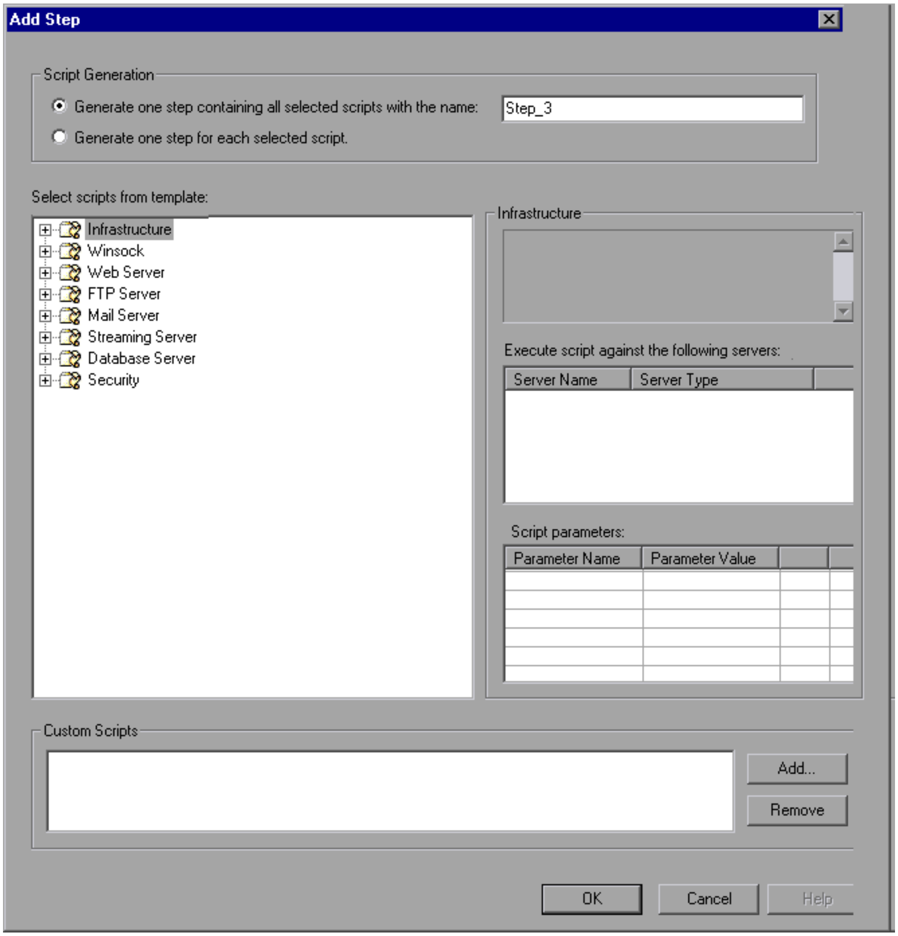
Adding Session Steps

Before creating session steps, you must create or load a topology (see Chapter 3, "Creating a Topology.") ProTune displays all of the relevant servers and session steps. When you click **OK** in the Topology window, ProTune opens the Session window's Design tab.

You can choose from the canned scripts or add custom scripts created with the Protune Virtual User Generator.

To create a new step:

1 In the Console window, click the **Add Step** button or choose **Session** > **Add Step**. The Add Step dialog box opens.



Add Step

Script Generation

● Generate one step containing all selected scripts with the name: Step_3

○ Generate one step for each selected script.

Select scripts from template:

- Infrastructure
- Winsock
- Web Server
- FTP Server
- Mail Server
- Streaming Server
- Database Server
- Security

Infrastructure

Execute script against the following servers:

Server Name	Server Type

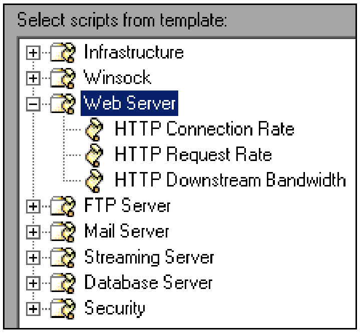
Script parameters:

Parameter Name	Parameter Value

Custom Scripts

2 You now specify whether you want to create a separate step for each selected script, or to include multiple scripts in one step. Click the appropriate radio button in the Step Generation section at the top of the window. If you are creating a step with multiple scripts, enter the step name in the text box, or accept ProTune's default. If you choose to create separate steps for each script, ProTune assigns a name to each step.

3 The left pane lists the types of canned scripts. Clicking a type expands it and shows all of its scripts. For example, clicking Web Server shows you the following scripts: HTTP Connection Rate, HTTP Request Rate and HTTP Downstream Bandwidth.



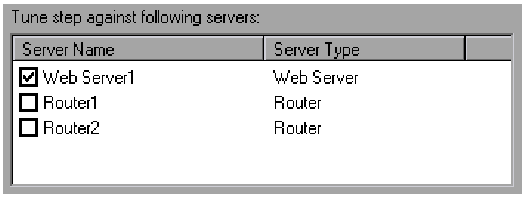
Select scripts from template:

- Infrastructure
- Winsock
- Web Server
  - HTTP Connection Rate
  - HTTP Request Rate
  - HTTP Downstream Bandwidth
- FTP Server
- Mail Server
- Streaming Server
- Database Server
- Security

Expand the script type that you want to use (if it is not already expanded).

4 Click the script that you want to test. Note that ProTune displays information about the script in the upper section of the window's right side.

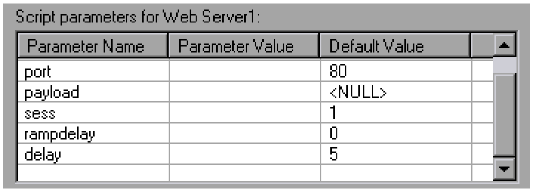
5 In the middle section of the right pane, ProTune displays the servers against which the script can be run. To specify the servers against which to run the script, check the boxes adjacent to those servers.



Tune step against following servers:

Server Name	Server Type
☑ Web Server1	Web Server
□ Router1	Router
□ Router2	Router

6 In the Script Parameters pane, ProTune displays a table containing the arguments for the selected script and their default values.

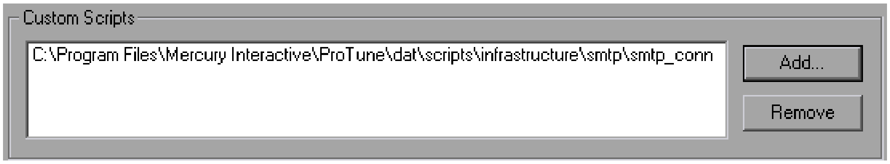


Script parameters for Web Server1:

Parameter Name	Parameter Value	Default Value
port		80
payload		<NULL>
sess		1
rampdelay		0
delay		5

To specify a different value, click the appropriate cell in the **Parameter Value** column and enter the value.

7 To add a custom script to a test, click the **Add** button in the lower right section of the window, browse to the script that you want to use, and click **Open**. The script name is added to the Custom Scripts pane.



C:\Program Files\Mercury Interactive\ProTune\dat\scripts\infrastructure\smtp\smtp_conn

8 Repeat steps through 7 for all the scripts and servers that you want to add to your session.

9 Click **OK**.

In the left pane, ProTune displays all the steps you defined. It displays separate steps for each server—if the same step is to be run against three servers, the step is displayed three times. You can see which scripts are included in each step (both canned and custom scripts).

Managing Steps

After you add steps to your session, you can manage them in several ways. The step commands are available from the right-click menu and the toolbar.

Add a step: Select a step and choose **Add Step** from the right-click menu.

Delete a step: Select a step and choose **Delete Step** from the right-click menu.

Rename a step: Select a step and choose **Rename Step** from the right-click menu to enable editing of the step name. Specify a new name for the step.

Duplicate a step: To place a copy of a step directly below it, select it and choose **Duplicate Step** from the right-click menu.

Move a step up: To move a step up in the list, select it and choose **Step Up** from the right-click menu.

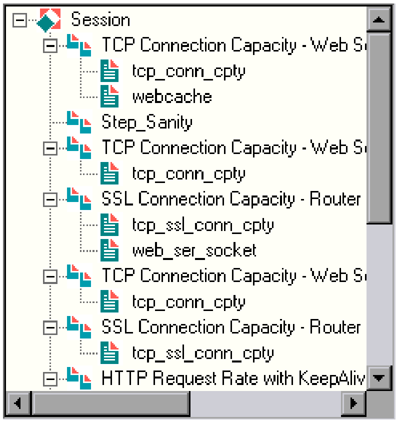
Move a step down: To move a step down in the list, select it and choose **Step Down** from the right-click menu.

Disable a step: To disable a step from the tuning session, select it and choose **Disable Step** from the right-click menu.

Enable a step: To enable a step that was disabled, select the step and choose **Enable Step** from the right-click menu.

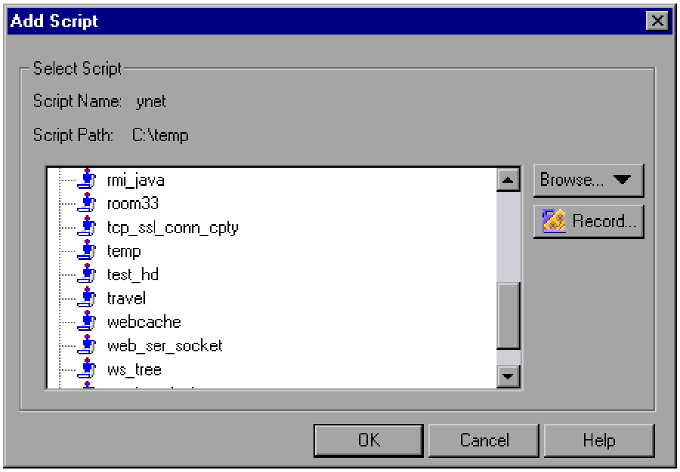
Managing Scripts

After you add scripts to your steps, you can add, delete, or view them from the step tree in the left pane of the Session window.



- Session
  - TCP Connection Capacity - Web S
    - tcp_conn_cpty
    - webcache
  - Step_Sanity
  - TCP Connection Capacity - Web S
    - tcp_conn_cpty
  - SSL Connection Capacity - Router
    - tcp_ssl_conn_cpty
    - web_ser_socket
  - TCP Connection Capacity - Web S
    - tcp_conn_cpty
  - SSL Connection Capacity - Router
    - tcp_ssl_conn_cpty
  - HTTP Request Rate with KeepAliv

Add a script: Select the step to which you want to add the script, and choose **Add Script** from the right-click menu or click the **Add Script** button. The Add script dialog box opens.



Script Name: ynet

Script Path: C:\temp

- rmi_java
- room33
- tcp_ssl_conn_cpty
- temp
- test_hd
- travel
- webcache
- web_ser_socket
- ws_tree

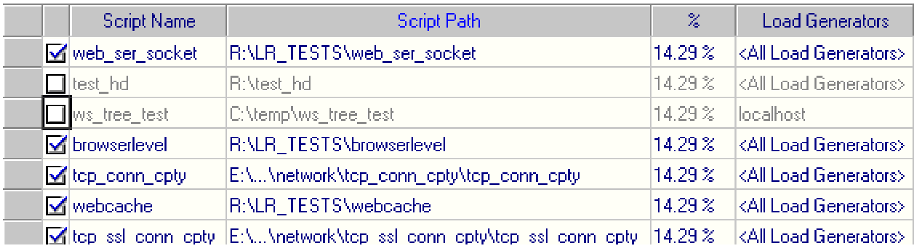
To add an existing script, select one of the displayed scripts and click **OK.** To change the path, click **Browse** and select an alternate path. To record a new script in the ProTune Virtual User Generator, click **Record**. For more information about recording scripts, see the *ProTune Creating Virtual User Scripts* guide.

Delete a script: Select the script you want to delete, and choose **Delete Script** from the right-click menu.

View or Modify a script: To view or modify a script, select it and click the **Modify the script** button.

Working with the Script List Window

The Script List shows all the scripts assigned to the current step.



		Script Name	Script Path	%	Load Generators
	☑	web_ser_socket	R:\LR_TESTS\web_ser_socket	14.29 %	<All Load Generators>
	□	test_hd	R:\test_hd	14.29 %	<All Load Generators>
	□	ws_tree_test	C:\temp\ws_tree_test	14.29 %	localhost
	☑	browserlevel	R:\LR_TESTS\browserlevel	14.29 %	<All Load Generators>
	☑	tcp_conn_cpty	E:\...\network\tcp_conn_cpty\tcp_conn_cpty	14.29 %	<All Load Generators>
	☑	webcache	R:\LR_TESTS\webcache	14.29 %	<All Load Generators>
	☑	tcp_ssl_conn_cpty	E:\...\network\tcp_ssl_conn_cpty\tcp_ssl_conn_cpty	14.29 %	<All Load Generators>

You can further manage your scripts from this window:

Sort the scripts: To sort the scripts by their name, path, percentage or Load Generator, click the title of the desired column.

Enable a Script: Select the check box adjacent to the script name. All selected scripts will be executed during the tuning session.

Disable a Script: Clear the check box adjacent to the script name. All disabled scripts will be ignored during the tuning session.

Rename a Script: Double-click on an item in the **Script Name** column and enter the desired name.

Add a Script to the List: Click in the **Script Name** column in the next available row, and click on the arrow to the right of the box. A list box opens showing all scripts in the most recent path. To add a script from the list, select it and click **OK.** To change the path, click **Browse** and select an alternate path.

Modify the Vuser Percentage: When you assign multiple scripts to a step, ProTune runs them simultaneously. By default ProTune distributes the scripts evenly between the Vusers. For example, if you assign two scripts to one step, fifty percent of the Vusers run one script, while the remaining fifty percent run the second script. To modify the script distribution, click in the **%** column, and modify the percentages accordingly. Note that the total sum or the percentages must equal 100.

To Add a Load Generator:

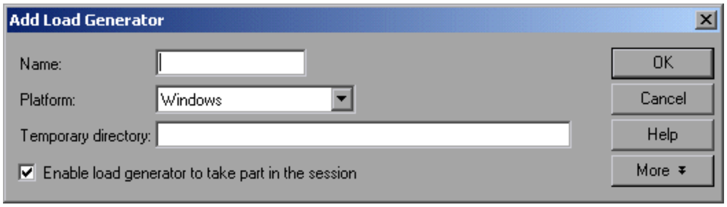
1 The Load Generators column automatically contains <All Load Generators> for each script. You can assign specific load generators for each script. Click in the **Load Generators** column in the next available row.

2 Click on the arrow to the right of the box. A list box opens showing the available load generator machines.

3 Select one or more machines and click **OK**.

4 Click **All Generators** to instruct ProTune to run the script on all available machines.

5 To add a new load generator, click the first entry in the list box, **Add**. The Add Load Generator dialog box opens:



Name:

Platform: Windows

Temporary directory:

☑ Enable load generator to take part in the session

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

7 In the **Platform** box, select the type of platform on which the load generator is running.

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

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

10 Click **OK** to close the Add Load Generator dialog box. ProTune adds the new load generator to the Load Generator Name list.

To include the new load generator in your session, select it from the Load Generator Name list, and click **OK**. Note that you can select multiple load generators.

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

For more information about setting up load generators, see Chapter 5, "Managing Load Generators."

To configure a load generator:

- Use the Load Generators dialog box to set a load generator's attributes while adding it to the load generator list, or to modify the attributes of an existing load generator at any time.
- You can also use the Load Generators dialog box to indicate which load generators will run Vusers in the session step. For example, if a load generator is unavailable for a particular session step, you can use the Load Generators dialog box to exclude it temporarily instead of removing it entirely from your list of load generators. For instructions on using the Load Generators dialog box, see "Configuring Load Generators" on page 57. To configure additional load generator settings, see "Configuring Load Generator Settings" on page 61.
- To configure global settings for all load generators participating in the session, use ProTune's Options dialog box. For more information, see Chapter 6, "Configuring Session Steps."

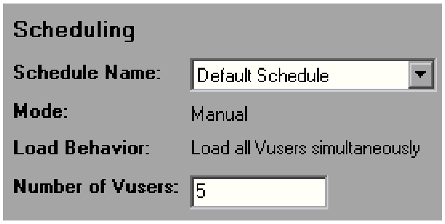
Setting an Initial Load (Manual Profiles)

If the profile associated with a step is a manual one, you can specify the number of Vusers to run on your system in order to test its connection capacity.

To define an initial load:

1. Select a session step in the left pane. The Description section displays a description of the script, and the Scheduling section shows the following fields:
   - Schedule Name
   - Mode (Manual or Goal Oriented)
   - Load behavior

➤ Number of Vusers



Scheduling

Schedule Name: Default Schedule

Mode: Manual

Load Behavior: Load all Vusers simultaneously

Number of Vusers: 5

2 In the Number of Vusers section, enter the number of Vusers to run during the tuning session.

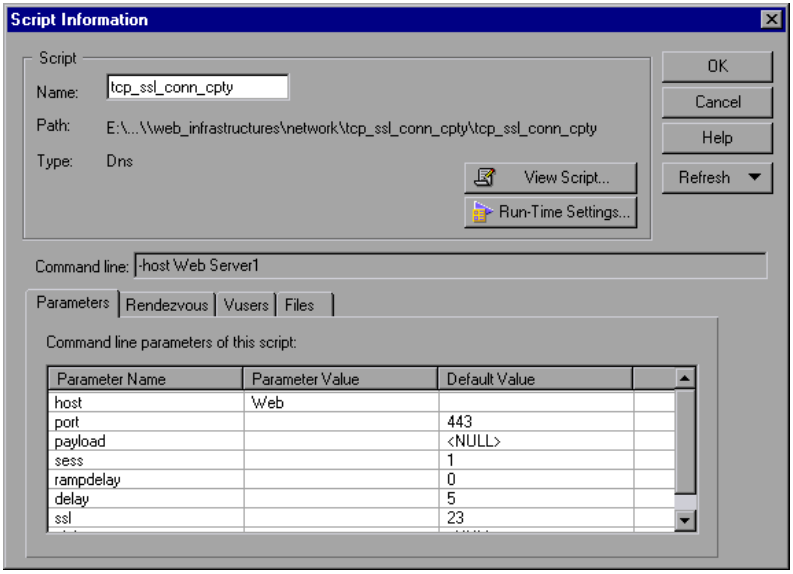
Note that in the **Execute** tab you can modify this value during the actual tuning. The value in this tab is useful for automatic scheduling—ProTune runs the session steps at the designated times with the number of Vusers specified in the **Total Number of Vusers to Run in Step** box.

Configuring Script Details

For scripts that are displayed in your list, you can view the details of the script you selected, edit the script, or change its run-time settings.

To view script details:

1 Select a script and click the **Details button** on the toolbar or double-click on the script in the left pane. The Script Information dialog box opens, displaying the Path, Name, and Type of the selected script. It also displays the script's command line options (for example -url Application Server1), and tabs displaying information about parameters, rendezvous points, Vusers and files.



Name: tcp_ssl_conn_cpty

Path: E:\...\web_infrastructures\network\tcp_ssl_conn_cpty\tcp_ssl_conn_cpty

Type: Dns

Command line: -host Web Server1

Command line parameters of this script:

Parameter Name	Parameter Value	Default Value
host	Web	
port		443
payload		<NULL>
sess		1
rampdelay		0
delay		5
ssl		23

2 Click **Run-Time Settings** to set the script’s run-time settings (optional), which allow you to customize the way the Console executes a script. The Run-Time Settings dialog box opens, displaying the settings you previously set using VuGen. If you did not set run-time settings for a script in VuGen, the default VuGen settings are displayed for all but the Log and Think Time tabs, which display the default Console settings. Note that several protocols, such as Web and Java, have specific settings.

For information on configuring the run-time settings, refer to the *ProTune Creating Virtual User Scripts* guide.

Note: If you modify the run-time settings from the Console, ProTune runs the script using the modified settings. To restore the initial settings, click the **Refresh** button and select **Run-Time Settings**.

---

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

Note: If you use VuGen to make changes to a script while the Console is running, click the **Refresh** button and select **Script** to update the script details in the session step.

---

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

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

5 Click the **Parameters** tab to view the arguments for the selected script and their default values. To specify your own value, click in the **Parameter Value** column and enter the desired value.

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

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

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

9 Click **Refresh** > **Script** to refresh the script and use the default setting in the tabs (File tab). Click **Refresh** > **Runtime settings** to use the script's default run-time settings.

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

After you add steps and scripts to your session, you can configure run-time options and set a schedule. Refer to "Getting Started with Designing a Session," on page 36 for a summary of the testing procedure. After you set up your session steps, you run the steps and begin your tuning session. For information on running the tuning session, see Chapter 10, "Running a Session."

Using Relative Paths for Scripts

To specify the location of a script, you can either browse to the script or type its relative location into the Script Path column. The location can be relative to the current session directory, or the ProTune installation directory.

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

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

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

For example, if the current session is located at F:\sessions, to specify a script located at F:\sessions\scripts\user.usr, you could type \scripts\user1.usr.:

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

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

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

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

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

You can instruct the Console to store the script on a shared network drive (see Chapter 6, "Configuring Session Steps.") If you configure the Console to save the script to a network drive, you must ensure that the Vuser load generator recognizes the drive. The Script window contains a list of all the scripts and their paths. A script's path is based on the Console load generator's mapping of that location. If a Vuser load generator maps to the script's path differently, path translation is required. Path translation converts the Console load generator's mapping to the Vuser load generator's mapping. For more information see Appendix B, "Performing Path Translation."

5

Managing Load Generators

After choosing the tests you want to run, and the components that you want to test, you need to specify the computers from which ProTune will run the tests. These computers are called load generators. This chapter describes how to define and manage load generators.

This chapter discusses:

➤ Configuring Load Generators

➤ Configuring Load Generator Settings

About Managing Load Generators

You use load generator machines for running the tests on your components. When you assign a script to a step, you also specify the load generators that will run the script, and can configure their properties.

Configuring Load Generators

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

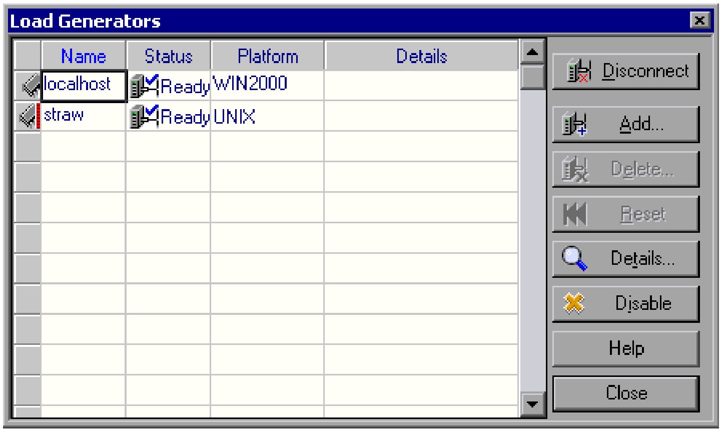
To configure global settings for all load generators participating in the session, use the **Timeout** tab in the **Tools** > **Options** dialog box. For more information, see Chapter 6, "Configuring Session Steps." To set properties specific to each load generator, use the Load Generators dialog box as described below.

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

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

To configure a load generator:

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



Load Generators

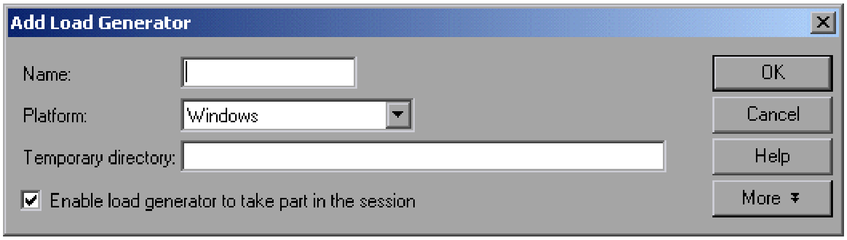
Name	Status	Platform	Details
localhost	Ready	WIN2000	
straw	Ready	UNIX	

2 Click **Connect** to change the Status of the load generator from DOWN to READY. Click **Disconnect** to change the Status of the load generator from READY to DOWN.

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

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

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



Add Load Generator

Name:

Platform: Windows

Temporary directory:

☑ Enable load generator to take part in the session

Type the name of the load generator in the **Name** box. In the Platform box, select the type of platform on which the load generator is running.

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

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

8 Click **More** to expand the dialog box and show the following additional tabs where you can configure load generator settings:

- **Status**
- **Run-Time Quota**
- **Firewall**
- **Run-Time File Storage**
- **Unix Environment**
- **Vuser Limits**

Note: For information on configuring these settings, see "Configuring Load Generator Settings" on page 61.

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

10 To remove a load generator, select it and click **Delete**.

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

Note: The Protune Console monitors a Windows load generator machine's CPU usage and automatically stops loading Vusers on a load generator when it becomes overloaded. You can monitor the status of a machine's CPU usage using the icons in the Load Generators dialog box. When the CPU usage of a load generator becomes problematic, the icon to the left of the load generator name contains a yellow bar. When the machine becomes overloaded, the icon contains a red bar.

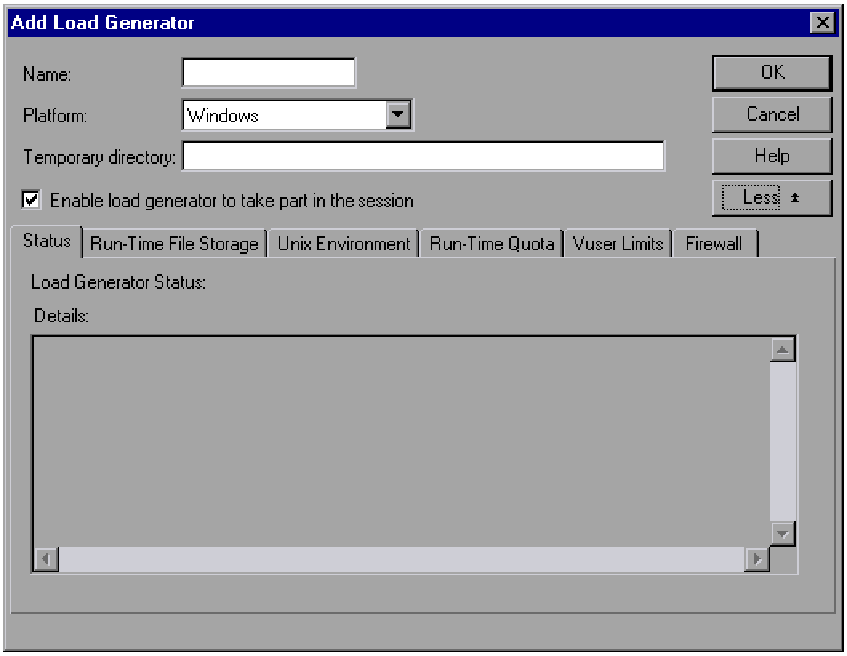
Configuring Load Generator Settings

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

You can configure global settings for all load generators participating in the session, using the Options dialog box. For more information, see Chapter 6, "Configuring Session Steps."

To configure load generator settings:

1 From the Add Load Generator or Load Generator Information dialog box, click **More** to expand the box and show the Status, Run-Time File Storage, UNIX Environment, Run-Time Quota, Vuser Limits, and Firewall (when the load generator is not the localhost) tabs.



Add Load Generator

Name:

Platform: Windows

Temporary directory:

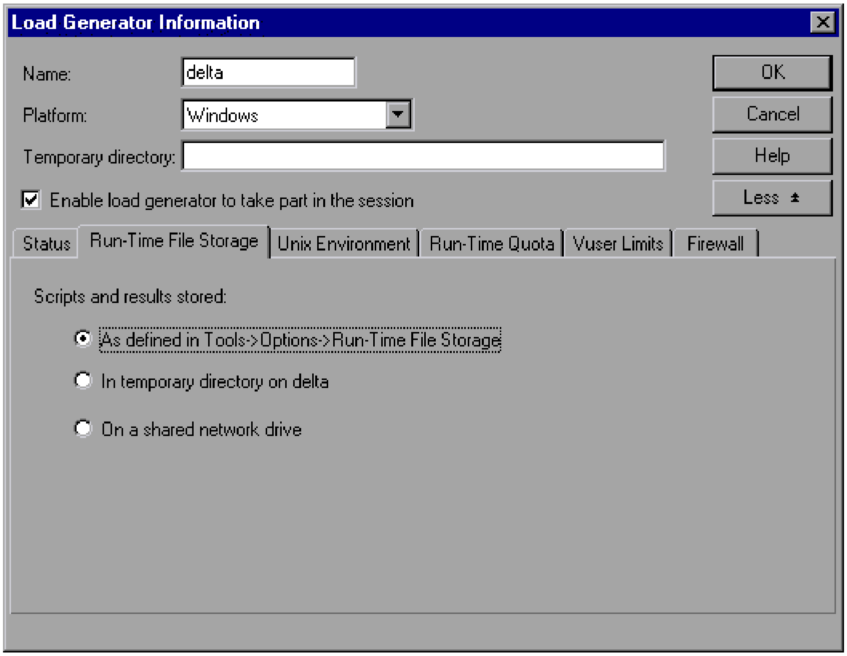
☑ Enable load generator to take part in the session

Load Generator Status:

Details:

2 Select the **Status** tab to display the Load Generator's status.

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



Load Generator Information

Name: delta

Platform: Windows

Temporary directory:

☑ Enable load generator to take part in the session

Scripts and results stored:

● As defined in Tools->Options->Run-Time File Storage

○ In temporary directory on delta

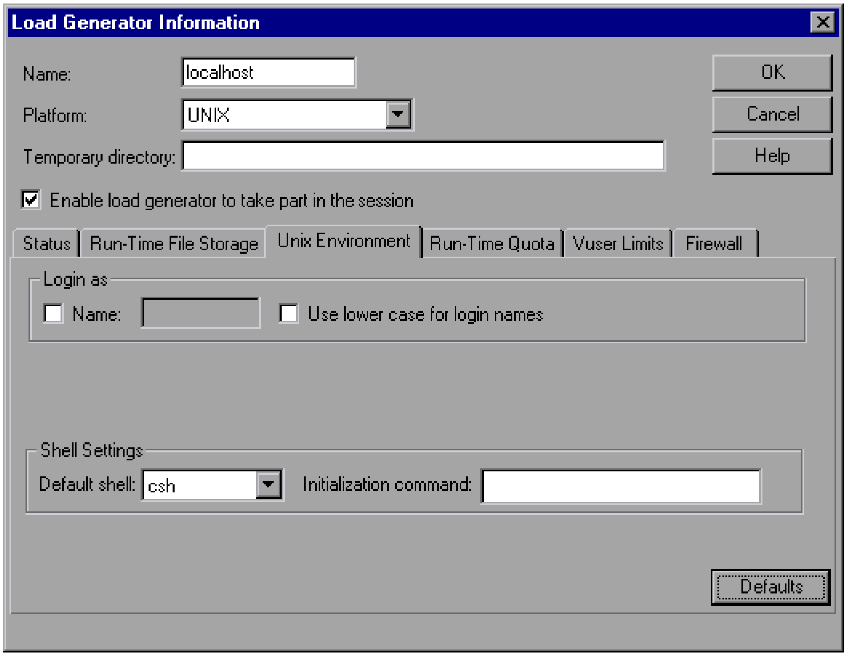
○ On a shared network drive

To store the results as specified in the global settings, click **As defined in Tools** > **Options** > **Run-Time File Storage.** To store the results temporarily on a hard drive of the load generator computer, click **In temporary directory on** <***load generator name***>. To store the session scripts or results on a shared network drive, click **On a shared network drive**. To set the network location for the results, see Chapter 9, "Preparing to Run a Session Step."

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

---

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



Load Generator Information

Name: localhost

Platform: UNIX

Temporary directory:

☑ Enable load generator to take part in the session

Login as

☐ Name: ☐ Use lower case for login names

Shell Settings

Default shell: csh Initialization command:

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

Note: For information on the Local User setting available in Expert mode, see "Working in Expert Mode" on page 515.

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

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

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

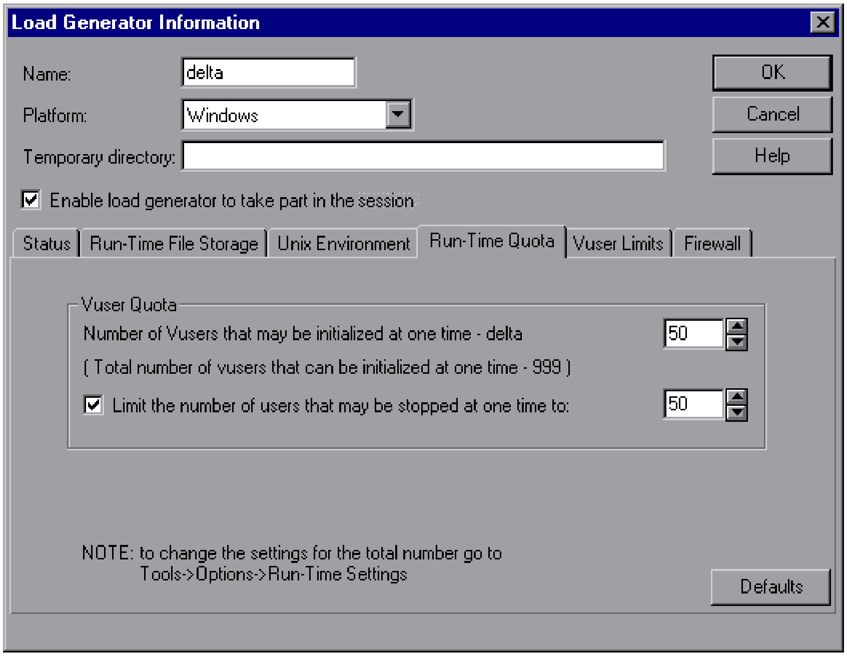
Note: If you are using a Korn shell (ksh), you must delete all ProTune settings from the *.cshrc* file (e.g. M_LROOT) before executing the session.

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

For example, you could select *ksh* and use the following initialization command:

```
. .profile;
```

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



Load Generator Information

Name: delta

Platform: Windows

Temporary directory:

☑ Enable load generator to take part in the session

Vuser Quota

Number of Vusers that may be initialized at one time - delta: 50

(Total number of vusers that can be initialized at one time - 999)

☑ Limit the number of users that may be stopped at one time to: 50

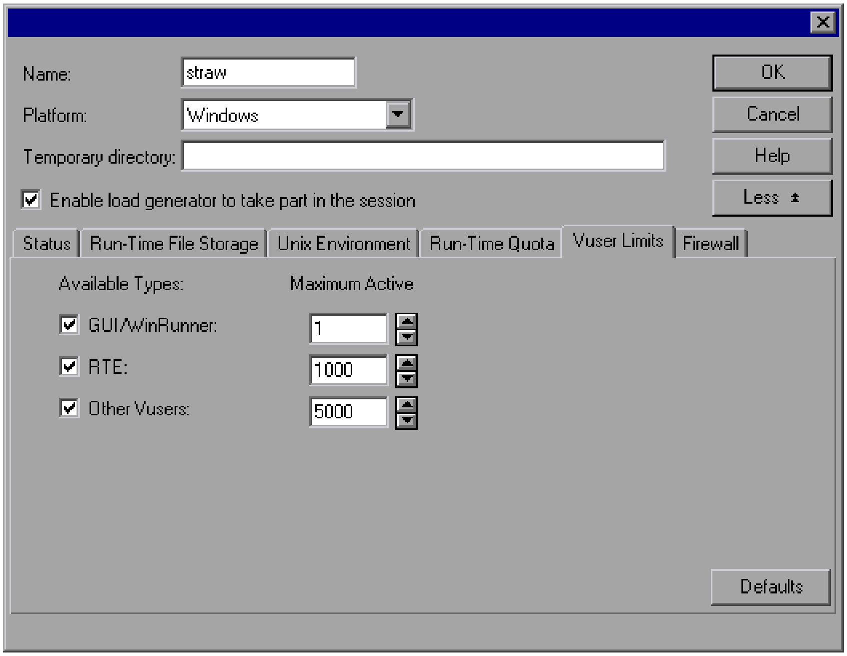
NOTE: to change the settings for the total number go to
Tools->Options->Run-Time Settings

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

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

You can set run-time quotas for an entire session using the Run-Time Settings tab in the Options dialog box. For information on setting quotas globally for an entire session, see Chapter 6, "Configuring Session Steps."

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



Name: straw

Platform: Windows

Temporary directory:

☑ Enable load generator to take part in the session

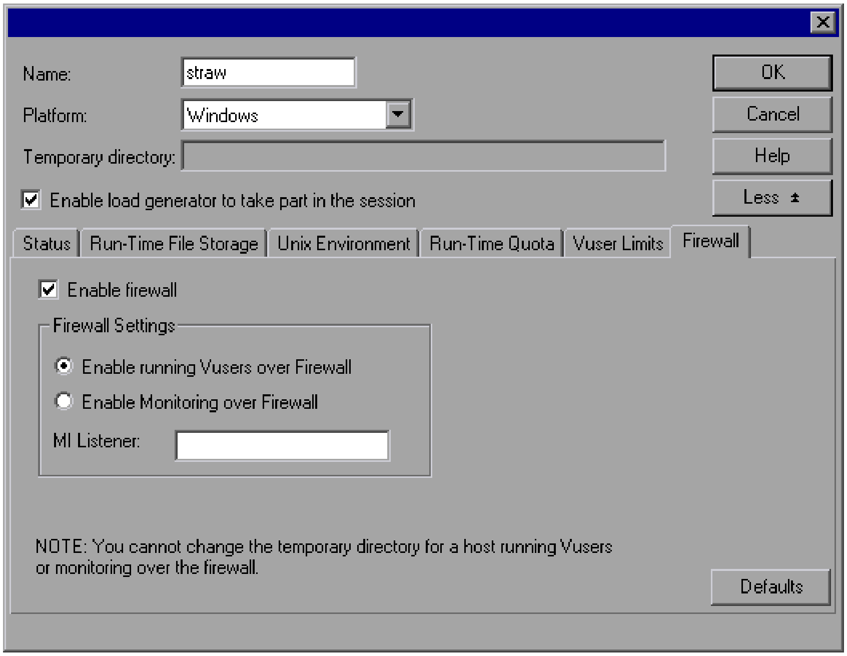
Available Types:	Maximum Active
☑ GUI/WinRunner:	1
☑ RTE:	1000
☑ Other Vusers:	5000

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

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

---

7 Select the **Firewall** tab to enable monitoring or running Vusers through a firewall.



Name: straw

Platform: Windows

Temporary directory:

☑ Enable load generator to take part in the session

☑ Enable firewall

Firewall Settings

● Enable running Vusers over Firewall

○ Enable Monitoring over Firewall

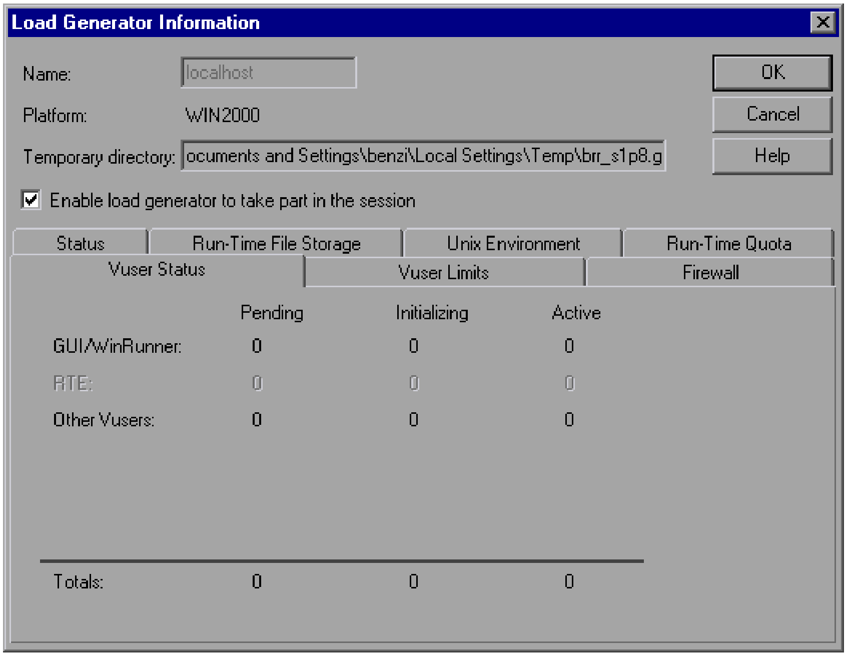
MI Listener:

NOTE: You cannot change the temporary directory for a host running Vusers or monitoring over the firewall.

You can enable or disable ProTune's firewall functionality, specify the name of the MI Listener the load generator is using, and click either the **Enable Monitoring over Firewall** radio button, or the **Enable running Vusers over Firewall** radio button.

Note: If the load generator is connected, you cannot change values in the **Firewall** tab, or if you change the name of the load generator to the name of the local host, you cannot set any values in the **Firewall** tab.

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



Load Generator Information

Name: localhost

Platform: WIN2000

Temporary directory: ocuments and Settings\benzi\Local Settings\Temp\brr_s1p8.g

☑ Enable load generator to take part in the session

	Pending	Initializing	Active
GUI/WinRunner:	0	0	0
RTE:	0	0	0
Other Vusers:	0	0	0
Totals:	0	0	0

Note: For information on the Connection Log tab available in Expert mode, see “Working in Expert Mode” on page 515.

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

6

Configuring Session Steps

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

This chapter describes:

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

About Configuring a Session

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

The settings described in this chapter apply to all the load generators in a session. To change the settings for individual load generator machines, refer to Chapter 3, “Creating a Topology.” If the global session settings differ from those of an individual load generator, the load generator settings override them.

The settings discussed in this chapter are unrelated to the Vuser run-time settings. These settings, which apply to individual Vusers or scripts, contain information about logging, think time, and the network, the number of iterations, and the browser. For information on setting the run-time settings, see the *ProTune Creating Virtual User Scripts* guide.

For information on setting the options for online monitors, see Chapter 13, "Online Monitoring."

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

Configuring Session Run-Time Settings

The session run-time settings relate to:

- Vuser Quotas
- Stopping Vusers
- Random Sequence Seed

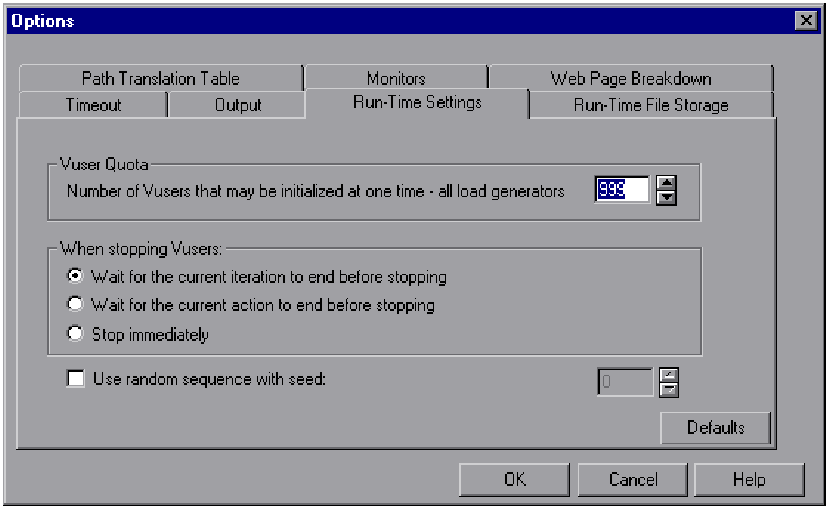
Vuser quotas: To prevent your system from overloading, you can set quotas for Vuser activity. The Vuser quotas apply to Vusers on all load generators. You can limit the number of Vusers initialized at one time (when you send an Initialize command).

Stopping Vusers: ProTune lets you control the way in which Vusers stop running when you click the Stop button. You can instruct ProTune to allow a Vuser to complete the iteration it is running before stopping, to complete the action it is running before stopping, or to stop running immediately.

Random sequence seed: ProTune lets you set a seed number for random sequencing. Each seed value represents one sequence of random values used for test execution. Whenever you use this seed value, the same sequence of values is assigned to the Vusers in the session. This setting applies to parameterized scripts using the Random method for assigning values from a data file. Enable this option if you discover a problem in the test execution and want to repeat the test using the same sequence of random values.

To set the session run-time settings:

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



Vuser Quota

Number of Vusers that may be initialized at one time - all load generators 999

When stopping Vusers:

● Wait for the current iteration to end before stopping

○ Wait for the current action to end before stopping

○ Stop immediately

☐ Use random sequence with seed: 0

2. To set a Vuser quota, specify the desired value.
3. Select the way in which you want ProTune to stop running Vusers.
4. To specify a seed value for a random sequence, select the **Use random sequence with seed** check box and enter the desired seed value.

Saving Messages to the Output Files

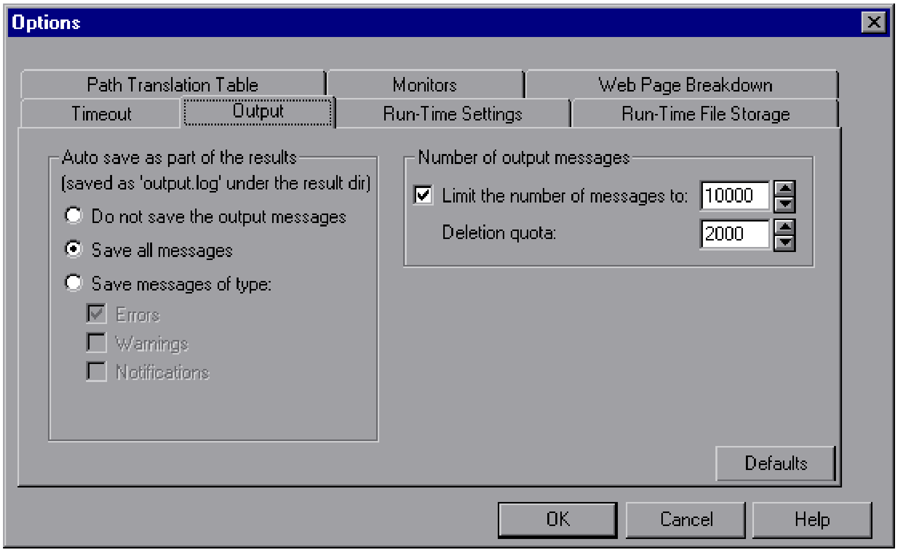
By default, ProTune generates an output file, *output.log*, and stores it in the script directory. This file contains error, warning, and notification messages issued by the Console during session execution. Using the Options dialog box, you can select the type of messages to save to the output file.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

You can specify the number of messages that will appear in the output window. If the number of messages exceeds the limit, the messages are deleted. A deletion quota value specifies the number of messages that may be deleted from the output. The deletion quota overrides the limit number.

For example, assume that you limit the number of messages to 500 and set the deletion quota to 50. If there are 600 messages, 550 will appear in the output.

To configure the output logging:

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



Auto save as part of the results
(saved as 'output.log' under the result dir)

○ Do not save the output messages
● Save all messages
○ Save messages of type:
☑ Errors
□ Warnings
□ Notifications

Number of output messages

☑ Limit the number of messages to: 10000
Deletion quota: 2000

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

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

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

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

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

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

Setting Timeout Intervals

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

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

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

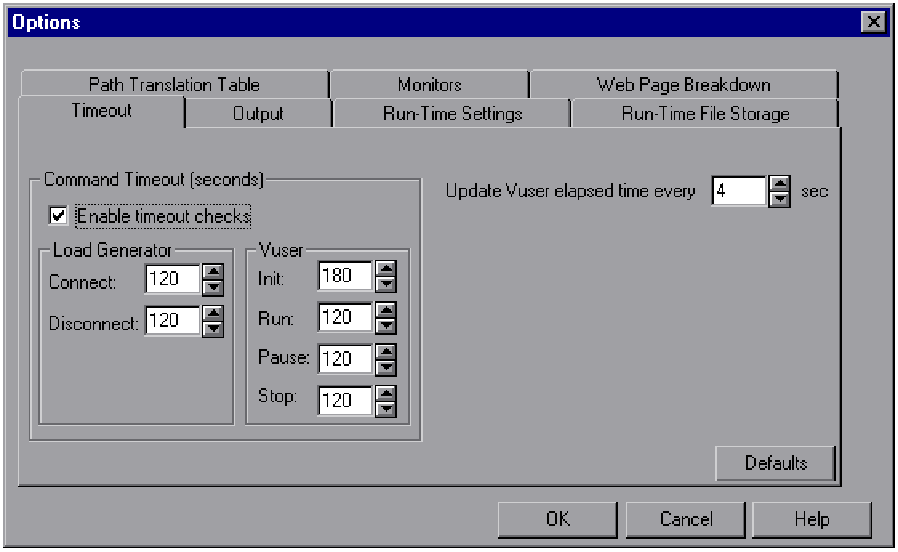
For example, the default *Init* timeout is 180 seconds. If you select a Vuser and click the **Initialize** button, ProTune checks whether the Vuser reaches the READY state within 180 seconds; if it does not, the Console issues a message indicating that the *Init* command timed out.

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

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

To set timeout intervals:

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



Command Timeout (seconds)

☑ Enable timeout checks

Load Generator

Connect: 120

Disconnect: 120

Vuser

Init: 180

Run: 120

Pause: 120

Stop: 120

Update Vuser elapsed time every 4 sec

2 Clear the **Enable timeout checks** check box to disable the timeout test. ProTune waits an unlimited time for the load generators to connect and disconnect, and for the Initialize, Run, Pause, and Stop commands to be executed.

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

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

Setting the Run-Time File Location

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

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

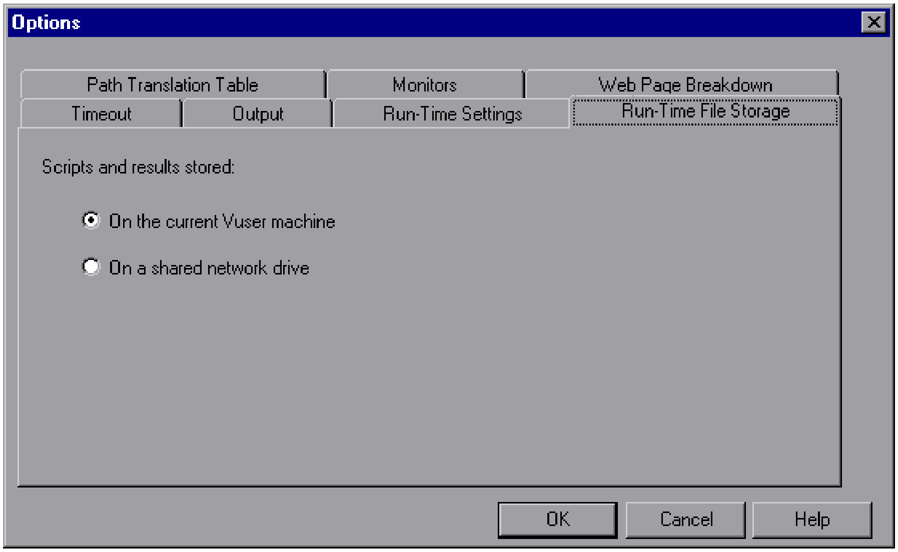
The primary run-time files are script and result files:

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

Result files: While you run a session, the participating Vusers write their results to the temporary run-time file directory. After session execution, these result files are collated or consolidated—results from all of the load generators are transferred to the results directory. You set the location of the results directory as described in Chapter 10, "Running a Session." After collating the results, the temporary run-time directory is deleted.

To specify where ProTune stores run-time files:

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



Scripts and results stored:

● On the current Vuser machine

○ On a shared network drive

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

2 To store script and result files on a shared network drive, click **On a shared network drive**. To set the exact location on the network drive, see Chapter 9, “Preparing to Run a Session Step.”

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

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

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

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

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

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

Specifying Path Translation

If you specified a shared network drive for run-time file storage, (see "Setting the Run-Time File Location" on page 75), you may need to perform *path translation*. Path translation is a mechanism used by ProTune to convert a remote path names. A typical session may contain several load generator machines that map the shared network drive differently. For more information, see the Appendix B, "Performing Path Translation."

7

Defining Alerts

Before tuning your session, you set up alerts that define what actions ProTune should take when server performance problems occur.

This chapter describes:

- ➤ Types of Alerts
- ➤ Specifying Alert Conditions
- ➤ Specifying Alert Actions
- ➤ Viewing Alert Descriptions
- ➤ Creating, Configuring, and Deleting Alerts
- ➤ Enabling and Disabling the Alert Mechanism
- ➤ Enabling and Disabling the Alert Mechanism
- ➤ Viewing the Alerts Output Window

About Defining Alert Schemes

ProTune uses alerts to let you know when server performance problems occur.

Before running scripts, you define alerts for one or more measurements. This includes specifying what conditions should trigger an alert, and the action that ProTune should take when the need to issue an alert is detected.

You use the Alerts window to specify alert conditions and an alert scheme. You specify separate alert conditions for each session step.

Types of Alerts

You can create alerts that will be triggered by specific values occurring in the following measurements:

Note: The following is a partial list of the measurements that you can use for triggering alerts. All measurements that are available in the Console can be used.

➤ **Running Vusers**

Informs you when the number of Running Vusers in the Running, Ready, Finished, or Error state, reaches a specific value.

➤ **Error Statistics**

Informs you when the number of errors reaches a specified number.

➤ **Vusers with Errors**

Informs you when the number of Vusers with errors reaches a specified number.

➤ **Transaction Response Time (Passed)**

Informs you when the transaction response time reaches a specified value, for the *vuser_int*, *Actions*, or *vuser_end* sections of the script.

➤ **Transactions Per Second (Passed)**

Informs you when the specified number of transactions per second is reached for the *vuser_int*, *Actions*, or *vuser_end* sections of the script.

➤ **Total Transactions Per Second (Passed)**

Informs you when the specified number of transactions per second is reached for the *vuser_int*, *Actions*, or *vuser_end* sections of the script.

➤ **Hits per Second**

Informs you when the specified number of hits per second is reached.

- **Throughput**

  Informs you when your server's throughput reaches a specific level.

- **Pages Downloaded per Second (Passed)**

  Informs you when a specific number of pages has been downloaded per second by the server.

Specifying Alert Conditions

ProTune allows you to specify when you want it to issue alerts for the measurement that you are monitoring. ProTune provides several types of alert schemes. The alert schemes instruct ProTune to issue an alert when it detects a specific value, a value for a specific duration, an out of range value, a change in value, or a standardized value.

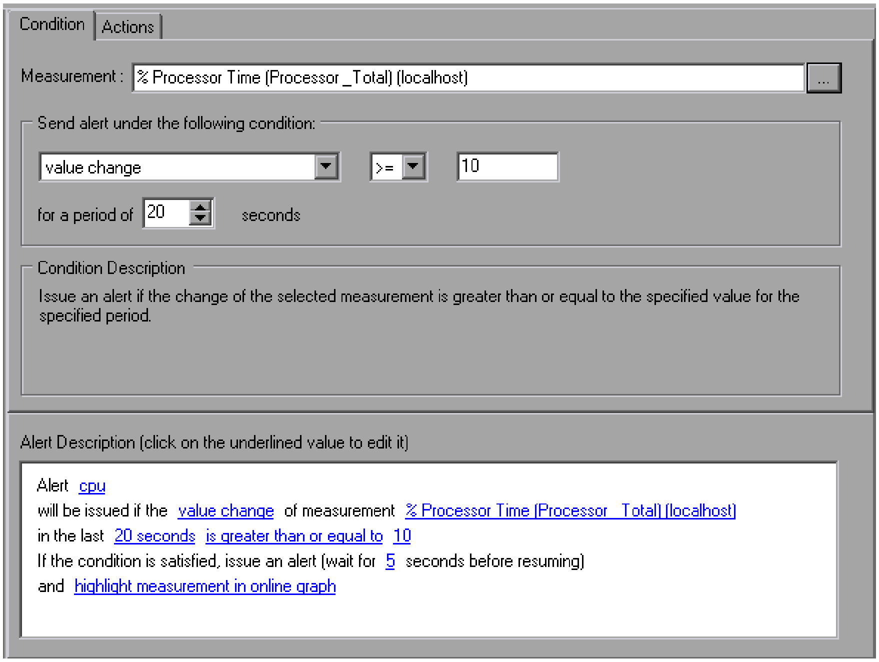
Scheme	Additional fields	Meaning
value	none	Issue an alert if the measurement compares with the specified value in one of the following ways: > >= < <= =
value for a duration of time	for a period of *n* seconds	Issue an alert if the selected measurement compares with the specified value in one of the following ways for *n* seconds: > >= < <= =
value out of range	range	Issue an alert if the measurement deviates from the specified range.

Scheme	Additional fields	Meaning
value change	for a period of *n* seconds	Issue an alert if the change in the selected measurement compares with the specified value in one of the following ways over the last *n* seconds: > >= < <= =
standardized value	for a period of *n* seconds	Issue an alert if the standardized value of the selected measurement compares with the specified value in one of the following ways, considering the period of the last *n* seconds: > >= < <= =

You can choose one of the following conditions for your alert scheme:

- > greater than
- >= greater than or equal to
- < less than
- <= less than or equal to
- = equal to

You define an alert by selecting a measurement, scheme, and condition in the Alerts Window's **Condition** tab. In the following example, ProTune is instructed to issue an alert when it detects a change of 10 or more in the value of the *% Processor Time* measurement, over a period of 20 seconds.



Measurement : % Processor Time (Processor _Total) (localhost)

Send alert under the following condition:

value change >= 10

for a period of 20 seconds

Condition Description

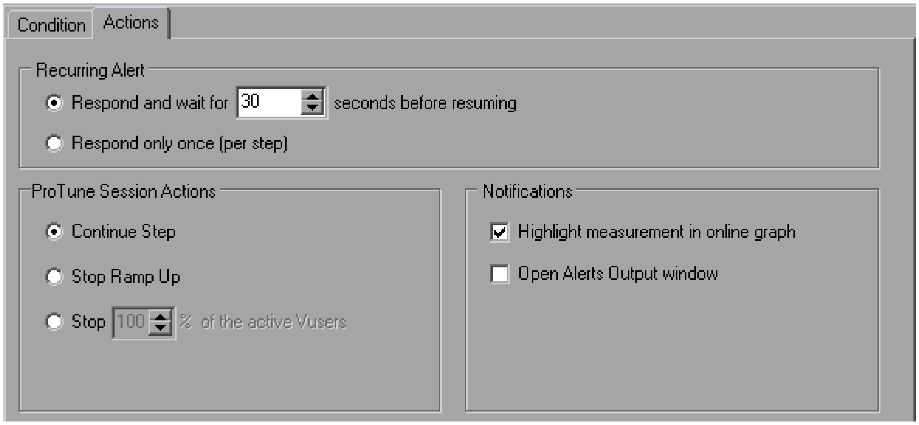
Issue an alert if the change of the selected measurement is greater than or equal to the specified value for the specified period.

Alert Description (click on the underlined value to edit it)

Alert cpu
will be issued if the value change of measurement % Processor Time (Processor _Total) (localhost)
in the last 20 seconds is greater than or equal to 10
If the condition is satisfied, issue an alert (wait for 5 seconds before resuming)
and highlight measurement in online graph

Specifying Alert Actions

The Alerts Window's Actions tab allows you to specify the actions ProTune takes when an alert is triggered.



Recurring Alert

● Respond and wait for 30 seconds before resuming

○ Respond only once (per step)

ProTune Session Actions

● Continue Step

○ Stop Ramp Up

○ Stop 100 % of the active Vusers

Notifications

☑ Highlight measurement in online graph

☐ Open Alerts Output window

You can specify settings for the following:

- Recurring Alerts
- Session Actions
- Notification Types

Recurring Alerts

The first notification setting relates to the frequency of responses for a recurring alert in a single step. You can instruct ProTune to respond in one of the following ways:

- **Respond and Wait *n* seconds before resuming**
- **Respond only once** (per ProTune step)

Default: respond and wait 30 seconds before resuming the test.

Session Actions

ProTune lets you specify what action to perform when encountering an alert condition:

- **Continue Step:** Continue executing the current step after the alert is triggered.
- **Stop Ramp Up:** Stop adding additional Vusers to the session step.
- **Stop a percentage of the active Vusers:** When an alert is triggered, stop a percentage of the active Vusers. You can set a percentage from 0 to 100.

Notification Types

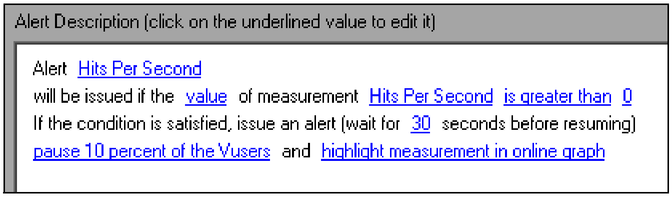
You can specify the type of notification to issue when an alert is triggered. The available notification types are:

- **Highlight measurement in online graph:** On the online graph, highlight the measurement that triggered the alert. This graph can be viewed on the Execute tab.
- **Open Alerts Output window:** Open the Alerts Output window when an alert trigger occurs. For more information about the Alerts Output Window, see the "Viewing the Alerts Output Window," on page 91.

You can enable either notification mechanism, both of them, or neither. Even if you do not enable a notification mechanism, you can still open the Alerts Output window to view the alerts that occurred.

Viewing Alert Descriptions

The Alert Description pane contains a description of the alert in plain language. It includes hyperlinks that help you to change the alert's condition and actions.



Alert Description (click on the underlined value to edit it)

Alert Hits Per Second
will be issued if the value of measurement Hits Per Second is greater than 0
If the condition is satisfied, issue an alert (wait for 30 seconds before resuming)
pause 10 percent of the Vusers and highlight measurement in online graph

When you click a hyperlink, ProTune positions the cursor at the field that you want to change. In some cases, clicking a hyperlink causes ProTune to open a dialog box so you can change the relevant value.

Creating, Configuring, and Deleting Alerts

After designing the session steps, you set alerts for your tuning session. The alerts provide real-time information about your server performance and inform you when specific thresholds are reached.

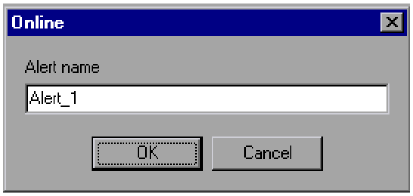
After creating an alert, you specify the alert conditions and actions.

Creating an Alert

Creating an alert includes specifying the alert name and choosing the measurement that will trigger the alert.

To create an alert:

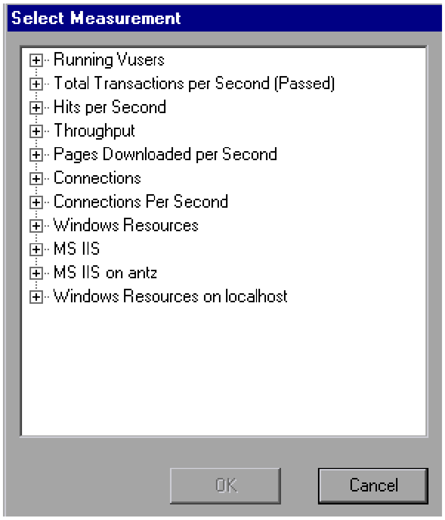
1. Click the **Alerts Definition** button in the ProTune Console window. The Alerts window opens.
2. Click **Add** in the bottom left corner. The Online dialog box opens.



Online

Alert name: Alert_1

3 Enter a name for the alert and click **OK**. The Select Measurement dialog box opens, displaying categories of measurements that you can monitor.

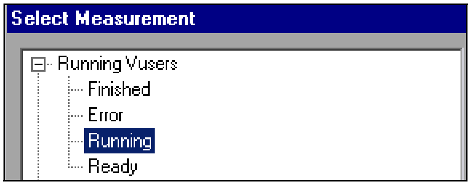


The following categories always appear, regardless of your topology and monitors:

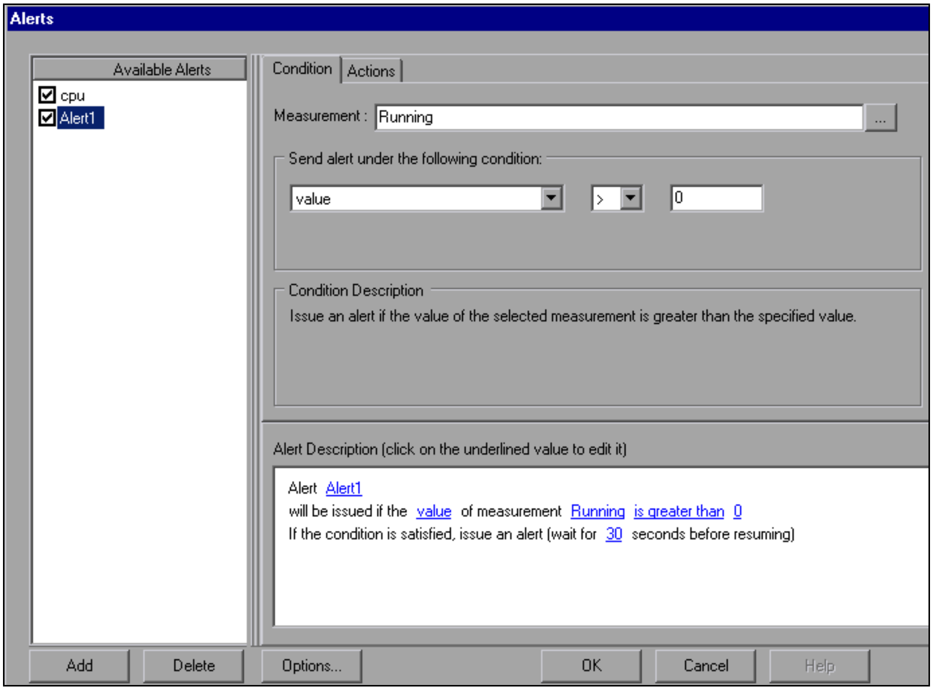
- Running Vusers
- Total Transactions per Second (Passed)
- Hits per Second
- Throughput
- Pages Downloaded per Second
- Connections
- Connections per second

Additional categories may appear, depending on your topology and the monitors you assigned to various elements.

4 Expand the required category to display its measurements.



5 Select the desired measurement and click **OK**. ProTune creates an alert with default settings. The alert name you supplied appears in the Available Alerts pane, the Condition and Actions tabs show the default values, and the Alert Description pane displays a plain language description of the alert.



Alerts

Available Alerts

☑ cpu
☑ Alert1

Condition | Actions

Measurement : Running

Send alert under the following condition:

value > 0

Condition Description

Issue an alert if the value of the selected measurement is greater than the specified value.

Alert Description (click on the underlined value to edit it)

Alert Alert1
will be issued if the value of measurement Running is greater than 0
If the condition is satisfied, issue an alert (wait for 30 seconds before resuming)

6 To delete an alert, select it in the Available Alerts pane and click **Delete**.

Specifying Alert Conditions

You can change the existing settings to suit your needs.

To specify alert conditions:

1 In the Condition tab, choose an alert scheme in the **Send alert under the following condition** box.

2 Choose an operator: >, >=, <, <=, or =.

3 Specify a condition:

➤ For the **value** scheme, specify a trigger value.

- For the **value for a duration of time** scheme, specify a trigger value and a duration.
- For the **value out of range** scheme, specify start and end range trigger values.
- For the **value change** scheme, specify a trigger value and a watch time—the alert is only triggered if the condition is reached within the watch time.
- For the **standardized value** scheme, specify a trigger value and a watch time—the alert is only triggered if the condition is reached within the watch time.

Note: You can use the Alert Description pane to change alert condition settings (see “Viewing Alert Descriptions,” on page 85).

Specifying Alert Actions

ProTune also assigns default settings for the actions that are triggered by an alert. ProTune displays these settings in the Actions tab. You can change the default settings to suit your needs.

The Actions tab lets you specify the following:

- Recurring alert behavior
- ProTune session actions
- Notifications

To specify alert actions:

1. In the **Recurring Alert** section, select one of the following response schemes:
   - **Respond and wait for *n* seconds before resuming**—specify a value for *n*
   - **Respond only once** (per ProTune step)
2. In the **ProTune Session Actions** section, choose one of the following actions:
   - **Continue current step**
   - **Stop Ramp Up**

- **Stop *n* % of active Vusers**—specify a value for *n*. The default is 100%

3 In the **Notifications** section, select the desired notification method(s):

- **Highlight measurement in online graph**
- **Open Alerts Output window**

Note: You can use the Alert Description pane to change alert action settings (see "Viewing Alert Descriptions," on page 85).

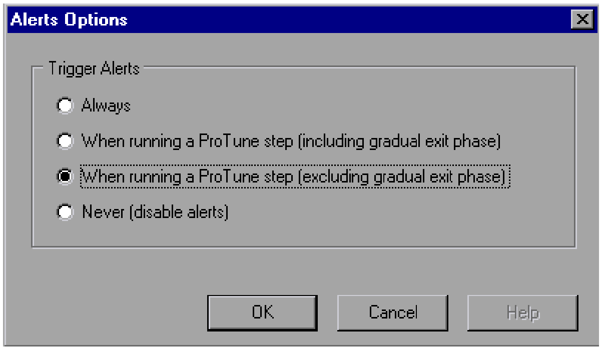
---

4 To set the alert triggers or enable/disable alerts, click **Options**. The Alerts options dialog box opens. Select a trigger option and click **OK**. For more information, see the section "Enabling and Disabling the Alert Mechanism," on page 90.

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

Enabling and Disabling the Alert Mechanism

You can enable, disable, or limit ProTune alerts, using the Alert Options dialog box.



Alerts Options

Trigger Alerts

○ Always

○ When running a ProTune step (including gradual exit phase)

● When running a ProTune step (excluding gradual exit phase)

○ Never (disable alerts)

You can choose one of the following trigger options:

Always: Always issue alerts when the alert condition is met.

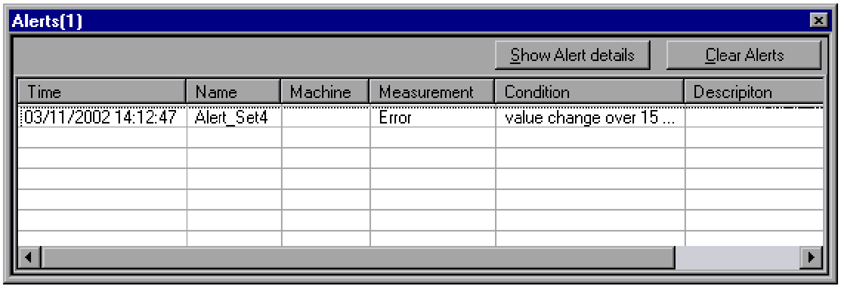
When Running a ProTune step (including gradual exit): Issue alerts only during test execution, even during the Vuser's gradual exit stage.

When Running a ProTune step (excluding gradual exit phase): Issue alerts only during test execution, except during the Vuser's gradual exit stage. This is the default option.

Never: Never issue alerts. This disables the alert mechanism for the active session.

Viewing the Alerts Output Window

The Alerts Output window in the Execute tab contains information about all the alerts triggered during a tuning session.



Time	Name	Machine	Measurement	Condition	Descripiton
03/11/2002 14:12:47	Alert_Set4		Error	value change over 15 ...	

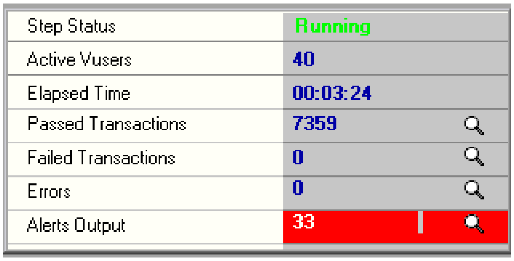
The Alerts Output window displays the following information about each alert:

- **Time:** The date and time when the alert was triggered.
- **Name:** The name of the Alert as defined by the user.
- **Machine:** The machine that triggered the alert.
- **Measurement:** The measurement for which the alert was triggered.
- **Condition:** The alert condition.
- **Description:** A user-supplied description of the alert.

To open the Alerts Output window:

1 Click the **Execute** tab.

2 In the statistics table (the middle part of the upper section) locate the Alerts Output row.



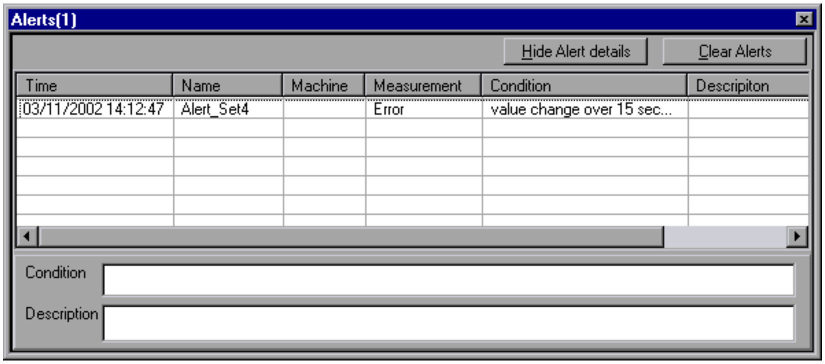
Step Status	Running	
Active Vusers	40	
Elapsed Time	00:03:24	
Passed Transactions	7359	🔍
Failed Transactions	0	🔍
Errors	0	🔍
Alerts Output	33	🔍

The Alerts Output row shows the number of alerts that have been triggered during the session.

3 Click the magnifying glass in the Alerts row to open the Alerts Output Window.

4 To sort the rows, click the column name by which you want to sort the entries.

5 To show alert details, click **Show Alert Details**. The details section opens in the lower part of the Output window.l



Time	Name	Machine	Measurement	Condition	Descripiton
03/11/2002 14:12:47	Alert_Set4		Error	value change over 15 sec...	

Add a description in the Description box beneath the Alerts list.

6 To hide alert details, click **Hide Alert Details**. The details section closes.

7 To close the Alerts output window, click the *x* in the upper right corner, or click the magnifying glass in the Alerts row in the Statistics section.

Note that alert information is only saved when alerts were enabled during the script's execution. If you enabled alerts after script execution, you will have to run the step again in order to generate alerts.

8

Scheduling Session Steps

After you create a step, you use the Schedule Builder to specify when the step should begin running. In addition, you can set the duration of the step, or specify that a step should be run till a goal is reached.

This chapter describes:

- Delaying the Start of a Session Step
- Creating and Selecting a Schedule Profile
- Creating a Manual Profile
- Creating a Goal Oriented Profile

About Scheduling Session Steps

An important factor in the creation of a test step, is developing a step that accurately portrays user behavior—the types of actions and the timing of those actions, represented by the scripts.

The Schedule Builder allows you to specify when to run steps, and for how long. You do this by creating *profiles* and associating them with the steps. Each profile defines a specific way of testing. By creating multiple profiles you can run the same step under different conditions.

The Design tab's Schedule Summary section displays information about the profile currently associated with the selected step. It allows you to choose a different profile from the list of available profiles in the Schedule Name box.

The Schedule Builder allows you to create the following types of profile: *manual* and *goal oriented*.

In a manual profile, you specify:

- start time of a test
- duration
- number of Vusers that will run the test
- the ramp up and ramp down processes

Manual profiles include benchmark profiles, which cause ProTune to run all the enabled scripts on all the Load Generators.

In a goal oriented profile, you specify the goal you want to reach (for example, the response time that is considered unacceptable). ProTune runs the step, adding Vusers, till the step reaches the goal.

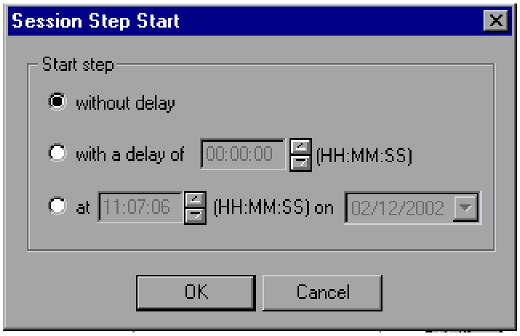
The profile you define is visually displayed in the Schedule Builder's Load Preview graph.

Delaying the Start of a Session Step

You can instruct ProTune to start running the session step at a later point in time. You can specify either the time you want ProTune to wait from the moment an *Execute* command is issued, or the specific time at which you want the step to begin.

To delay the start of a session step:

1 Select **Session** > **Start Time**. The Session Step Start dialog box opens, with the default option—**without delay**—selected.



Session Step Start

Start step

● without delay

○ with a delay of 00:00:00 (HH:MM:SS)

○ at 11:07:06 (HH:MM:SS) on 02/12/2002

2 Select **with a delay of xxx (HH:MM:SS)** and enter the period (in hours:minutes:seconds format) by which you want to delay the start of the step.

Alternatively, you can select **at xxx (HH:MM:SS) on xxx** and specify the time (in hours:minutes:seconds format) and date for the start of the step.

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

The next time you run a test, the start time will be delayed as specified.

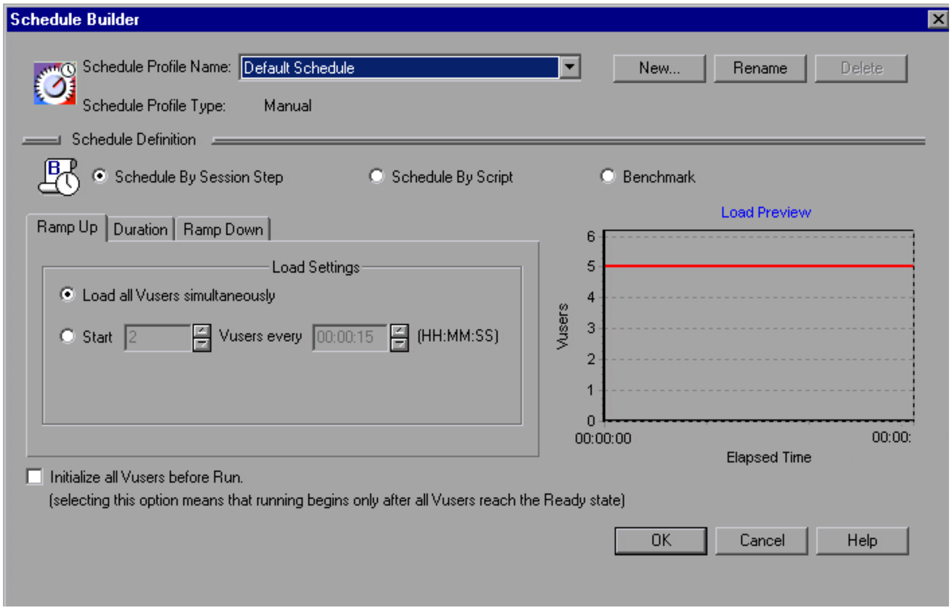
Creating and Selecting a Schedule Profile

You select the schedule profile that you want to use for your session step from the Schedule Profile Name list box in the Schedule Builder window. The default profile loads all Vusers simultaneously. You use the Schedule Builder for creating new profiles and modifying existing ones.

To invoke the Schedule Builder:

Scheduler

➤ Choose **Session** > **Schedule Builder** or click one of the Schedule Builder icons (in the toolbar or in the Schedule Summary section). The Schedule Builder window opens, displaying the last profile that was associated with the step. The profile name is displayed in the Schedule Profile Name list box. If no profile has been associated with the step, ProTune by default uses a manual profile and assigns it the name *Default Schedule*.



Schedule Builder

Schedule Profile Name: Default Schedule

Schedule Profile Type: Manual

Schedule Definition

● Schedule By Session Step ○ Schedule By Script ○ Benchmark

Ramp Up | Duration | Ramp Down

Load Settings

● Load all Vusers simultaneously

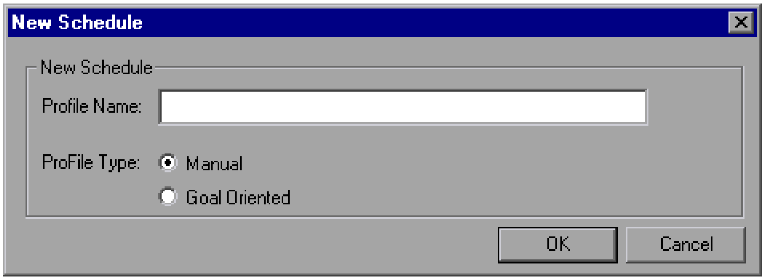
○ Start 2 Vusers every 00:00:15 (HH:MM:SS)

□ Initialize all Vusers before Run.
(selecting this option means that running begins only after all Vusers reach the Ready state)

To create a new profile:

1 Invoke the Schedule Builder.

2 Click the **New** button in the Schedule Builder window. The New Schedule dialog box opens.



New Schedule

New Schedule

Profile Name:

ProFile Type: ● Manual

○ Goal Oriented

3 In the **Schedule Profile Name** text box, enter the name of the new profile.

4 Choose the profile type—Manual or Goal Oriented—by clicking the appropriate radio button, and click **OK**.

The new profile name appears in the **Schedule Profile Name** list box in the Schedule Builder window.

5 Set the values for your profile (see “Creating a Manual Profile,” on page 100 or “Creating a Goal Oriented Profile,” on page 106).

To modify the properties of an existing profile:

1 Invoke the Schedule Builder.

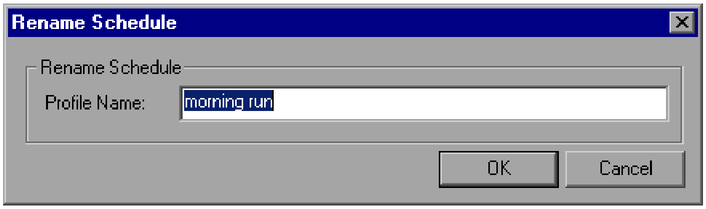
2 In the Schedule Builder window, select the profile you want to modify from the Schedule Profile Name list box.

3 In the Schedule Builder dialog box, modify the profile as required.

To rename a profile:

1 In the Schedule Builder window, select the profile you want to rename from the Schedule Profile Name list box.

2 Click **Rename**. The Rename Schedule dialog box appears.



Profile Name: morning run

3 Enter a new name for the selected profile, and click OK. The new name appears in the Schedule Profile Name list box.

To delete a profile:

1 In the Schedule Builder window, select the profile you want to delete from the Schedule Profile Name list box.

2 Click **Delete**. The profile is deleted, and no longer appears in the Schedule Profile Name list box.

Creating a Manual Profile

If you are using a manual profile, you can use the following types of scheduling:

- Session Step Scheduling
- Script Scheduling
- Benchmark Scheduling

Session Step Scheduling

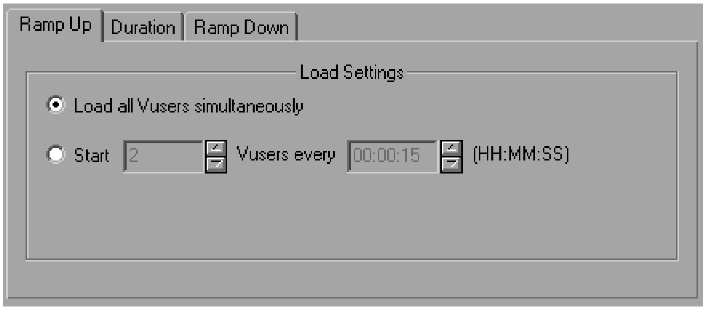
Session step scheduling includes specifying the following:

- how the step should be started (ramped up). This allows you to choose between gradually adding Vusers to the running test, or starting all the Vusers simultaneously when the test starts.
- the step duration
- how the step should be stopped (ramped down). This allows you to choose between gradually stopping Vusers that are running, and stopping them all simultaneously.

Note: When you schedule by session step, your settings apply to all the scripts included in the step. For example, if you specify a duration, all the scripts will be executed for the specified period.

To specify how the step should be started:

1. In the Schedule Builder window, click the Schedule by Session Step radio button.
2. Click the Ramp Up tab:



Load Settings

● Load all Vusers simultaneously

○ Start 2 Vusers every 00:00:15 (HH:MM:SS)

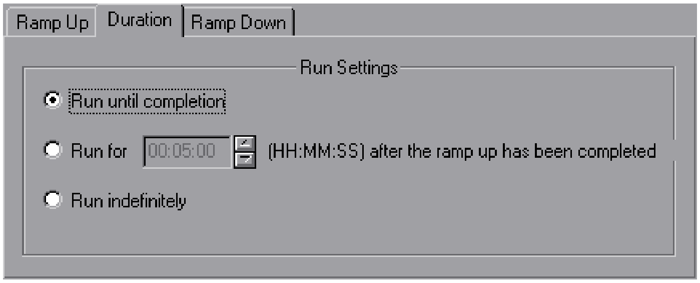
➤ To start all the Vusers running at the same time when the test starts, select **Load all Vusers simultaneously**.

➤ To gradually run the Vusers, enter the number of Vusers you want to begin running concurrently, and the period that you want ProTune to wait before adding more Vusers.

Note: While a step is running, you can add scripts to the step and enable them, as long as not all of the Vusers have been ramped up. However, if you add a script after all the Vusers have been ramped up, the new script will not run in the step. To enable running the new script in the step, you need to stop the step and restart it.

To specify the duration of the step:

1 In the Schedule Builder window, click the **Duration** tab.



Run Settings

● Run until completion

○ Run for 00:05:00 (HH:MM:SS) after the ramp up has been completed

○ Run indefinitely

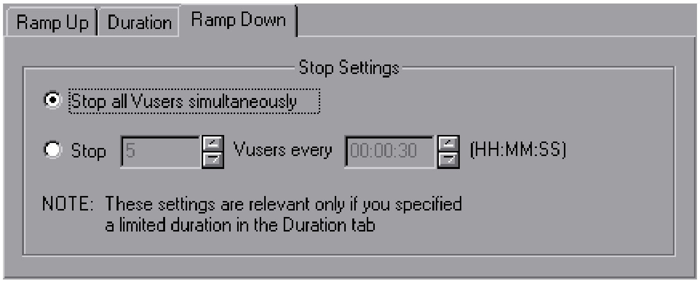
2 Choose one of the following options:

- Run until completion.
- Run for a specified period after all the Vusers have been ramped up.
- Run indefinitely.

Note: The duration setting overrides the Vuser iteration run-time setting. For example, if you specify the duration here as five minutes, the Vusers will continue to run as many iterations as required in five minutes, even if the run-time settings specify only one iteration.

To specify how the step should be stopped:

1 Click the **Ramp Down** tab.



Stop Settings

● Stop all Vusers simultaneously

○ Stop 5 Vusers every 00:00:30 (HH:MM:SS)

NOTE: These settings are relevant only if you specified
a limited duration in the Duration tab

2 Choose one of the following options:

➤ **Stop all Vusers simultaneously:** Stops all the Vusers in the session step at once.

➤ **Stop X Vusers every X (HH:MM:SS):** Stops a certain number of Vusers within a specified time frame. For example, you might want to stop 5 Vusers every 30 seconds.

Note: The Ramp Down tab settings are enabled only if you selected the second option in the Duration tab.

To instruct ProTune to initialize Vusers before beginning to load them:

1 Check the **Initialize all Vusers before run** box. ProTune will begin to load the Vusers only after they have all reached the READY state.

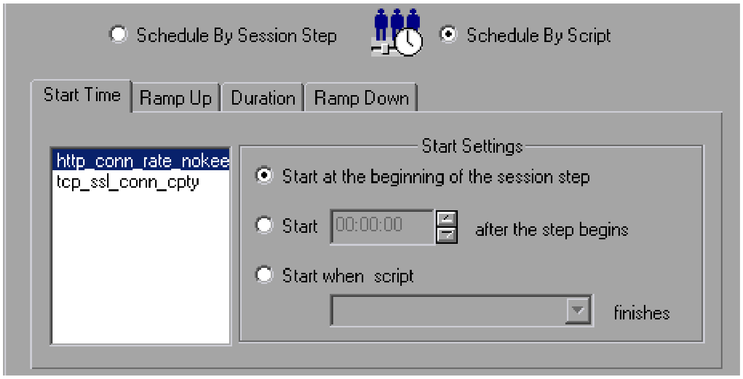
Click **OK** to close the Schedule Builder and save your settings.

Script Scheduling

Script scheduling allows you to specify settings for each script in the step separately. For example, you can schedule a different duration for each script.

To schedule scripts:

1. In the Schedule Builder window, click the Schedule by Script radio button. ProTune displays a Start Time tab, in addition to the tabs displayed when you schedule by session step (see "Session Step Scheduling," on page 100).



○ Schedule By Session Step ● Schedule By Script

http_conn_rate_nokee
tcp_ssl_conn_cpty

Start Settings

● Start at the beginning of the session step

○ Start 00:00:00 after the step begins

○ Start when script finishes

2. The Start Time tab includes a list of the scripts contained by the step. To specify settings for a particular script, click the step name to select it and choose one of the following settings:

   ➤ To run the script at the beginning of the session step, click the appropriate radio button.

   ➤ To delay running the script, click the **Start HH:MM:SS after the step begins** radio button, and specify the delay period in HH:MM:SS format. ProTune will run the script when the specified period has passed after the step has begun executing.

   ➤ To make the script dependent on another script in the step finishing execution, click the **Start when script finishes** radio button and choose a script from the list box. ProTune will run your script only after the specified script has been run.

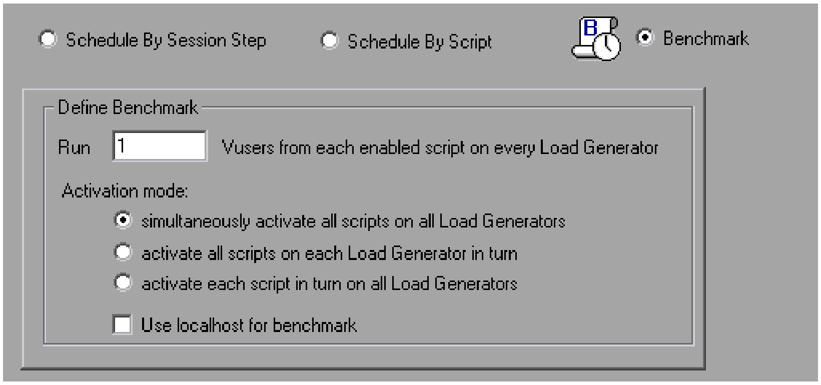
3. Specify the settings in the other tabs (ramp up, duration and ramp down) as described in "Scheduling Session Steps," on page 95.

Benchmark Scheduling

Benchmark scheduling allows you to test all of the enabled scripts on all the Load Generators (instead of only testing each script on the machine to which it is assigned). Benchmark scheduling is particularly useful when dealing with a complicated system topology that has a large number of servers and hosts. Running a benchmark test before tuning lets you verify that all the hosts and scripts are valid.

To schedule by benchmark:

1 In the Schedule Builder window, click the Schedule by Benchmark radio button. ProTune displays the Define Benchmark pane:



○ Schedule By Session Step ○ Schedule By Script ● Benchmark

Define Benchmark

Run 1 Vusers from each enabled script on every Load Generator

Activation mode:

● simultaneously activate all scripts on all Load Generators

○ activate all scripts on each Load Generator in turn

○ activate each script in turn on all Load Generators

□ Use localhost for benchmark

2 In the Run Vusers text box, enter the number of Vusers that ProTune should run from each enabled script or Load Generator. For example, if you enter the number 5, each script will be run by five Vusers.

3 Click the appropriate radio button to choose one of the following benchmark modes:

- Simultaneously activate all the enabled scripts on all of the Load Generators.
- Activate all the enabled scripts on each Load Generator in turn. For example, ProTune first runs all the scripts on Load Generator 1, next on Load Generator 2, and so on.
- Activate each enabled script in turn on all the Load Generators. For example, ProTune runs script 1 on Load Generator 1, next on Load Generator 2, and so on until the script has been run on all the Load Generators. ProTune next runs script 2 on all of the Load Generators, and so on until all the scripts have been run on all the Load Generators.

4 If you want to use the local machine as one of the Load Generators, check the **Use localhost as one of the Load Generators** box.

Creating a Goal Oriented Profile

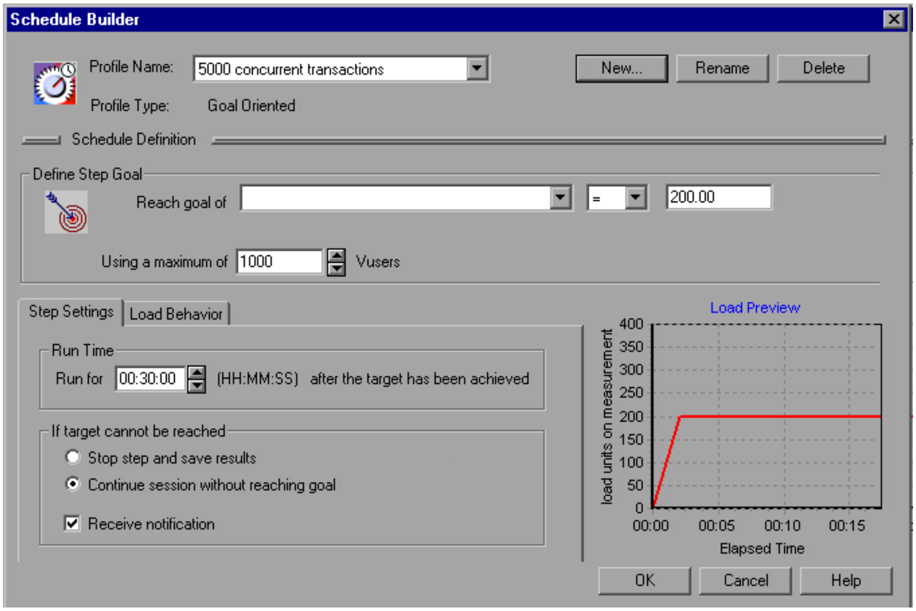
You can set one goal per session step—not per script.

When you run a session step, the goal you defined is displayed in the appropriate graph, along with the session results. This enables you to compare the results with your target goal and determine if your goal was reached. If your goal is not reached, you reconfigure your applications and servers accordingly, in order to reach the desired goal.

Creating a goal oriented profile includes defining the following:

- Step goal—the goal that you want to achieve before terminating execution of the step. This includes what you want to measure (for example, the number of concurrent transactions) and the number of Vusers participating in the test.
- Step settings—when ProTune should run the step and what it should do if the goal is not reached.
- Load behavior—how and when you want Protune to reach your target.

When you create a goal oriented profile, the Schedule Builder displays the following window:



Profile Name: 5000 concurrent transactions

Profile Type: Goal Oriented

Reach goal of = 200.00

Using a maximum of 1000 Vusers

Run for 00:30:00 (HH:MM:SS) after the target has been achieved

If target cannot be reached

○ Stop step and save results

● Continue session without reaching goal

☑ Receive notification

To define the step goal:

1 Choose the type of goal for your step from the **Reach goal of** list box.

The list of goal types includes:

- basic HTTP-related goals. These goal types are displayed when you open the list box.
- additional goals that are related to the monitors and measurements you have specified. To view these goal types, choose <**more**> in the **Reach goal of** box.

Following is the list of basic HTTP-related goals:

- **Throughput**—target downstream bandwidth.
- **Hits / Second**—target number of hits per second (HTTP requests per second) that you would like your step to reach. When you choose this goal, you also need to enter the maximum number of Vusers for the session step.

- **Number of Connections**—target number of connections that you would like the server to host.
- **Connections / Second**—target number of connections per second you would like the server to handle.

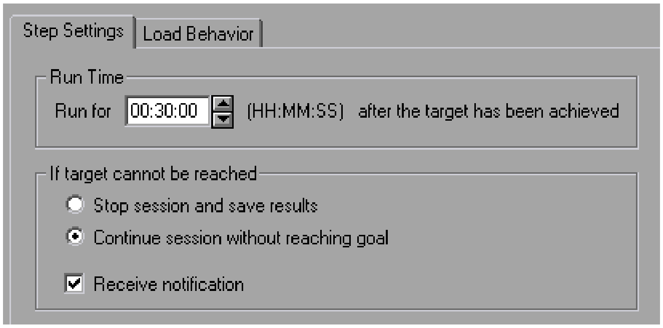
2 Choose an operator (either "=" or ">=")from the middle list box.

3 Enter a value in the text box on the right side.

4 Specify the maximum number of Vusers in the **Using a Maximum of ... Vusers** box.

To define step settings:

1 Click the Step Settings tab.



Step Settings | Load Behavior

Run Time

Run for 00:30:00 (HH:MM:SS) after the target has been achieved

If target cannot be reached

○ Stop session and save results

● Continue session without reaching goal

☑ Receive notification

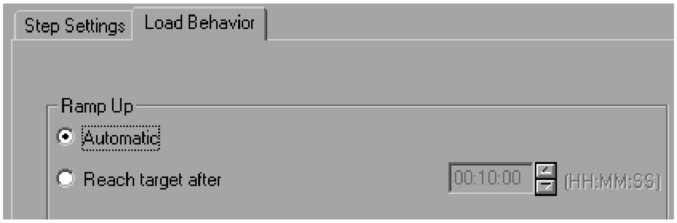
2 Enter a value (in HH:MM:SS format) in the **Run For** box. This value specifies how long the step will run after the goal has been reached.

3 Specify what ProTune should do if the goal is not reached: click the appropriate radio button to either stop the session and save the results, or continue the session despite not reaching the goal.

4 Check the Receive Notification box if you want ProTune to display a message when it determines that the goal cannot be reached.

To define load behavior:

1 Click the Load Behavior tab.



Step Settings | Load Behavior

Ramp Up

● Automatic

○ Reach target after 00:10:00 (HH:MM:SS)

2 Specify the ramp up as follows:

➤ To run the default number of Vusers in a batch, click the **Automatic** radio button.

➤ To specify the period of step time that should elapse before the Console reaches your target, click the **Reach target after** radio button and enter the period in the box in HH:MM:SS format.

Note: ProTune will start execution of the step by running one Vuser. When 2 minutes have passed, ProTune will calculate the required number of Vusers, and start ramping up as defined in this tab.

9

Preparing to Run a Session Step

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

This chapter describes:

- Specifying a Results Location
- Results Directory File Structure
- Collating Results

About Preparing to Run a Session Step

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

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

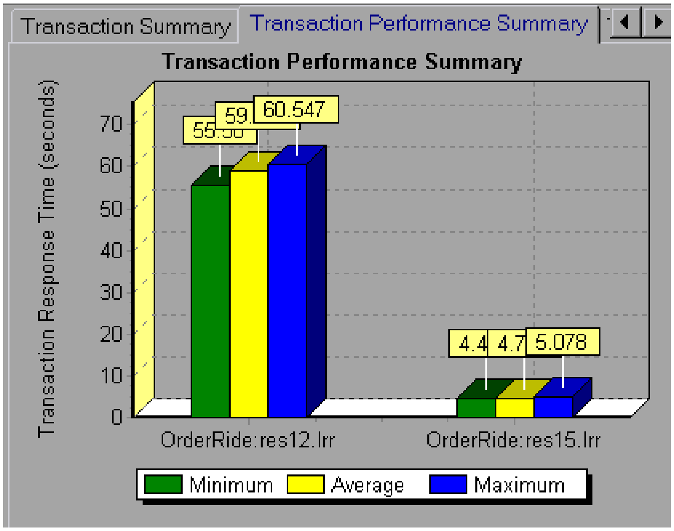
For information on one-time configuration settings such as timeout, output, and quotas, see Chapter 6, "Configuring Session Steps."

Specifying a Results Location

When you run a session step, by default the run-time files are stored locally on each load generator. After the session step, the results are collated together and processed on the Console machine. Alternatively, you can instruct ProTune to save the results on a shared network drive. For information about specifying a file storage method, see the Run-Time File Storage settings in Chapter 6, "Configuring Session Steps."

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

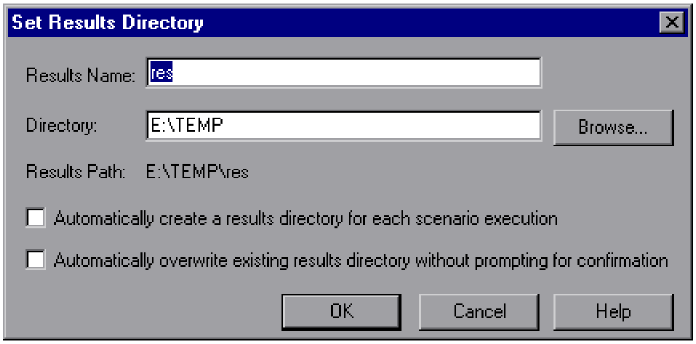
In the example below, the results of two session step runs are superimposed. The result sets are *res12*, and *res15*.



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

To specify where results are stored:

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



Set Results Directory

Results Name: res

Directory: E:\TEMP

Results Path: E:\TEMP\res

□ Automatically create a results directory for each scenario execution

□ Automatically overwrite existing results directory without prompting for confirmation

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

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

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

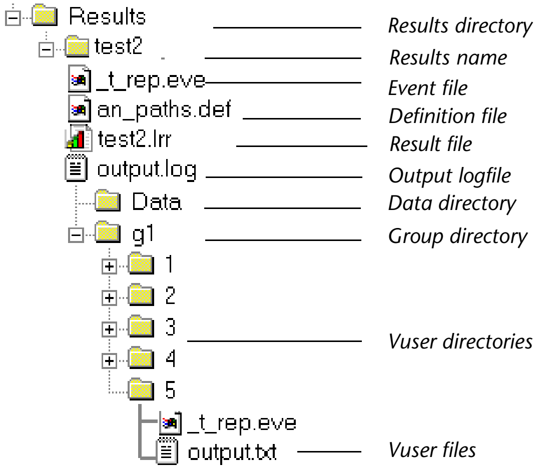
4 Select the appropriate check box for subsequent executions: **Automatically create a results directory for each session step execution** or **Automatically overwrite existing results directory without prompting for confirmation**.

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

Results Directory File Structure

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

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



- ➤ *t_rep.eve* in the main result directory contains Vuser and rendezvous information.
- ➤ **.def* are definition files for graphs that describe the online and other custom monitors.
- ➤ *results_name.lrr* is the ProTune Analysis document file.
- ➤ *output.log* contains output information about the session step generated during test execution.
- ➤ The *Data* directory contains the database created by the Analysis (from the results files).
- ➤ *g1* is a group directory. A separate directory exists for each Vuser group that runs in the session step. Each group directory consists of Vusers subdirectories.
- ➤ *t_rep.eve* in each Vuser directory contains transaction information.
- ➤ *output.txt* in each Vuser directory contains output information generated during replay.

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

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

Collating Results

When you run a session step, by default all Vuser information is stored locally on each load generator. After session step execution, the results are automatically collated or consolidated—results from all of the load generators are transferred to the results directory. You set the location of the results directory as described in "Specifying a Results Location," on page 111.

Note: If you have selected to store all the session step results directly to a shared network drive, then collation of the results is not required. See "About Configuring a Session," on page 69 for details on changing how results are stored.

To disable automatic collation and clear the check mark adjacent to the option, choose **Results** > **Auto Collate Results**. To manually collate results, choose **Results** > **Collate Results** > **Collate**. The Collating Files dialog box opens, displaying the progress of result and log file collation from each load generator. To stop collating the results and close the dialog box, click **Stop** and then **Close.** To resume collating the results, select **Results** > **Collate Results** > **Continue stopped collation**.

Note: You can choose to disable log file collation. For more information, see "Options - General Settings," on page 517.

The log and result directories are only deleted from a load generator once ProTune successfully collates the results from the machine. You can therefore close the Console after saving a session step, and collate the results once you reopen the session step in the Console.

If collation fails due to a lack of disk space, select **Results** > **Collate Results** > **Recollate**. ProTune attempts to collate the results again, without compressing the *.eve* file.

Before generating the analysis data, ProTune automatically collates the results if they have not previously been collated.

Note: If you enabled the **Auto Load Analysis** option in the Results menu, the Analysis may open during a lengthy collation process, displaying Analysis summary data.

Part IV

Executing a Tuning Session

10

Running a Session

When you tune a session, ProTune simulates your environment and measures the system's performance.

This chapter describes:

- Running an Entire Session
- Controlling a Specific Number of Vusers
- Adding Vusers to a Running Session
- Controlling Individual Vusers
- Invoking the System Topology Window

About Running a Session Step

When you run a session, the Vusers are assigned to their load generators and execute their scripts. During session execution, ProTune:

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

You can run an entire session unattended, or you can interactively select the Vusers that you want to run. When the session starts running, the Console first checks the session configuration information. Next, it invokes the scripts that you selected to run with the session. Then, it distributes each script to its designated load generator. When the Vusers are ready, they start executing their scripts.

While the session runs, you can monitor each Vuser, view error, warning, and notification messages generated by the Vusers, and stop both Vuser groups and individual Vusers. You can instruct ProTune to allow an individual Vuser or the Vusers in a group to complete the iterations they are running before stopping, to complete the actions they are running before stopping, or to stop running immediately. For more information, see "Configuring Session Run-Time Settings" on page 70.

You can also activate additional Vusers while the session is running, using the Run/Stop Vusers dialog box. For more information, see "Adding Vusers to a Running Session" on page 123.

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

The following procedure outlines how to run a session step:

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

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

Running an Entire Session

You can run all the Vusers in a session, or you can select the number of Vusers that you want to run. Note that when you run your session, ProTune runs them as soon as they reach the READY state.

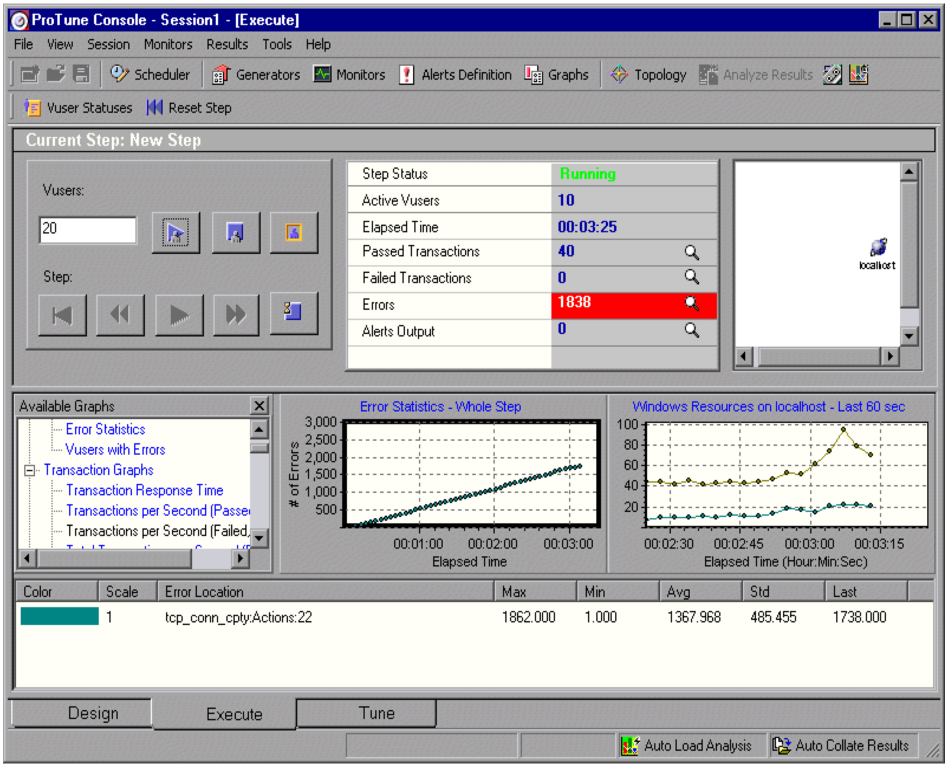
The following section describes how to run an entire session. “Controlling Individual Vusers,” on page 126 describe how to manipulate individual Vusers.

To run a session step:

1 Open an existing session or create a new one.

▶ Execute

2 Click the **Execute** button in the bottom right of the Design tab, or click the Execute tab and then click the **Start the current Session Step** button. The ProTune Console Session window displays the selected monitors.


ProTune Console - Session1 - [Execute]
File View Session Monitors Results Tools Help
Scheduler Generators Monitors Alerts Definition Graphs Topology Analyze Results
Vuser Statuses Reset Step
Current Step: New Step
Vusers:
20
Step:
localhost
Available Graphs
Error Statistics
Vusers with Errors
Transaction Graphs
Transaction Response Time
Transactions per Second (Passe
Transactions per Second (Failed,
Error Statistics - Whole Step
# of Errors
3,000
2,500
2,000
1,500
1,000
500
00:01:00
00:02:00
00:03:00
Elapsed Time
Windows Resources on localhost - Last 60 sec
100
80
60
40
20
00:02:30
00:02:45
00:03:00
00:03:15
Elapsed Time (Hour:Min:Sec)
Design
Execute
Tune
Auto Load Analysis
Auto Collate Results

Step Status	Running
Active Vusers	10
Elapsed Time	00:03:25
Passed Transactions	40
Failed Transactions	0
Errors	1838
Alerts Output	0

Color	Scale	Error Location	Max	Min	Avg	Std	Last
🟩	1	tcp_conn_cpty:Actions:22	1862.000	1.000	1367.968	485.455	1738.000

3 Choose **Session** > **Stop** or click the **Stop** button to terminate session step execution. If you selected the **Exit immediately** option in the Run-Time Settings tab of the Options dialog box, all of the Vusers in the session move to the EXITING status.

If you selected the **Wait for the current iteration to end before exiting** or **Wait for the current action to end before exiting** options in the Run-Time Settings tab of the Options dialog box, the Vusers in the session move to the GRADUAL EXITING status and exit the session gradually. To stop the Vusers immediately, click **Stop Now**.

4 Click the **First Step** button to load the first session step.

5 Click the **Previous Step** button to load the previous session step.

6 Click the **Current Step** button to run the Vusers in the first session step.

7 Click the **Next Step** button to load the next session step.

8 Click the **Reset Step** button to reset the session step. This initializes all the Vusers, returning them to DOWN status, and clears the step statistics.

You can also add Vusers during the session run. For more information, see "Adding Vusers to a Running Session" on page 123.

Controlling a Specific Number of Vusers

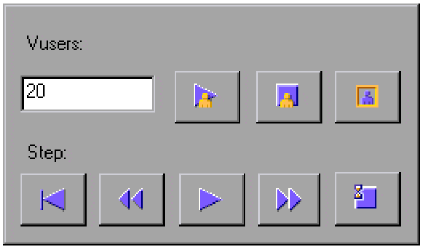
You can instruct ProTune to run and stop a specific number of users. It distributes these users between the load generators as you defined in the design stage.

To control a specific number of Vusers:

1 Open an existing session or create a new one in the **Design** tab.

2 Select the **Execute** tab. The ProTune Console Session window opens with the selected monitors.

3 In the upper left section, enter the number of Vusers you want to manipulate in the Vusers box



Vusers: 20

Step:

4 To run a specific number of Vusers, click the **Run Specific Vusers** button. ProTune begins running those Vusers.

5 To stop a specific number of Vusers, click the **Stop Specific Vusers** button. ProTune stops those Vusers.

Adding Vusers to a Running Session

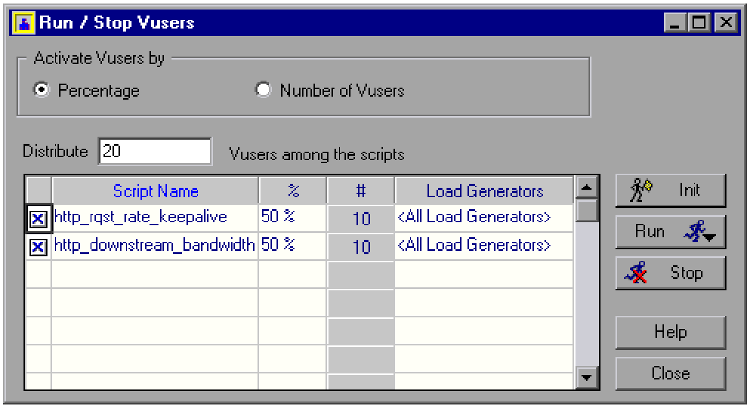
You can add Vusers to a running session to manually ramp up the load and see how your system performs when users are added.

To add Vusers to a running session:

1 Click the **Add Vusers to a Running Step** button in the top left section of the Execute tab. The Run/Stop Vusers dialog box opens.

Enter the total number of Vusers to distribute between the checked scripts.

- Select **Percentage** in the **Activate Vusers by** section to display the Vusers by their percentage distribution.



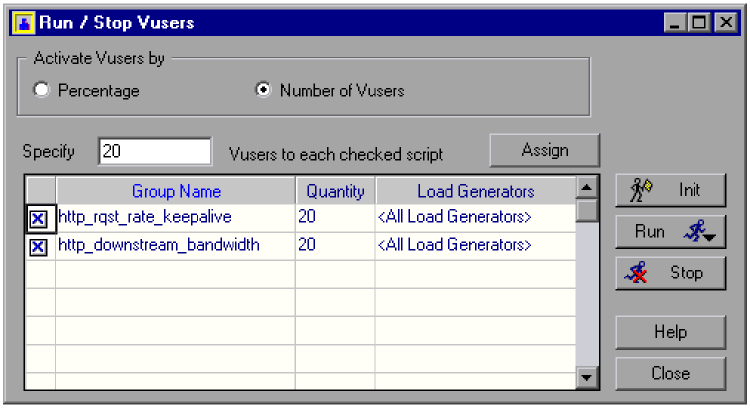
Activate Vusers by

● Percentage ○ Number of Vusers

Distribute 20 Vusers among the scripts

	Script Name	%	#	Load Generators
☒	http_rqst_rate_keepalive	50 %	10	<All Load Generators>
☒	http_downstream_bandwidth	50 %	10	<All Load Generators>

- Select **Number of Vusers** in the **Activate Vusers by** section to display the Vusers by their quantity distribution. Note that when you choose this option, the **Assign** button is displayed.



Activate Vusers by

○ Percentage ● Number of Vusers

Specify 20 Vusers to each checked script

	Group Name	Quantity	Load Generators
☒	http_rqst_rate_keepalive	20	<All Load Generators>
☒	http_downstream_bandwidth	20	<All Load Generators>

2 Enter the total number of Vusers to distribute between the checked scripts. ProTune distributes the Vusers evenly between the scripts.

3 To manually set the distribution:

- By **Percentage**: If you are in Percentage mode, enter the number of Vusers you want to distribute by percentage among the checked scripts. The % column indicates the percentage of Vusers distributed to each script. The # column indicates the number of Vusers distributed to each script

- By **Number of Vusers**: Enter the number of Vusers you want to run for each group in the group's Quantity column, or click the **Assign** button to assign the number of Vusers specified in the **Specify ... Vusers** field to each checked enabled script.

Note: If more than one load generator is defined for a script, the added Vusers are proportionally distributed among the defined load generators.

To disable a script, clear the check box to the left of the script name. Note that a script will automatically appear disabled if it is disabled in the Design view.

Note: If you disable a script, no Vusers will be distributed to it. However, 100 percent of the Vusers will not be distributed among the remaining scripts, unless you define a 0 percent value for the disabled script.

---

4. Click the **Init** button to initialize the number of Vusers you added. The Console first initializes the Vusers in your session that have not yet been run, on the load generator(s) defined in the Run/Stop Vusers dialog box. It then adds additional Vusers, as required, to reach the quantity defined in the Run/Stop Vusers dialog box.
5. Click the **Run** button, and select one of the following two options:
   - **Run Initialized:** Runs the Vusers in the session that have already been initialized on the load generators defined in the Run/Stop Vusers dialog box. The Console runs only those Vusers that have already been initialized, regardless of their quantity.
   - **Run New:** Runs the number of Vusers you specified. The Console first runs the Vusers in your session that have not yet been run, on the load generator(s) defined in the Run/Stop Vusers dialog box. It then adds

additional Vusers, as required, to reach the quantity defined in the Run/Stop Vusers dialog box.

6 Click **Stop** to stop the Vusers that are running on the load generator(s) defined in the Run/Stop Vusers dialog box. The Console stops the Vusers according to the settings you defined in the Run-Time Settings tab of the Options dialog box.

7 Click **Close** to close the Run/Stop Vusers dialog box.

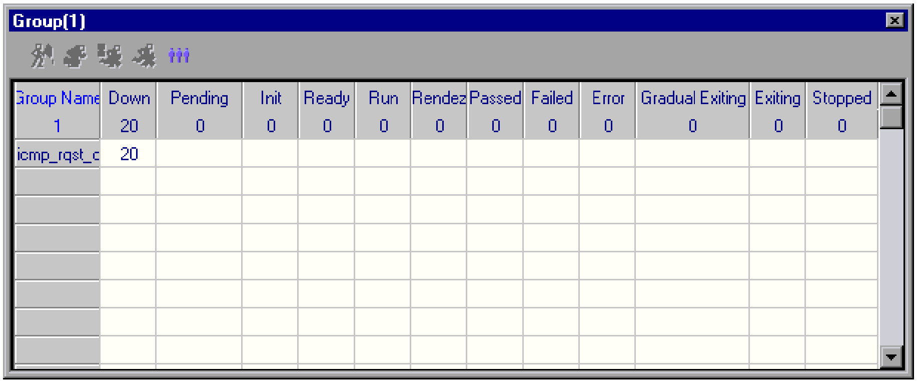
Controlling Individual Vusers

You can also manipulate individual Vusers. This section describes how to initialize, run, and stop individual Vusers.

To control an individual Vuser:

Vuser Statuses

1 Choose **Session** > **See Vuser statuses** or click the **Vuser Statuses** button to open the Groups window. ProTune lists all of the active scripts and the statuses of the Vusers running them.

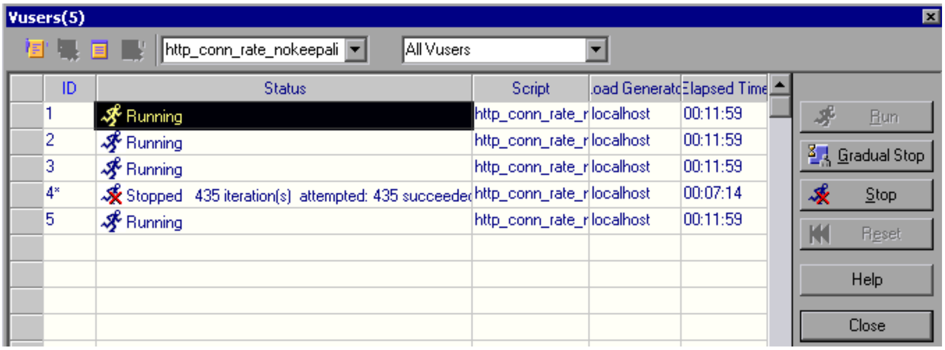


Group(1)

Group Name	Down	Pending	Init	Ready	Run	Rendez	Passed	Failed	Error	Gradual Exiting	Exiting	Stopped
1	20	0	0	0	0	0	0	0	0	0	0	0
icmp_rqst_c	20											

2 Double-click on a script, or select the script and click the **Groups** button. The Vusers dialog box opens, with a list of the ID, Status, Script, Load

Generator, and Elapsed Time (since the beginning of the session) for each of the Vusers in the group.



Vusers(5)

http_conn_rate_nokeepali

All Vusers

ID	Status	Script	.oad Generatc	Elapsed Time
1	Running	http_conn_rate_r	localhost	00:11:59
2	Running	http_conn_rate_r	localhost	00:11:59
3	Running	http_conn_rate_r	localhost	00:11:59
4*	Stopped 435 iteration(s) attempted: 435 succeedec	http_conn_rate_r	localhost	00:07:14
5	Running	http_conn_rate_r	localhost	00:11:59

You can control an individual Vuser using the following utilities:

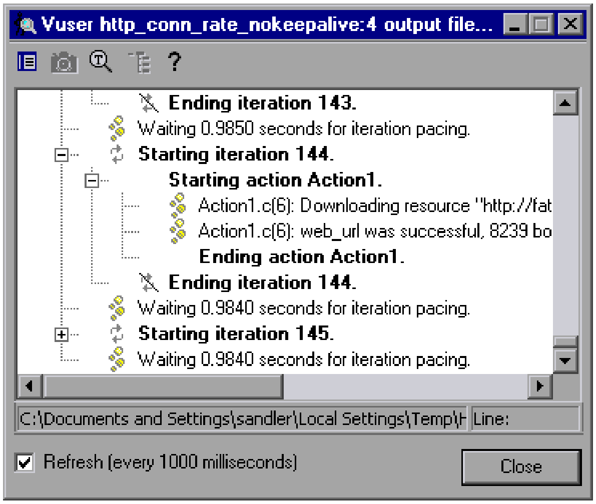
- Select a Vuser and click **Run** to run it.
- Select a Vuser and click **Stop** to stop it immediately from running.

  If you selected the **Wait for the current iteration to end before exiting** or **Wait for the current action to end before exiting** options in the Run-Time Settings tab of the Options dialog box, and want to gradually stop a Vuser in the RUN state, click the **Gradual Stop** button. The Vuser moves to the GRADUAL EXITING status and exits the session gradually.

3 To pause a Vuser, right-click it and select **Pause**.

- Select a Vuser and click **Reset** to revert its status to DOWN.
- To initialize a Vuser, right-click it and select **Initialize Vuser/s**.
- To renumber the Vusers in a group, right-click the Vusers you want to renumber and select **Renumber**.
- To filter the Vusers listed, right-click in one of the columns and select **Filter Vusers.** Select the way in which you want to filter the Vusers. Alternatively, you can select the filter option you want to use from the right-hand filter selector at the top of the Vusers dialog box.
- To sort the Vusers listed, right-click in one of the columns and select **Sort Vusers**. Select the way in which you want to sort the Vusers.

- To view a Vuser executing its assigned script, select the Vuser and click the **Show** button. The Run-Time Viewer opens, allowing you to see the Vuser executing the script.
- To close the Run-Time Viewer, click the **Hide** button.
- To view the script log, click the **Vuser log** button. A script log, such as the following, appears.



Vuser http_conn_rate_nokeepalive:4 output file...

```
Ending iteration 143.
Waiting 0.9850 seconds for iteration pacing.
Starting iteration 144.
Starting action Action1.
Action1.c(6): Downloading resource "http://fat
Action1.c(6): web_url was successful, 8239 bo
Ending action Action1.
Ending iteration 144.
Waiting 0.9840 seconds for iteration pacing.
Starting iteration 145.
Waiting 0.9840 seconds for iteration pacing.
```

C:\Documents and Settings\sandler\Local Settings\Temp\F Line:

☑ Refresh (every 1000 milliseconds)

To close the script log, click **Close** in the log window or click the Close Log button in the Vusers dialog box. For more information on the script log, see page 138.

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

Invoking the System Topology Window

The Execute view’s topology pane displays the system topology you defined (see “Creating a Topology,” on page 19).

During a tuning session, you may want to make adjustments to the topology or view its details. You can open the System Topology window either by clicking the Topology button on the main toolbar, or double-clicking the white space in the Execute View’s topology pane.

11

Viewing Vusers During Execution

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

This chapter describes:

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

About Viewing Vusers During Execution

ProTune lets you view Vuser activity during a session:

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

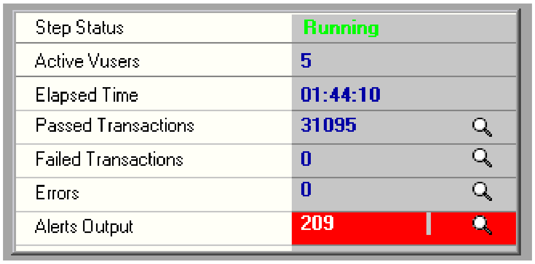
Monitoring Vuser Status

During session execution, you can use the Session Groups window in the Execute view to monitor the actions of all the Vusers and Vuser groups in the session.

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

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

You can also view a synopsis of the running session in the status window at the top of the Execute view.



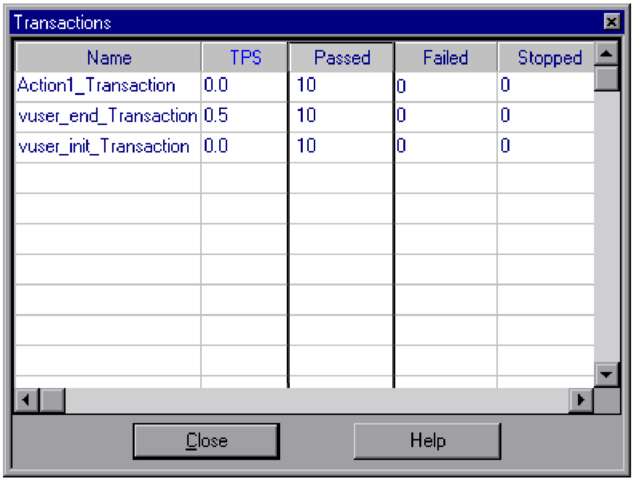
Step Status	Running
Active Vusers	5
Elapsed Time	01:44:10
Passed Transactions	31095
Failed Transactions	0
Errors	0
Alerts Output	209

Status Summary	**Description**
STEP STATUS	indicates whether the session is RUNNING or DOWN
ACTIVE VUSERS	indicates how many Vusers are being executed on a load generator machine
ELAPSED TIME	indicates how much time has elapsed since the beginning of the session
PASSED TRANSACTIONS	indicates how many transactions have been executed successfully
FAILED TRANSACTIONS	indicates how many transactions have been executed unsuccessfully
ERRORS	indicates how many problems have occurred with the Vusers
ALERTS OUTPUT	indicates how many alerts have been triggered during the tuning session

Note: When running a goal oriented step, the measurement used for the goal appears following the elapsed time.

To view details of the transactions and errors:

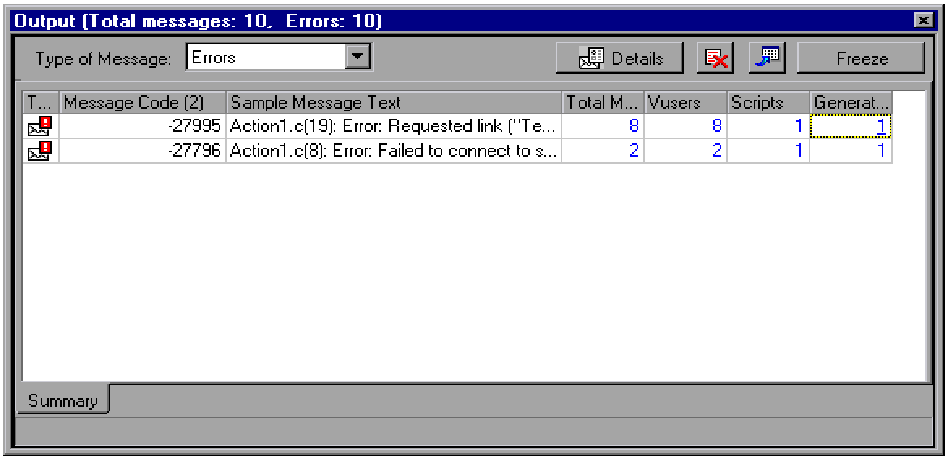
1 Click the **Show Snapshot** button to the right of the Passed Transactions or Failed Transactions in the Session Status window. The Transactions dialog box opens.



Name	TPS	Passed	Failed	Stopped
Action1_Transaction	0.0	10	0	0
vuser_end_Transaction	0.5	10	0	0
vuser_init_Transaction	0.0	10	0	0

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

2 Choose **View** > **Show Output** or click the **Show Snapshot** button to the right of the Errors listing. The Output window opens, displaying a list of the Error log information.



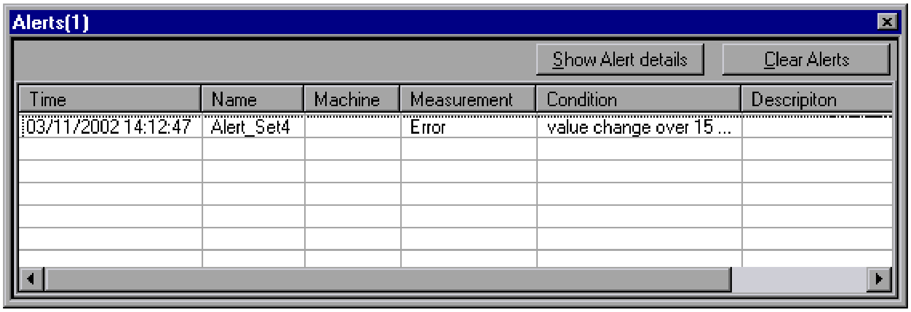
Output (Total messages: 10, Errors: 10)

Type of Message: Errors

T...	Message Code (2)	Sample Message Text	Total M...	Vusers	Scripts	Generat...
	-27995	Action1.c(19): Error: Requested link ("Te...	8	8	1	1
	-27796	Action1.c(8): Error: Failed to connect to s...	2	2	1	1

For each type of error message code, the Output window lists a sample message text, the total number of messages generated, the Vusers and load generators that generated the code, and the scripts in which the errors occurred. To view details of the log information by message, Vuser, script, or load generator, click the link in the respective column. For more information on the Output window, see the following section.

3 Click the **Show Snapshot** button to the right of the Alerts Output in the Session Status window to open the Alerts Output dialog box.



Alerts(1)

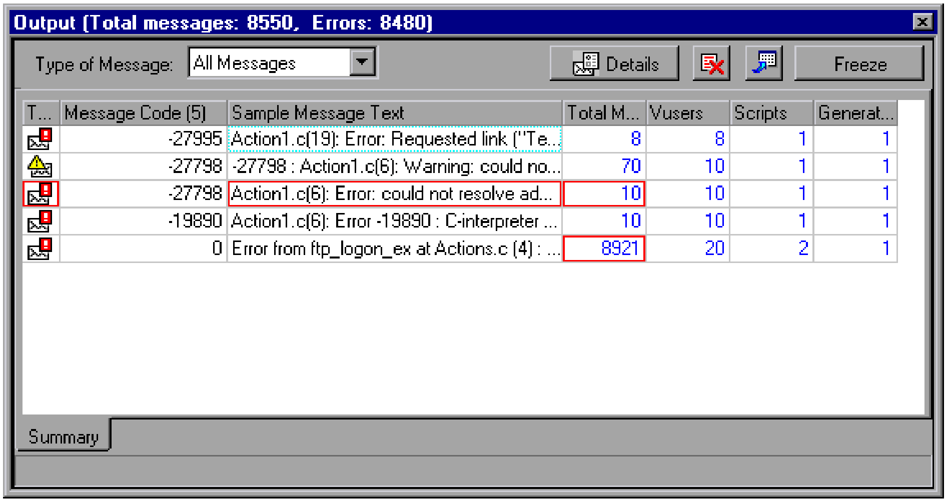
Time	Name	Machine	Measurement	Condition	Descripiton
03/11/2002 14:12:47	Alert_Set4		Error	value change over 15 ...	

The Alerts Output window displays the following information about each alert:

- **Time:** The date and time when the alert was triggered.
- **Name:** The name of the Alert as defined by the user.
- **Machine:** The machine that triggered the alert.
- **Measurement:** The measurement for which the alert was triggered.
- **Condition:** The alert condition.
- **Description:** A user-supplied description of the alert.

Viewing the Output Window

While the session runs, the Vusers and load generators send error, notification, warning, debug, and batch messages to the Console. You can view these messages in the Output window.



Output (Total messages: 8550, Errors: 8480)

Type of Message: All Messages

T...	Message Code (5)	Sample Message Text	Total M...	Vusers	Scripts	Generat...
❗	-27995	Action1.c(19): Error: Requested link ("Te...	8	8	1	1
⚠	-27798	-27798 : Action1.c(6): Warning: could no...	70	10	1	1
❗	-27798	Action1.c(6): Error: could not resolve ad...	10	10	1	1
❗	-19890	Action1.c(6): Error -19890 : C-interpreter ...	10	10	1	1
❗	0	Error from ftp_logon_ex at Actions.c (4) : ...	8921	20	2	1

The total number of messages received is displayed in the title bar. Note that ProTune clears the messages in the Output window at the start of each session execution, or when you reset a session.

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

The Output window provides the following information in the Summary tab:

Column	Description
TYPE	the type of message sent: Error, Notify, Warning, Debug, or Batch (each represented by a different icon) Note: Debug messages will only be sent if you enable the debugging feature in Tools > Options > Debug Information (Expert Mode). Batch messages will be sent instead of message boxes appearing in the Console, if you are using automation.
MESSAGE CODE	the code assigned to all similar messages. The number in parentheses indicates the number of different codes displayed in the Output window.
SAMPLE MESSAGE TEXT	an example of the text of a message with the specified code
TOTAL MESSAGES	the total number of sent messages with the specified code
VUSERS	the number of Vusers that generated messages with the specified code
SCRIPTS	the number of scripts whose execution caused messages with the specified code to be generated
GENERATORS	the number of load generators from which messages with the specified code were generated

You can view and manipulate the log information in the Summary tab using the following utilities:

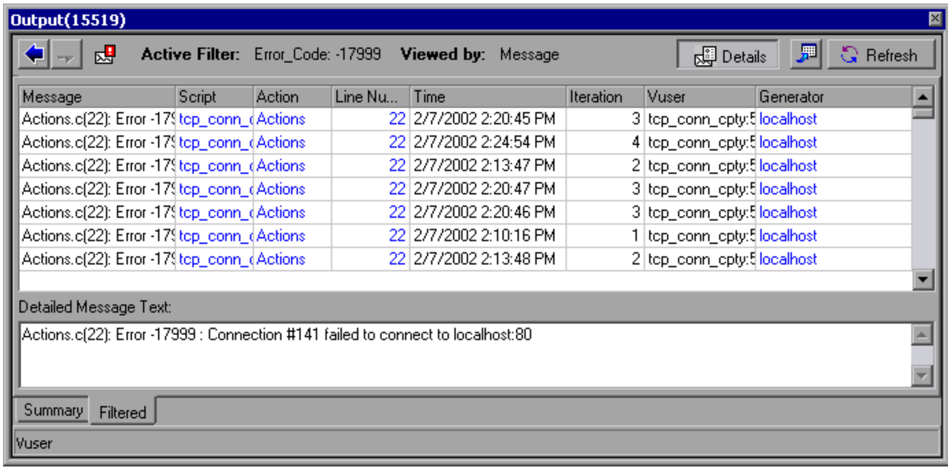
➤ To show (or hide) the Output window, choose **View** > **Show Output**.

- To sort the log information, click the appropriate column header. The messages are sorted in descending/ascending order.
- To filter the Output window to display only certain message types, select the type of message you want to view from the Type of Message box. By default, all types of output messages are displayed, unless you click the Show Snapshot button in the Session Status window.
- To view a message in detail, select the message and click the **Details** button. The Detailed Message Text box opens in the Output window displaying the complete message text.
- To save the Output window view to a file, click the **Export the view** button.
- To clear all log information from the Output window, click the **Remove all messages** button.
- To halt the updating of the Output window, click the **Freeze** button. To instruct ProTune to resume updating the Output window with messages, click the **Resume** button. Note that newly updated log information is displayed in a red frame.

Viewing Log Information Details

You can view details of each message, Vuser, script, and load generator associated with an error code by clicking the blue link in the respective column. The Output window displays a drilled down view by message, Vuser, script, or load generator in the Detailed tab.

For example, if you drill down on the Vusers column, the Output window displays all the messages with the code you selected, grouped by the Vusers that sent the messages.



Active Filter: Error_Code: -17999 **Viewed by:** Message

Message	Script	Action	Line Nu...	Time	Iteration	Vuser	Generator
Actions.c(22): Error -17	tcp_conn_	Actions	22	2/7/2002 2:20:45 PM	3	tcp_conn_cpty:5	localhost
Actions.c(22): Error -17	tcp_conn_	Actions	22	2/7/2002 2:24:54 PM	4	tcp_conn_cpty:5	localhost
Actions.c(22): Error -17	tcp_conn_	Actions	22	2/7/2002 2:13:47 PM	2	tcp_conn_cpty:5	localhost
Actions.c(22): Error -17	tcp_conn_	Actions	22	2/7/2002 2:20:47 PM	3	tcp_conn_cpty:5	localhost
Actions.c(22): Error -17	tcp_conn_	Actions	22	2/7/2002 2:20:46 PM	3	tcp_conn_cpty:5	localhost
Actions.c(22): Error -17	tcp_conn_	Actions	22	2/7/2002 2:10:16 PM	1	tcp_conn_cpty:5	localhost
Actions.c(22): Error -17	tcp_conn_	Actions	22	2/7/2002 2:13:48 PM	2	tcp_conn_cpty:5	localhost

Detailed Message Text:

```
Actions.c(22): Error -17999 : Connection #141 failed to connect to localhost:80
```

Note that the message type, the message code, and the column that you selected to drill down on, are displayed above the grid.

You can drill down further on the entries displayed in blue. Note that when you drill down on a Vuser, the Vuser log opens. When you drill down on a load generator, the Load Generators dialog box opens, displaying the load generator you selected. When you drill down on a script (or Action or Line Number), VuGen opens, displaying the script you selected.

Note: To limit the number of rows displayed when you drill down, open the *wlrun7.ini* file in any text editor, and located the following line:

```
MaxOutputUIRowsToShow=0
```

Change the 0 (no limit) to the number of rows you want to view.

When new messages arrive in the Output window, the Refresh button is enabled. Click **Refresh** to add the new log information to the Detailed tab view.

To move between the various drill down levels, click the **Previous view** and **Next view** buttons in the upper left-hand corner of the Output window.

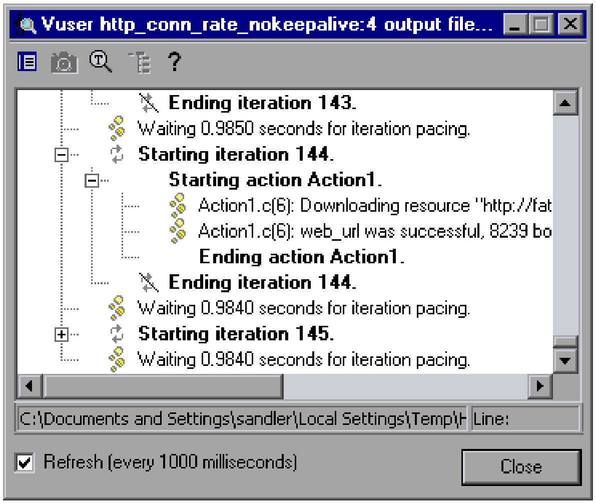
Viewing the Script Log

During session execution, you can view a log containing run-time information about each running Vuser.

To view the script log for a particular Vuser:

1 In the Vusers dialog box, select the Vuser whose log you want to view, and click the **Show Vuser Log** button, or right-click the Vuser and select **Show Vuser Log**.

The script log opens, displaying run-time information about the Vuser that is refreshed, by default, every 1000 milliseconds.



Vuser http_conn_rate_nokeepalive:4 output file...

```
Ending iteration 143.
Waiting 0.9850 seconds for iteration pacing.
Starting iteration 144.
    Starting action Action1.
        Action1.c(6): Downloading resource "http://fat
        Action1.c(6): web_url was successful, 8239 bo
        Ending action Action1.
    Ending iteration 144.
Waiting 0.9840 seconds for iteration pacing.
Starting iteration 145.
Waiting 0.9840 seconds for iteration pacing.
```

C:\Documents and Settings\sandler\Local Settings\Temp\H Line:

☑ Refresh (every 1000 milliseconds)

To change the default refresh settings, see “Options - Output Settings” on page 519.

Note: If you disabled the logging feature in the Run-Time Settings’ Log tab, the script log will be empty. If you selected the **Send messages only when an error occurs** option in the Log tab, the script log will contain output only if there are script errors.

- To disable the refreshing of this log, clear the **Refresh** check box.
- To view the information in text format, click the **Show Text View** button. To revert to the tree view, click the button again.
- If you are running a Web Vuser, and want to view a snapshot of the Web page where an error occurred, highlight the error in the Vuser log and click the **Display** button. Note that this option is only available for Vusers running on Windows load generators.

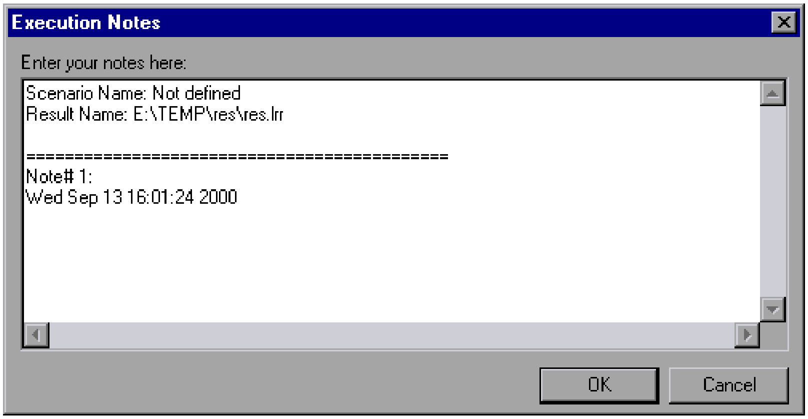
Note: In order to view a snapshot of the Web page where an error occurred, you must select the **Activate snapshot on error** option in the General tab of the Run-Time Settings dialog box before running the session.

- ➤ To search the Vuser log for specific text, click the **Find Text** button, and enter the text you want to search for in the text box.
- ➤ To collapse the tree view, click the **Collapse Node** button. To revert to the expanded tree view, click the same button again.

2 Click **Close** to close the script log.

Logging Execution Notes

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



Execution Notes

Enter your notes here:

```
Scenario Name: Not defined
Result Name: E:\TEMP\res\res.lrr

==========================================
Note# 1:
Wed Sep 13 16:01:24 2000
```

To log execution notes:

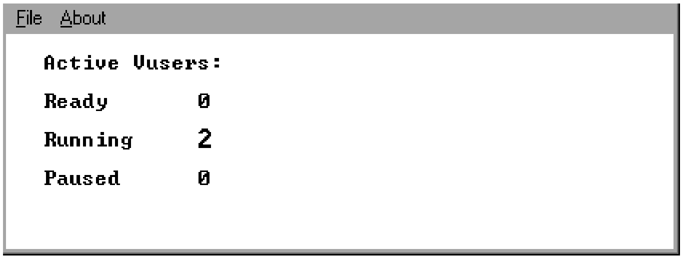
1 Select **Session** > **Execution Notes**. The Execution Notes dialog box opens.

2 Enter the notes that you want to log.

3 Click **OK** to close the dialog box. ProTune saves the notes you recorded.

Viewing the Agent Summary

When you run a session with non-GUI Vusers, the machine running the Vusers invokes an agent that controls the Vuser execution on that load generator. To view the Agent window during session execution, double-click the Agent icon on the task bar. ProTune displays a summary of the Ready, Running, and Paused Vusers.



```
Active Vusers:
Ready        0
Running      2
Paused       0
```

12

Working with Firewalls

You can run Vusers and monitor your servers, while the Console is outside of the firewall.

This chapter describes:

- Overview of Running or Monitoring over the Firewall
- Configuring the ProTune Agents in LAN1
- Configuring the Firewall to Allow Agent Access
- Installing and Configuring the MI_Listener in LAN2
- Configuring the Console to Run Vusers or Monitor over the Firewall

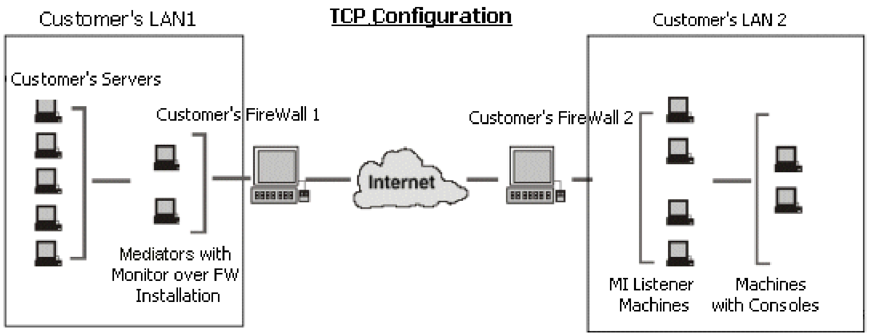
About Using Firewalls in ProTune

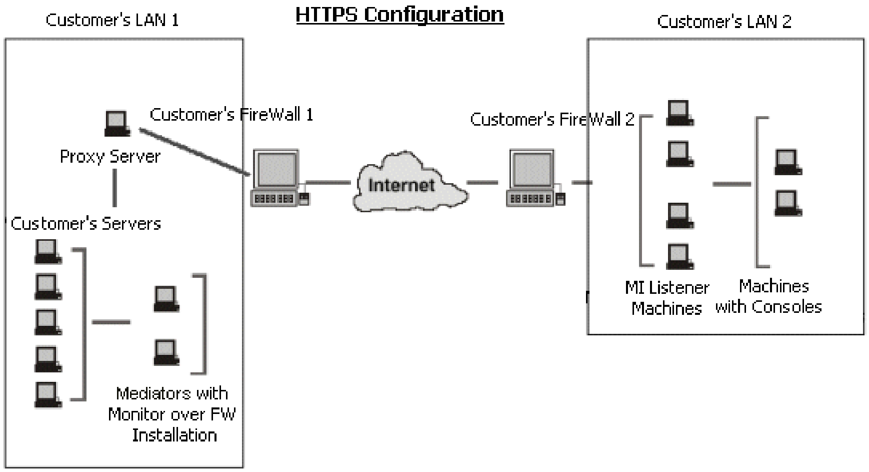
Working with a firewall means that you can prevent access to the outside world and from the outside world, on specific port numbers.

For example, you can specify that there is no access to any port from the outside world, with the exception of the mail port (23), or you can specify that there is no outside connection to any ports except for the mail port and WEB port (80). The port settings are configured by the system administrator.

In a regular ProTune session (not over the firewall), the Console has direct access to the ProTune agents running on remote machines. This enables the Console to connect directly to those machines. However, when running Vusers or monitoring servers over the firewall, this direct connection is blocked due to the firewall. The connection cannot be initiated by the Console, because it does not have permissions to make an opening in the firewall.

Configure your system according to one of the following configurations to run Vusers or monitor servers over the firewall. Note that these configurations contain a firewall on each LAN. There may also be configurations where there is a firewall only for LAN1:





During installation, the ProTune agent is added either as a Windows service or as an executable run from the Startup folder. The MI Listener component serves as a router between the Console and the ProTune agent.

In the above configuration, the MI Listener is on a different machine than the Console. Every ProTune agent can behave as a MI Listener, so you can use the Console machine as the MI Listener also, and you do not need a separate installation.

TCP Configuration

The TCP configuration requires every ProTune agent machine behind the FireWall 1 to be allowed to open a port in the firewall for outgoing communication. If this is the firewall configuration at hand, use the TCP configuration.

HTTPS Configuration

In the HTTPS configuration, only one machine (the proxy server) is allowed to open a port in the firewall. Therefore it is necessary to tunnel all outgoing communications through the proxy server.

After installation, you configure the ProTune agent to operate over the firewall. You also modify firewall settings to enable communication between the agent machine(s) inside the firewall and machines outside the firewall. In addition, you prepare the Console to work over the firewall.

Refer to Chapter 14, “Monitoring over a Firewall” for additional information about configuring ProTune to monitor servers from outside the firewall.

Overview of Running or Monitoring over the Firewall

To prepare for running Vusers or monitoring servers over the firewall, perform the following steps:

1. **Make sure that the ProTune agent is installed on the machines running Vusers, or on the servers to be monitored behind the firewall.**

   The agents can run on Windows or Unix machines. See the diagram, “About Using Firewalls in ProTune,” on page 143.

2. **Configure the ProTune agent to operate over the firewall.**

   Configure the ProTune agent on the machines running Vusers, or agents acting as mediators for the servers to be monitored. See “Configuring the ProTune Agents in LAN1,” on page 146 for instructions.

3. **Configure the firewall(s).**

   Configure the firewall, to allow communication between the agents inside the firewall, and the machines outside the firewall. See “Configuring the Firewall to Allow Agent Access” on page 154.

4 Install the Monitoring over Firewall Component.

To monitor a server over the firewall, install this component on the machine which sits inside the firewall, and acts as a mediator between the Console, and the monitored server. See the diagram, "About Using Firewalls in ProTune," on page 143 for information about where to install the Monitoring over the Firewall component, and refer to the *ProTune Installation Guide* for installation instructions.

5 Install the MI Listener on a machine outside the firewall.

See the diagram, "About Using Firewalls in ProTune," on page 143 for information about where to install the MI Listener, and refer to the *ProTune Installation Guide* for installation instructions.

6 Configure the MI Listener machines.

Configure the security attributes on each MI Listener machine. See "Installing and Configuring the MI_Listener in LAN2," on page 154.

7 Configure the Console machine.

Configure the Console machine to recognize the agent and MI Listener machines. See "Configuring the Console to Run Vusers or Monitor over the Firewall," on page 156.

Configuring the ProTune Agents in LAN1

The machines within LAN1 can either be Load Generator machines running Vusers, or mediator machines connected to the servers to be monitored by the Console. You configure the ProTune agents in LAN1 to operate over firewall. The Console machine resides outside the firewall, and LAN1 is inside the firewall.

Configuring and Running the Windows ProTune Agent

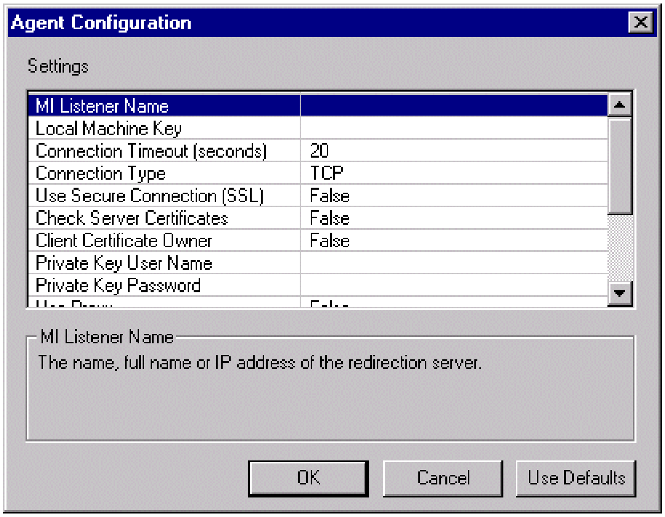
To configure the ProTune agents on Windows machines:

1 Stop the ProTune agent by right-clicking its icon in the system tray and selecting **Close**.

2 Select **Agent Settings** from Start > Programs > ProTune > Advanced Settings (or open *<ProTune root>\launch_service\dat\br_lnch_server.cfg* in a text editor).

3 In the Firewall section set FireWallServiceActive to 1, and save your changes.

4 Run **Agent Configuration** from Start > Programs > ProTune > Advanced Settings, or run *<ProTune root>\launch_service\bin\AgnetConfig.exe*.



Agent Configuration

Settings

MI Listener Name	
Local Machine Key	
Connection Timeout (seconds)	20
Connection Type	TCP
Use Secure Connection (SSL)	False
Check Server Certificates	False
Client Certificate Owner	False
Private Key User Name	
Private Key Password	

MI Listener Name

The name, full name or IP address of the redirection server.

5 Set each option as described in “Agent Configuration Settings” on page 152.

6 Click **OK** to save your changes, **Cancel** to cancel them, or **Use Defaults**.

7 Restart the ProTune agent by double-clicking the shortcut on the desktop, or from Start > Programs > ProTune > ProTune Agent Service/Process.

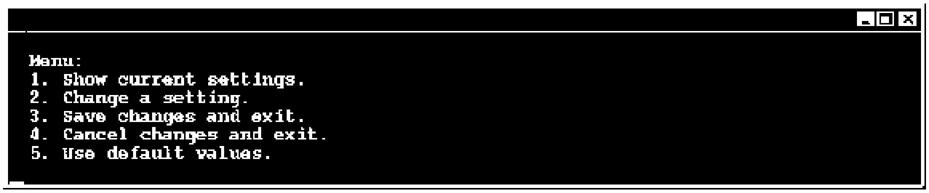
Configuring and Running the UNIX ProTune Agent

To configure the ProTune agents on UNIX machines:

1 Open *<ProTune root folder>/dat/br_lnch_server.cfg* in a text editor.

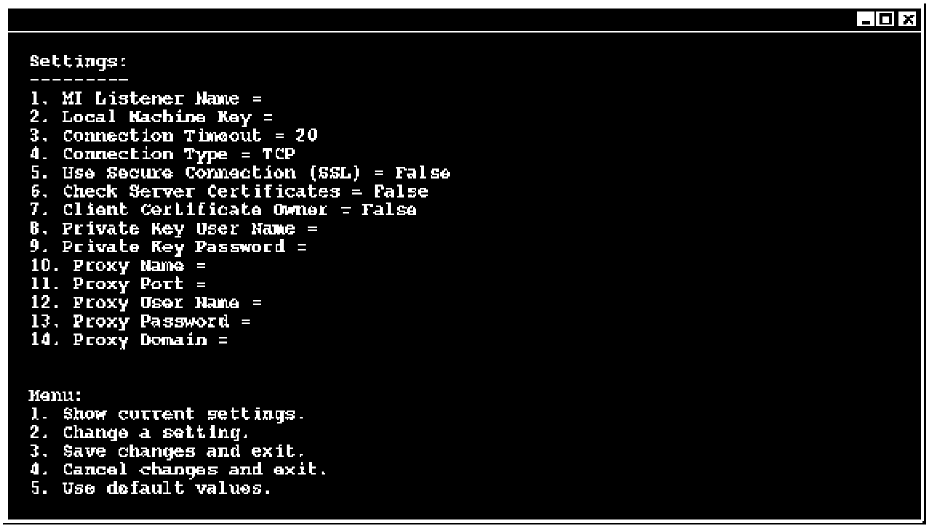
2 In the Firewall section, set FireWallServiceActive to 1 and save your changes.

3 Run the *agent_config* from the *<ProTune root folder>*/bin directory to display the following menu:



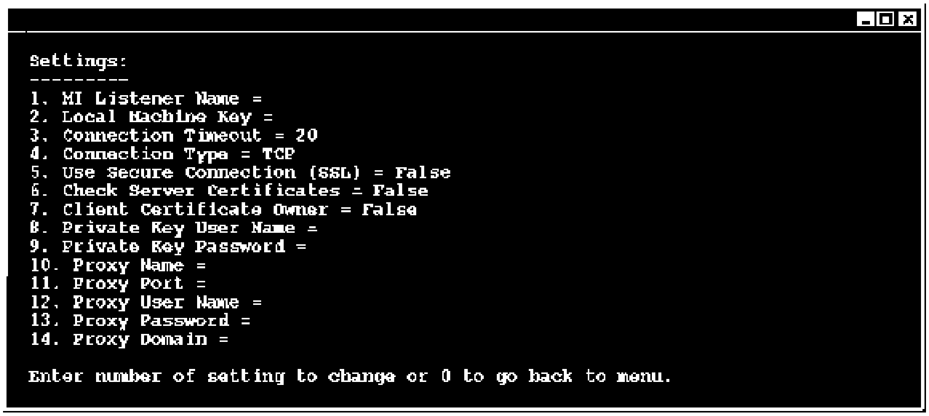
```
Menu:
1. Show current settings.
2. Change a setting.
3. Save changes and exit.
4. Cancel changes and exit.
5. Use default values.
_
```

4 Enter 1 to display the current settings:



```
Settings:
---------
1. MI Listener Name =
2. Local Machine Key =
3. Connection Timeout = 20
4. Connection Type = TCP
5. Use Secure Connection (SSL) = False
6. Check Server Certificates = False
7. Client Certificate Owner = False
8. Private Key User Name =
9. Private Key Password =
10. Proxy Name =
11. Proxy Port =
12. Proxy User Name =
13. Proxy Password =
14. Proxy Domain =


Menu:
1. Show current settings.
2. Change a setting.
3. Save changes and exit.
4. Cancel changes and exit.
5. Use default values.
```

5 To change a setting, enter 2 to display the settings menu:



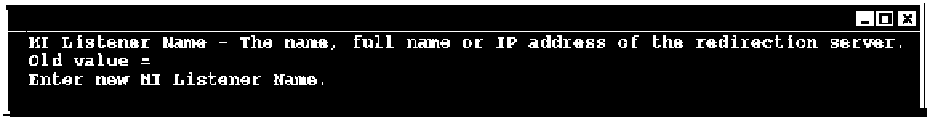
```
Settings:
---------
1. MI Listener Name =
2. Local Machine Key =
3. Connection Timeout = 20
4. Connection Type = TCP
5. Use Secure Connection (SSL) = False
6. Check Server Certificates = False
7. Client Certificate Owner = False
8. Private Key User Name =
9. Private Key Password =
10. Proxy Name =
11. Proxy Port =
12. Proxy User Name =
13. Proxy Password =
14. Proxy Domain =

Enter number of setting to change or 0 to go back to menu.
```

Enter the setting and continue according to the menu instructions. Set each option according to the "Agent Configuration Settings" on page 152.

Examples of Changing Agent Settings in Unix

To change the MI Listener Name:

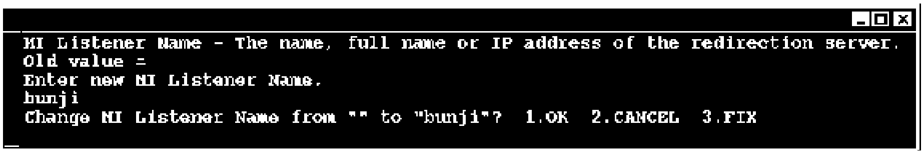
1 Enter 1 in the Settings menu to display the following screen:



```
MI Listener Name - The name, full name or IP address of the redirection server.
Old value =
Enter new MI Listener Name.
```

Line one is a description of the setting. Line two shows the current value of the setting.

2 Enter the new value, (For example, 'bunji') to display the following:



```
MI Listener Name - The name, full name or IP address of the redirection server.
Old value =
Enter new MI Listener Name.
bunji
Change MI Listener Name from "" to "bunji"?  1.OK  2.CANCEL  3.FIX
_
```

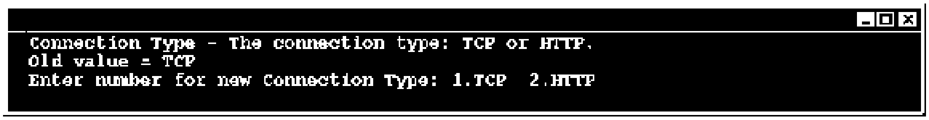
3 To keep the new value and return to the menu, enter '1'.

To discard the new value and return to the menu, enter '2'.

To discard the new value and change the setting once more, enter '3'.

To change the Connection Type:

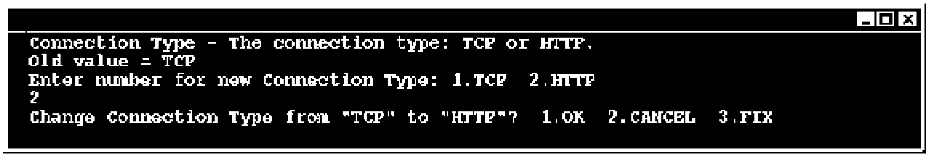
1 Enter 1 in the Settings menu to display the following screen:



```
Connection Type - The connection type: TCP or HTTP.
Old value = TCP
Enter number for new Connection Type: 1.TCP  2.HTTP
```

Line one is a description of the setting. Line two shows the current value of the setting.

2 Enter 1 to set the connection type to TCP, or enter 2 to set it to HTTP and display the following:



```
Connection Type - The connection type: TCP or HTTP.
Old value = TCP
Enter number for new Connection Type: 1.TCP  2.HTTP
2
Change Connection Type from "TCP" to "HTTP"?  1.OK  2.CANCEL  3.FIX
```

3 To keep the new value and return to the menu, enter '1'.

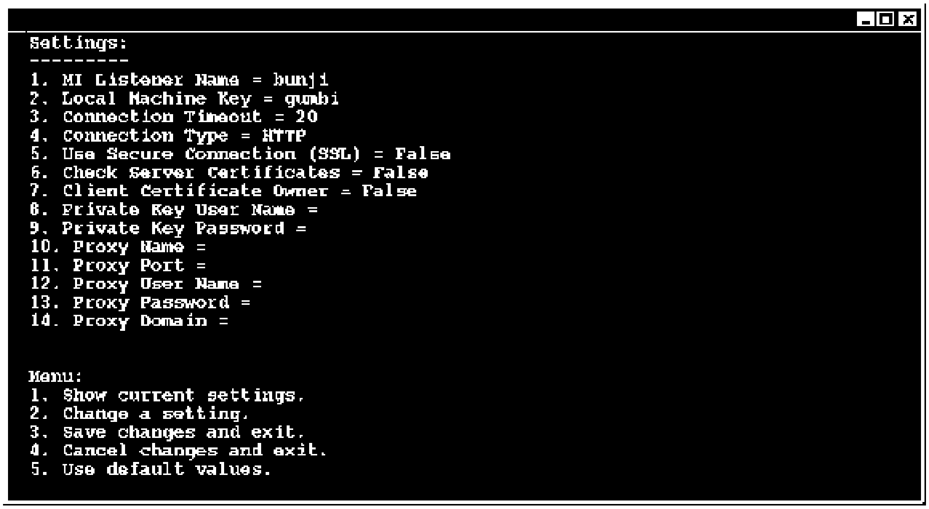
To discard the new value and return to the menu, enter '2'.

To discard the new value and change the setting once more, enter '3'.

Viewing the Settings and Restarting the Agent

To view the current settings:

1. Return to the main menu by entering 1.
2. Enter 1 to display the settings. The following example includes the new settings for MI Listener Name and Connection Type:



```
Settings:
---------
1. MI Listener Name = bunji
2. Local Machine Key = gumbi
3. Connection Timeout = 20
4. Connection Type = HTTP
5. Use Secure Connection (SSL) = False
6. Check Server Certificates = False
7. Client Certificate Owner = False
8. Private Key User Name =
9. Private Key Password =
10. Proxy Name =
11. Proxy Port =
12. Proxy User Name =
13. Proxy Password =
14. Proxy Domain =


Menu:
1. Show current settings.
2. Change a setting.
3. Save changes and exit.
4. Cancel changes and exit.
5. Use default values.
```

3. To save your changes, enter 3 from the main menu.

   To cancel your changes, enter 4.

   To use the default values supplied by ProTune (as described in “Agent Configuration Settings,” on page 152), enter 5.

To start or remove the ProTune agent:

1. To start the ProTune agent, run the command 'm_daemon_setup -install' from the <ProTune root folder>/bin directory.
2. To remove the ProTune agent, run the command 'm_daemon_setup -remove' from the <ProTune root folder>/bin directory.

For more information about running the ProTune agent, refer to “UNIX Shell” in Appendix A, “Troubleshooting the Console.”

Agent Configuration Settings

Option	Default Value	Description
MI Listener name	none	The name, full name or IP address of the Mercury Interactive listener machine, MI Listener.
Local Machine Key	none	A string identifier used to establish a unique connection between the Console host and the agent machine, via the MI Listener machine.
Connection Timeout (seconds)	20 seconds	The length of time you want the agent to wait before retrying to connect to the MI Listener machine. If zero, the connection is kept open from the time the agent is run.
Connection Type	TCP	Choose either TCP (default) or HTTP, depending on the configuration you are using.
Use Secure Connection (SSL)	False	Choose True to connect using the Secure Sockets Layer protocol.
Check Server Certificates	False	Choose True to authenticate SSL certificates that are sent by servers. This option is relevant only if the **Use Secure Connection** option is set to **True**.
Client Certificate Owner	False	Choose True to load the SSL certificate. In some cases the server requests a certificate to allow the connection to be made. This option is relevant only if the **Use Secure Connection** option is set to **True**.

Option	**Default Value**	**Description**
Private Key User Name	None	The user name that may be required during the SLL certificate authentication process. This option is relevant only if the **Client Certificate Owner** option is set to **True**.
Private Key Password	None	The password that may be required during the SLL certificate authentication process. This option is relevant only if the **Client Certificate Owner** option is set to **True**.
Proxy Name	<IE proxy server name> or None	The name of the proxy server. This option is mandatory if the **Connection Type** option is set to **HTTP.**
Proxy Port	<IE proxy server port> or None	The proxy server connection port.This option is mandatory if the **Connection Type** option is set to **HTTP**.
Proxy User Name	None	The username of a user with connection rights to the proxy server.
Proxy Password	None	The user's password.
Proxy Domain	None	The user's domain if defined in the proxy server configuration. This option is required only if NTLM is used.

Configuring the Firewall to Allow Agent Access

You modify your firewall settings to enable communication between the machine(s) inside the firewall and machines outside the firewall.

TCP configuration:

The ProTune agent tries to establish a connection with MI Listener using port 443 at an interval of seconds specified in the Connection Timeout field in the agent configuration. To enable this connection, allow an outgoing connection for HTTPS service on the FireWall for port 443. As a result, the agent connects to MI Listener and MI Listener connects back to the agent. From this point on, the agent listens to commands from the MI Listener.

HTTPS configuration:

The ProTune agent tries to establish a connection with MI Listener using the proxy port specified in the Proxy Port field and at an interval of seconds specified in the Connection Timeout field in the agent configuration. On successful connection, the proxy server connects to MI Listener. To enable this connection, allow an outgoing connection for HTTPS service on the FireWall for port 443. As a result, the proxy server connects to MI Listener and MI Listener connects back to the agent through the proxy server. From this point on, the agent listens to commands from the MI Listener.

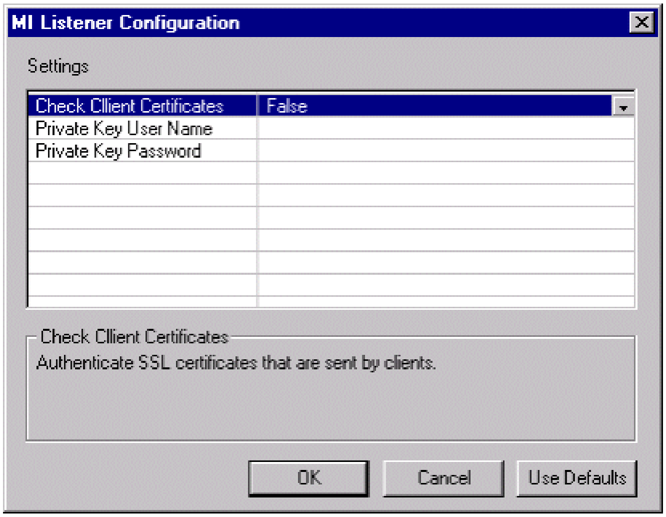
Installing and Configuring the MI_Listener in LAN2

To enable running Vusers or monitoring over a firewall, you need to install MI Listener on one or more machines in your LAN2. For instructions, refer to the *ProTune Installation Guide*.

To configure the MI Listener:

1. Open incoming HTTPS service for port 443.
2. Stop the ProTune agent by right-clicking its icon in the system tray and selecting **Close** from the popup menu.

3 Run **MI Listener Configuration** from Start > Programs > ProTune > Advanced Settings, or run *<ProTune root folder>\launch_service\bin\MILsnConfig.exe*.



MI Listener Configuration

Settings

Check Client Certificates	False
Private Key User Name	
Private Key Password	

Check Client Certificates

Authenticate SSL certificates that are sent by clients.

4 Set each option as described in “MI Listener Configuration Settings,” on page 156.

5 Click **OK** to save your changes, **Cancel** to cancel them, or **Use Defaults**.

6 Restart the ProTune agent by double-clicking the shortcut on the desktop, or running it from Start > Programs > ProTune.

7 Make sure that port 433 is free on the MI Listener machine.

MI Listener Configuration Settings

Option	Default Value	Description
Check Client Certificates	False	Choose True to request that the client send an SSL certificate when connecting, and to authenticate the certificate.
Private Key User Name	None	The user name that may be required during the SLL certificate authentication process.
Private Key Password	None	The password that may be required during the SLL certificate authentication process.

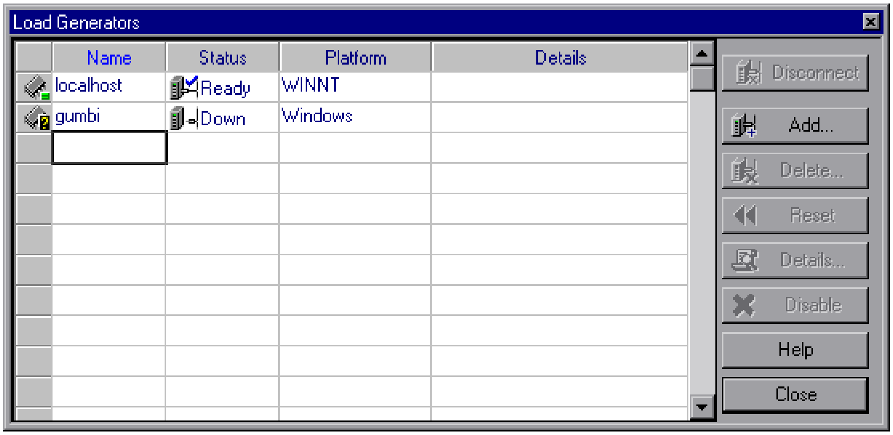
Configuring the Console to Run Vusers or Monitor over the Firewall

In order to obtain information for the monitors configured inside the firewall, or to run Vusers inside the firewall, you create a unique connection between the Console and the agent machine, via the Mercury Interactive listener machine, MI Listener. You establish this connection by defining the agent machine as a load generator.

To configure the Console for running vusers or monitoring over the firewall:

1. Run the Console from Start > Programs > ProTune and create a new session, or load an existing one.
2. Click **Generators** to display the Load Generators window. In the Name field, enter the symbolic name of the server. This is the same name that you entered in the Local Machine Key setting in the Agent Configuration. In the example below, the server name is *gumbi*.

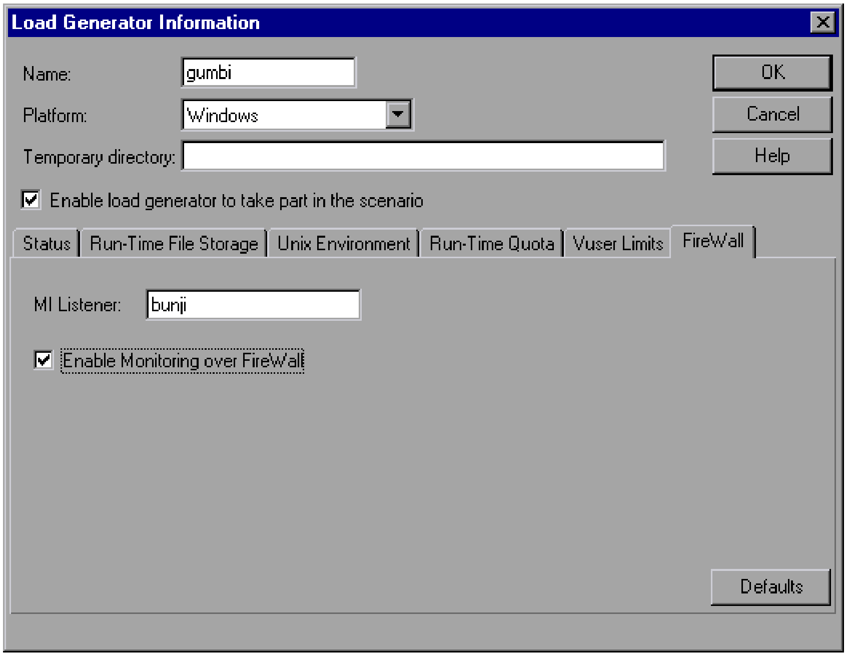
If the server is a UNIX server, change the Platform field to *UNIX*.



Load Generators

Name	Status	Platform	Details
localhost	Ready	WINNT	
gumbi	Down	Windows	

3 Select the Load Generator, and click **Details** to display the Load Generator Information.



Load Generator Information

Name: gumbi

Platform: Windows

Temporary directory:

☑ Enable load generator to take part in the scenario

MI Listener: bunji

☑ Enable Monitoring over FireWall

4 In the Firewall tab, enter the MI Listener machine's name in the **MI Listener** field. This is the same name that you entered in the Agent Configuration, in the MI Listener Name setting.

5 To run Vusers over the Firewall, make sure to clear the **Enable Monitoring over Firewall** option.

6 To monitor over the firewall, select the **Enable Monitoring over Firewall** option.

7 Click **OK** to return to the Load Generators dialog box.

8 Select the Load Generator and click **Connect**.

Part V

Monitoring a Session

13

Online Monitoring

You can monitor session step execution using the ProTune online run-time, transaction, Web resource, system resource, network delay, firewall server resource, Web server resource, Web application server resource, database server resource, streaming media resource, ERP server resource, and Java performance monitors.

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

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

About Online Monitoring

ProTune provides the following online monitors:

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

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

The **Web Resource** monitor measures statistics at the Web server(s) during session step runs. It provides information about the number of Web connections, throughput volume, HTTP responses, server retries, and downloaded pages during the session step. For more information on the Web Resource monitor, see Chapter 16, "Web Resource Monitoring."

The **System Resource** monitors gauge the Windows, UNIX, TUXEDO, SNMP, and Antara FlameThrower resources used during a session step. To activate the System Resource monitors, you must set the monitor options before you run your session step. For information on setting these options, see Chapter 17, "System Resource Monitoring."

The **Network Delay** monitor displays information about the network delays on your system. To activate the Network Delay monitor, you must set up the network paths to monitor before you run your session step. For more information see Chapter 18, "Network Monitoring."

The **Firewall** monitor measures statistics at the firewall servers during the session step. To activate the Firewall monitor, you must set up a list of resources to monitor before you run your session step. For more information, see Chapter 19, "Firewall Server Performance Monitoring."

The **Web Server Resource** monitors measure statistics at the Apache, Microsoft IIS, and iPlanet/Netscape Web servers during the session step. To activate the Web Server Resource monitors, you must set up a list of resources to monitor before you run your session step. For more information, see Chapter 20, "Web Server Resource Monitoring."

The **Web Application Server Resource** monitors measure statistics at the Web application server(s) during the session step. To activate the Web Application Server Resource monitors, you must set up a list of resources to monitor before you run your session step. For more information, see Chapter 21, "Web Application Server Resource Monitoring."

The **Database Server Resource** monitors measure statistics related to the SQL server, Oracle, Sybase, and DB2 databases. To activate the Database Server Resource monitors, you must set up a list of measurements to monitor before you run your session step. For more information, see Chapter 22, "Database Resource Monitoring."

The **Streaming Media** monitors measure statistics at the Windows Media Server and RealPlayer audio/video servers, as well as the RealPlayer client. To activate the Streaming Media monitors, you must set up a list of resources to monitor before you run your session step. For more information, see Chapter 23, "Streaming Media Monitoring."

The **ERP Server Resource** monitor measures statistics at the ERP servers during the session step. To activate the ERP Server Resource monitor, you must set up a list of resources to monitor before you run your session step. For more information, see Chapter 24, "ERP Server Resource Monitoring."

The **Java Performance** monitors measure statistics of Enterprise Java Bean (EJB) objects and Java-based applications, using EJB, JProbe, Sitraka JMonitor, and TowerJ Java virtual machines. To activate the Java Performance monitors, you must set up lists of resources to monitor before you run your session step. For more information, see Chapter 25, "Java Performance Monitoring."

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

Note: For a detailed list of ProTune's monitors, see Mercury Interactive's Web site (http://www-heva.mercuryinteractive.com/resources/library/technical/loadtesting_monitors/supported.html).

Choosing Monitors and Measurements

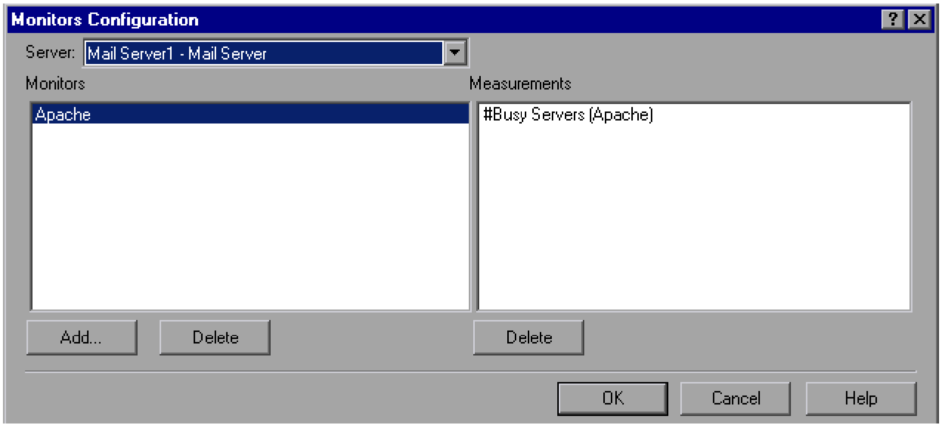
You can select the measurements to monitor for each of your servers via the Monitors button on the main toolbar or the Element Monitors tab in the System Topology window.

To select measurements to monitor via the System Topology window:

➤ Follow the directions in "Selecting Monitors," on page 26.

To select measurements to monitor via the Monitors button on the main toolbar:

1 Click the **Monitors** button. The **Monitors Configuration** window is displayed:



Monitors Configuration

Server: Mail Server1 - Mail Server

Monitors: Apache

Measurements: #Busy Servers (Apache)

2 Select the server whose monitors you want to configure from the list box. The monitors that are currently assigned to monitor the specified server are displayed in the **Monitors** pane. When you click on a monitor in the **Monitors** pane, the measurements that have been specified to be monitored by that monitor are listed in the **Measurements** pane.

3 To add measurements to monitor, click **Add**. The **Select Measurements to Monitor** dialog box is displayed. Choose the monitor and the measurements for the specific server.

Note: For detailed instructions on using the **Select Measurements to Monitor** dialog box, see “Selecting Monitors,” on page 26.

To delete a monitor from the **Monitors** pane, select the monitor and click the left **Delete** button.

To delete a measurement from the list in the **Measurements** pane, click the right **Delete** button.

4 Click **OK** to save your configuration.

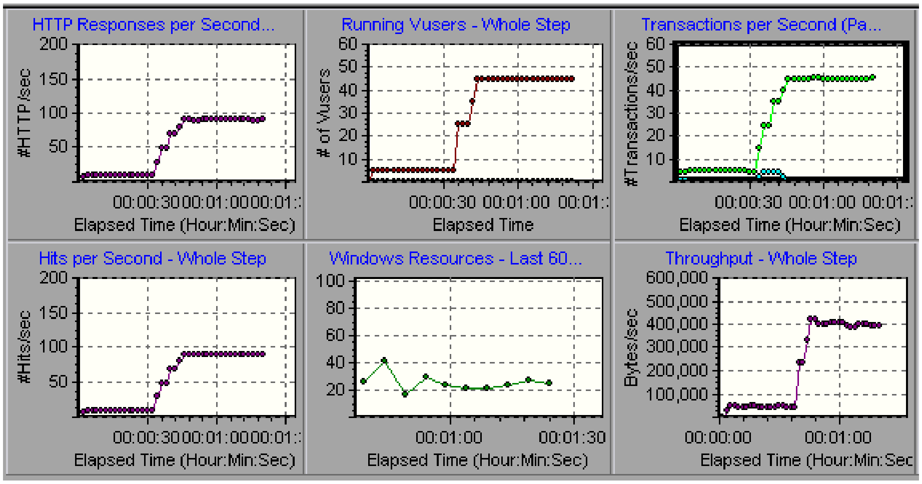
Starting the Monitors

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

To start the online monitors:

1 Start the session step. Select the Vuser groups you want to run and click the **Start Session** button, or choose **Session Step** > **Start**.

2 Click the **Execute** tab. The default graphs are displayed below the Session Step Groups window.



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

4 Click the "+" in the left pane to expand the graph tree. To hide the graph tree view, select **View** > **Hide Available Graphs**, or click the **X** button in the right-hand corner of the Available Graphs list.

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

Note: The Transaction Monitor graphs will not contain any data unless transactions are being executed. In addition, the System Resource, Network, Firewall, Web Server, Web Application Server, Database, Streaming Media, ERP Resource, and Java Performance graphs will not contain any data unless you set up a list of resources to monitor before running your session step.

Configuring Online Monitors

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

Monitor options: global sampling rate, error handling, debugging, and the frequency settings. For more information, see "Setting Monitor Options" on page 168.

Graph properties: refresh rate, display type, graph time for the x-axis, and the y-axis scale. For more information, see "Configuring Online Graphs" on page 170.

Measurement settings: line color, scale of the y-axis, and whether to show or hide the line. For more information, see "Configuring Online Measurements" on page 177.

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

Setting Monitor Options

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

- **Sampling Rate:** The sampling rate is the period of time (in seconds) between consecutive samples. By default, the online monitor samples the data at intervals of three seconds. If you increase the sampling rate, the data is monitored less frequently. This setting applies to all graphs. To set a sampling rate for a specific graph, see "Configuring Online Graphs" on page 170.

  The sampling rate you set is applied to all server monitors that you subsequently activate. It is not applied to server monitors that have already been activated. To apply the new sampling rate to activated server monitors, save your session step and reopen it.

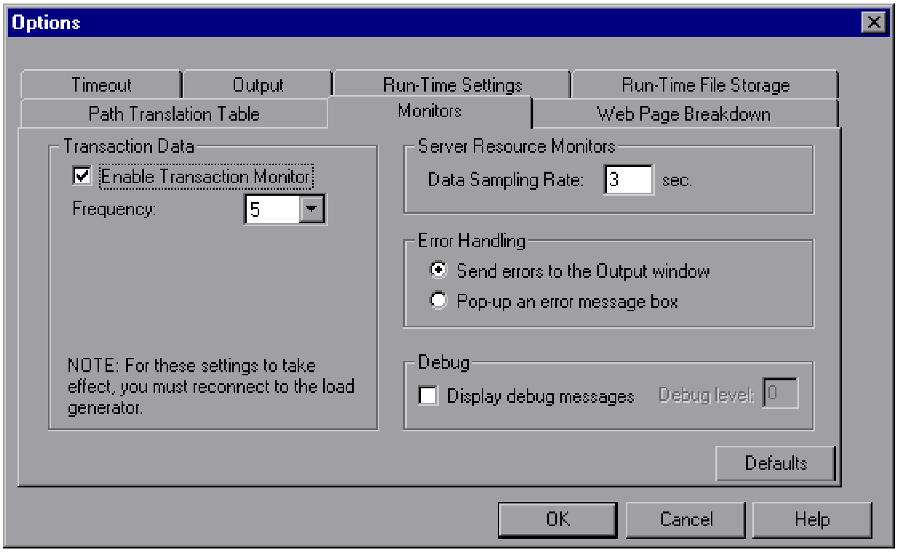
Note: Each monitor has a different minimum sampling rate. If the default sampling rate, or the rate set in the Options Monitors tab is less than a monitor's minimum sampling rate, the monitor will sample data at its minimum sampling rate. For example, the minimum sampling rate for the Oracle Monitor is 10 seconds. If the sampling rate in the Options Monitors tab is set at less than 10 seconds, the Oracle Monitor will continue to monitor data at 10 second intervals.

- **Error Handling**: You indicate how ProTune should behave when a monitor error occurs—issue a popup message box or send error messages to the Output window (default).
- **Debug:** The online monitor provides debugging capabilities. You can display the debug messages in the output log. For the Network monitor, you can indicate the debug (detail) level of messages sent to the log, ranging from 1-9.
- **Frequency:** You set the frequency at which the monitor sends updates to the Console for the Transaction, Data Point, and Web Resource graphs. The data is averaged for the frequency period defined, and only one value is sent to the Console.

For information on enabling and disabling the Transaction monitor and Web page breakdown, see Chapter 15, “Run-Time and Transaction Monitoring.”

To set monitor options:

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



Options

Transaction Data

☑ Enable Transaction Monitor

Frequency: 5

NOTE: For these settings to take effect, you must reconnect to the load generator.

Server Resource Monitors

Data Sampling Rate: 3 sec.

Error Handling

● Send errors to the Output window

○ Pop-up an error message box

Debug

□ Display debug messages Debug level: 0

2 Specify the frequency at which the monitor should send updates to the Console for the Transaction, Data Point, and Web Resource graphs. The default value is 5 seconds. For a small session step, it is recommended that you use a frequency of 1. For a large session step, it is recommended that you use a frequency of 3-5. The higher the frequency, the less network traffic there will be.

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

---

3 Enter a sampling rate.

4. Set the desired **Error Handling** option.
5. To display debug messages in the Output window, select the **Display Debug Messages** check box. For the Network monitor, specify a **Debug level** from 1-9.
6. Click **OK** to save your settings and close the Options dialog box.

You can configure an additional monitor setting while working in Expert mode. For information on working in Expert mode, see Appendix C, "Working in Expert Mode."

Configuring Online Graphs

You can customize your graph in the following areas:

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

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

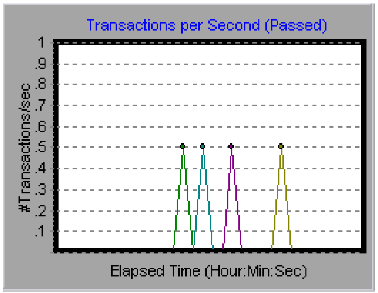
Refresh Rate

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

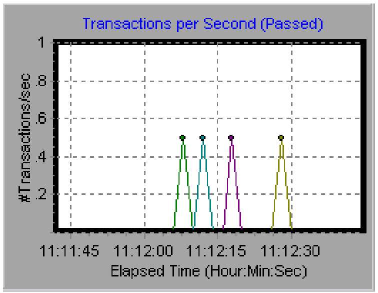
Note: In a large load test, it is recommended to use a refresh rate of three to five seconds. This enables you to avoid problems with CPU resource usage.

X-Axis Style

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



Don't Show



Clock Time

Graph Time

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

Display Type

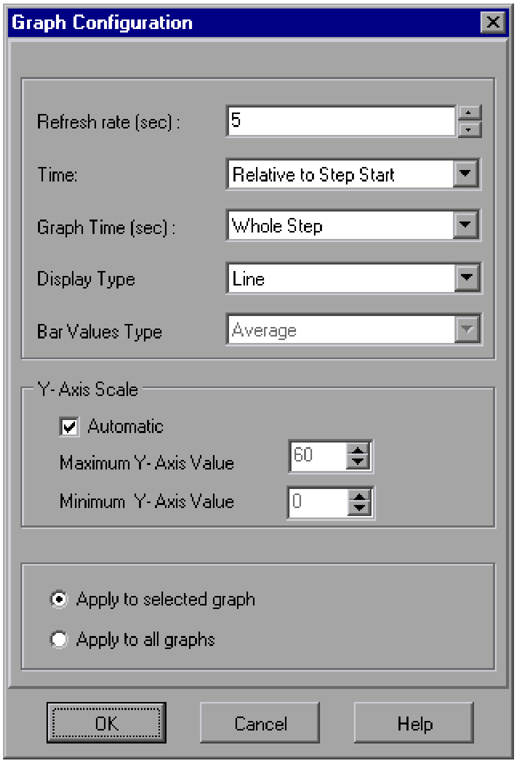
You can specify whether ProTune displays the Network Delay Time graph as a line, pie, or area graph. By default, the graph is displayed as a line graph. Note that all other graphs can only be displayed as line graphs.

Y-Axis Style

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

To customize your graphs:

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



Graph Configuration

Refresh rate (sec) : 5

Time: Relative to Step Start

Graph Time (sec) : Whole Step

Display Type Line

Bar Values Type Average

Y- Axis Scale

☑ Automatic

Maximum Y- Axis Value 60

Minimum Y- Axis Value 0

● Apply to selected graph

○ Apply to all graphs

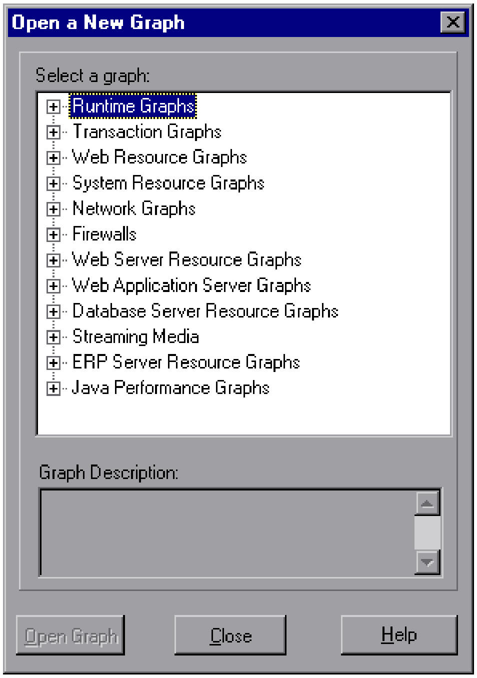
2. To apply the dialog box settings to all graphs, select **Apply to all graphs**.
3. Enter the desired refresh rate—the time between graph updates—in the Refresh Rate box.
4. Select a style for the x-axis from the Time box.
5. Select a value from the Graph Time box. The graph time is the time in seconds displayed by the x-axis.
6. For the Network Delay Time graph, select a graph style—Line, Pie, or Area—from the Display Type box.
7. If the selected display type is Bar, choose a value from the Bar Values Type box. This determines the type of value that will be displayed in the bar graph. You can choose between **Average**, **Last Value**, **Minimum** and **Maximum**.
8. Select a maximum or minimum value for the y-axis, or choose **Automatic** to view graphs using the default y-axis scale.
9. Click **OK** to save your settings and close the Graph Configuration dialog box.

Opening Online Monitor Graphs

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

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

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



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

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

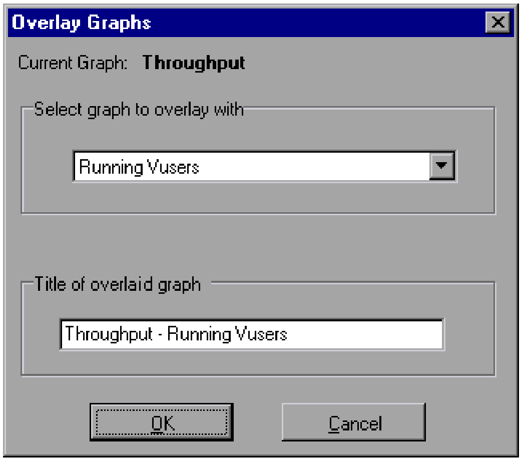
Merging Graphs

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

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

To overlay two graphs:

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


Overlay Graphs

Current Graph: **Throughput**

Select graph to overlay with: Running Vusers

Title of overlaid graph: Throughput - Running Vusers

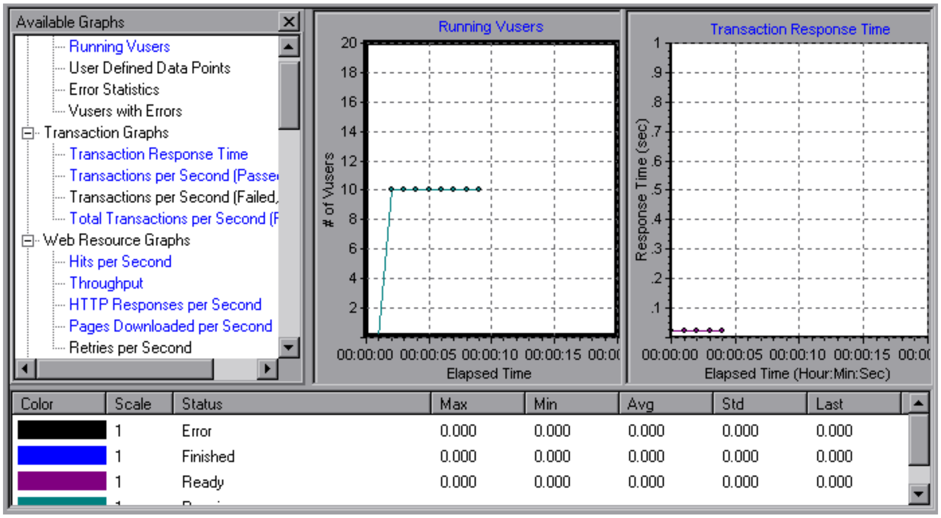
2 Select a graph with which you want to overlay the current graph. The drop-down list only shows the active graphs that have a common x-axis with the current graph.

3 Enter a title for the overlaid graph.

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

Understanding Online Monitor Graphs

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



Color	Scale	Status	Max	Min	Avg	Std	Last
⬛	1	Error	0.000	0.000	0.000	0.000	0.000
🟦	1	Finished	0.000	0.000	0.000	0.000	0.000
🟪	1	Ready	0.000	0.000	0.000	0.000	0.000

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

Note: In a goal-oriented session step, the goal you defined is also displayed in the appropriate graph.

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

To focus on a particular line, you can:

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

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

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

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

Configuring Online Measurements

You can configure the following online measurement settings:

- Changing Line Colors
- Setting the Scale of the Measurement
- Showing and Hiding Transactions

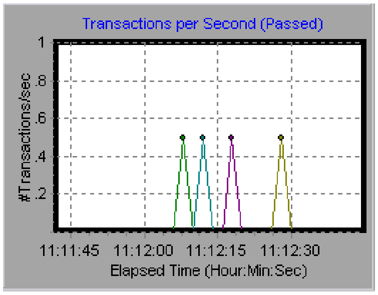
Changing Line Colors

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

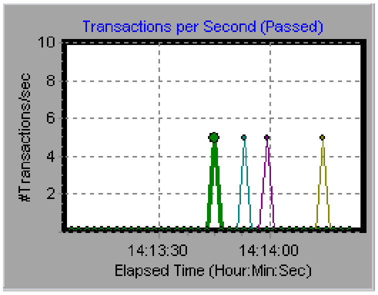
Setting the Scale of the Measurement

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

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



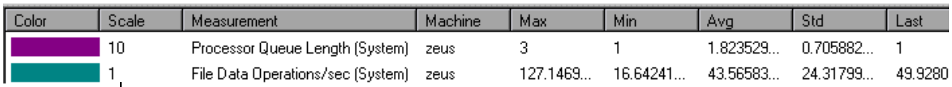
scale = 1



scale = 10

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

The legend below the graph indicates the scale factor.



Color	Scale	Measurement	Machine	Max	Min	Avg	Std	Last
🟪	10	Processor Queue Length (System)	zeus	3	1	1.823529...	0.705882...	1
🟦	1	File Data Operations/sec (System)	zeus	127.1469...	16.64241...	43.56583...	24.31799...	49.9280

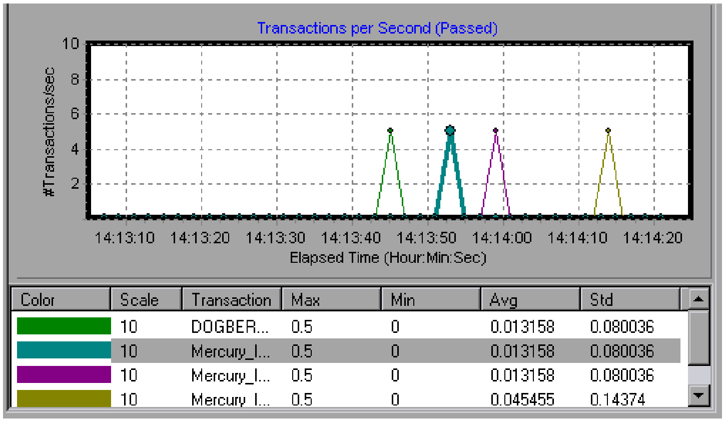
scale factor

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

Showing and Hiding Transactions

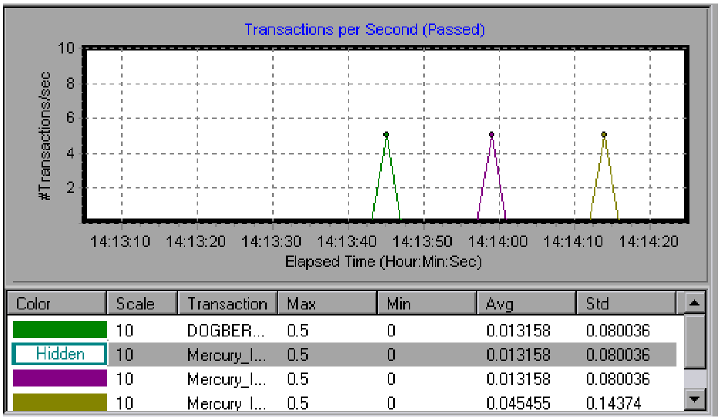
By default, the Transaction Monitor displays a line for each item in the transaction list. You can hide the line for any of the monitored transactions in order to focus on a specific measurement.

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



Color	Scale	Transaction	Max	Min	Avg	Std
🟩	10	DOGBER...	0.5	0	0.013158	0.080036
🟦	10	Mercury_I...	0.5	0	0.013158	0.080036
🟪	10	Mercury_I...	0.5	0	0.013158	0.080036
🟨	10	Mercury I...	0.5	0	0.045455	0.14374

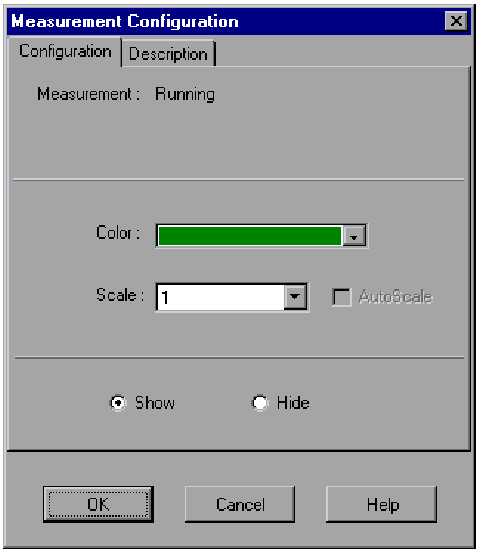
In this example, the second item in the legend is hidden.



Color	Scale	Transaction	Max	Min	Avg	Std
🟩	10	DOGBER...	0.5	0	0.013158	0.080036
Hidden	10	Mercury_I...	0.5	0	0.013158	0.080036
🟪	10	Mercury_I...	0.5	0	0.013158	0.080036
🟨	10	Mercury I...	0.5	0	0.045455	0.14374

To configure a measurement:

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



Measurement : Running

Color : 🟩

Scale : 1 ☐ AutoScale

● Show ○ Hide

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

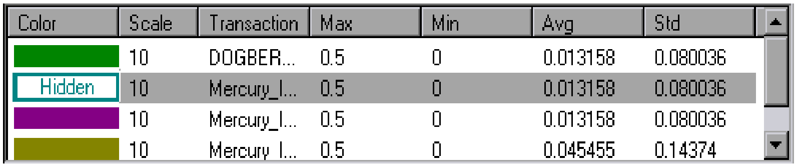
3 To change the scale, clear the **Autoscale** check box and select the desired ratio from the Scale list.

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

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

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

The specified changes are reflected in the graph and in the legend beneath the graph. The color is displayed in the first column of the legend. Hidden transactions are displayed as unfilled boxes. The scale is displayed in the legend's second column.



Color	Scale	Transaction	Max	Min	Avg	Std
🟩	10	DOGBER...	0.5	0	0.013158	0.080036
Hidden	10	Mercury_I...	0.5	0	0.013158	0.080036
🟪	10	Mercury_I...	0.5	0	0.013158	0.080036
🟫	10	Mercury I...	0.5	0	0.045455	0.14374

Exporting Online Monitor Graphs

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

To export online graphs to HTML:

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

Viewing Data Offline

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

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

14

Monitoring over a Firewall

To enable monitoring of your servers from outside the firewall, *Monitors over Firewall* is installed on designated machines inside the firewall. The installation sets up the Server Monitor mediator (referred to as the "mediator" in this chapter) as well as the Server Monitor configuration tool. You then configure the servers to monitor, and define the specific measurements that ProTune collects for each monitored server.

This chapter describes:

- Installing Monitors over Firewall
- Installing MI_Listener
- Preparing for Data Collection
- Configuring Server Monitor Properties
- Adding and Removing Measurements
- Configuring Measurement Frequency
- Configuring the Network Delay Monitor over a Firewall

About Monitoring over the Firewall

Once you set up your environment, as described in Chapter 12, "Working with Firewalls," and install the Monitoring over Firewall component, as described in "Installing Monitors over Firewall," on page 184, continue with the steps below:

1 **Prepare for data collection.**

Check that you can obtain information for the monitors configured inside the firewall. Refer to "Preparing for Data Collection," on page 189.

2 **Configure server monitor properties.**

Refer to "Configuring Server Monitor Properties," on page 190.

3 **Add and remove measurements.**

Add measurements to monitor for each server. If ProTune added default measurements, you can edit them as required. Refer to "Adding and Removing Measurements," on page 192.

4 **Configure measurement frequencies.**

Set a measurement schedule for each measurement to be reported. Refer to "Configuring Measurement Frequency," on page 193.

Installing Monitors over Firewall

Monitors over Firewall may have been installed during ProTune installation. To check whether it was installed, click **Start** > **Programs** > **ProTune** > **Advanced Settings.** If the **Monitor Configuration** option appears on the list of ProTune options, then *Monitors over Firewall* was already installed, and you can proceed to "Installing MI_Listener" on page 189.

If Monitors over Firewall was not yet installed, you need to install it using one of the following:

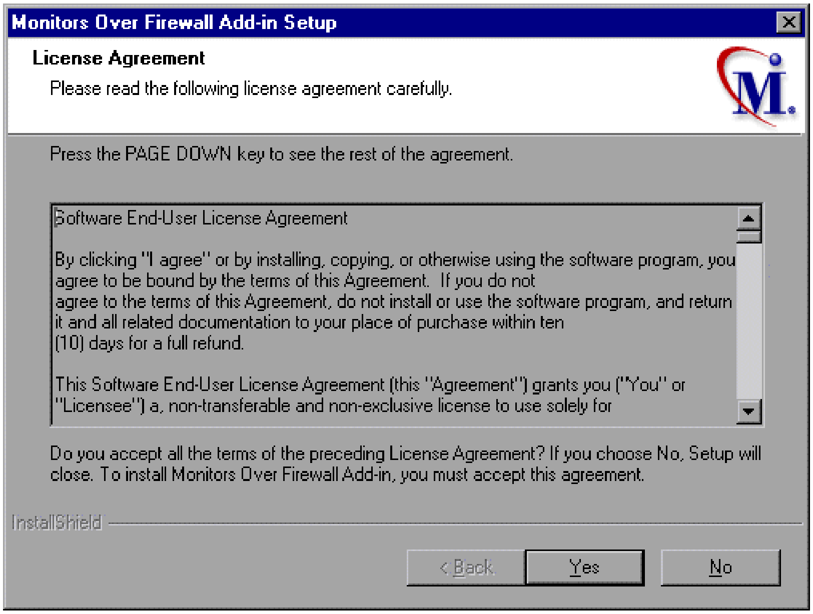
➤ Perform a custom installation of ProTune from the ProTune CD, choosing only the Monitors over Firewall option. For instructions on performing a custom installation of ProTune, refer to the *ProTune Installation Guide*.

➤ Obtain the *Monitors over Firewall* file from the Mercury Interactive Customer Support Web site (http://support.mercuryinteractive.com). *Monitors over Firewall* is a standalone downloadable installation. It comes as a self-extracting installer file.

To install Monitors over Firewall from the Mercury Interactive Customer Support Web site:

1 Copy the self-extracting installer file to the mediator machine.

2 Double-click the installer file to begin installation. The software license agreement appears. Read the agreement, and click **Yes** to accept it. If you click **No**, Setup closes.



Monitors Over Firewall Add-in Setup

License Agreement

Please read the following license agreement carefully.

Press the PAGE DOWN key to see the rest of the agreement.

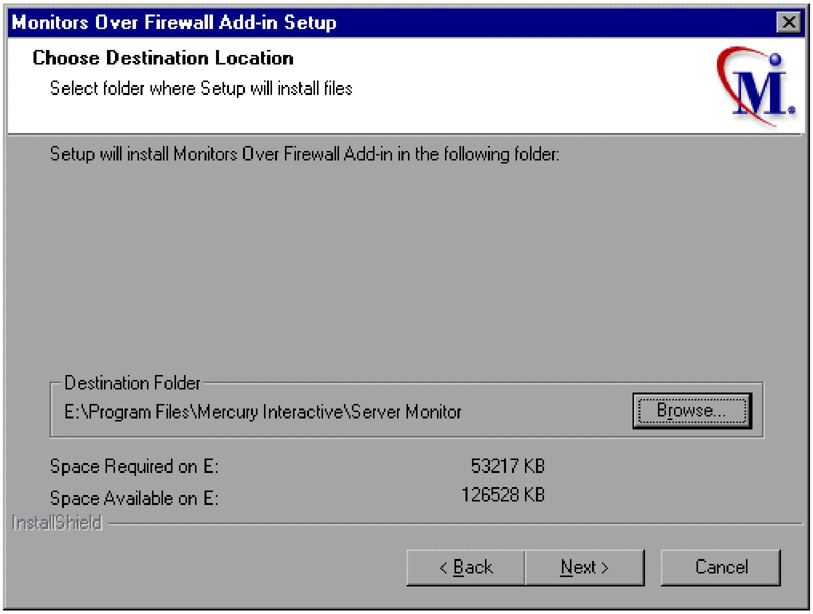
Software End-User License Agreement

By clicking "I agree" or by installing, copying, or otherwise using the software program, you agree to be bound by the terms of this Agreement. If you do not agree to the terms of this Agreement, do not install or use the software program, and return it and all related documentation to your place of purchase within ten (10) days for a full refund.

This Software End-User License Agreement (this "Agreement") grants you ("You" or "Licensee") a, non-transferable and non-exclusive license to use solely for

Do you accept all the terms of the preceding License Agreement? If you choose No, Setup will close. To install Monitors Over Firewall Add-in, you must accept this agreement.

3 In the Choose Destination Location screen, specify the folder in which to install the add-in. To select a different location, click **Browse**, choose a folder, and click **OK.**



Monitors Over Firewall Add-in Setup

Choose Destination Location

Select folder where Setup will install files

Setup will install Monitors Over Firewall Add-in in the following folder:

Destination Folder

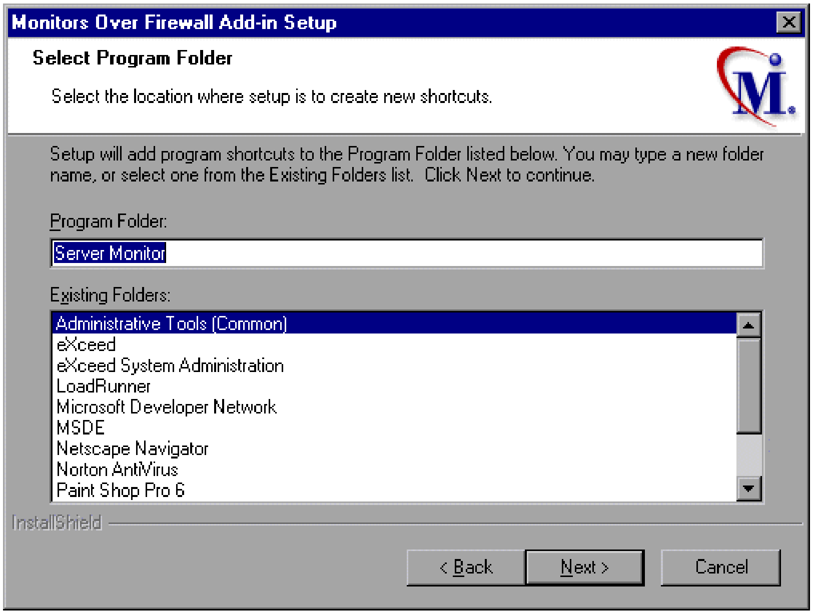
E:\Program Files\Mercury Interactive\Server Monitor

Space Required on E: 53217 KB

Space Available on E: 126528 KB

Click **Next**.

4 In the Select Program Folder screen, specify a program folder, or accept the default folder, *Server Monitor.*



Monitors Over Firewall Add-in Setup

Select Program Folder

Select the location where setup is to create new shortcuts.

Setup will add program shortcuts to the Program Folder listed below. You may type a new folder name, or select one from the Existing Folders list. Click Next to continue.

Program Folder:

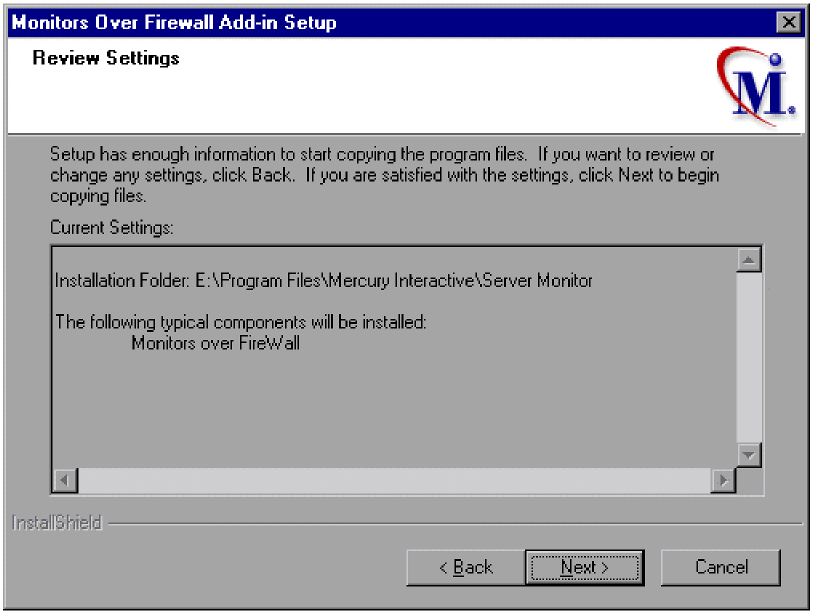
Server Monitor

Existing Folders:

Administrative Tools (Common)
eXceed
eXceed System Administration
LoadRunner
Microsoft Developer Network
MSDE
Netscape Navigator
Norton AntiVirus
Paint Shop Pro 6

Click **Next.**

5 In the Start Copying Files screen, review your settings. To make changes, click **Back**.



Setup has enough information to start copying the program files. If you want to review or change any settings, click Back. If you are satisfied with the settings, click Next to begin copying files.

Current Settings:

Installation Folder: E:\Program Files\Mercury Interactive\Server Monitor

The following typical components will be installed:
Monitors over FireWall

Click **Next**.

6 The installation process begins. To quit the installation, click **Cancel**.

7 Setup completes the installation process. The Setup Complete screen prompts you to restart your computer. You can delay restarting your computer until a later point, however, you must restart your computer before you use ProTune Server Monitors.

Click **Finish** to complete the setup process.

Installing MI_Listener

To enable monitoring over a firewall, you need to install MI Listener on one or more machines in the same LAN as the Console machine. Note that the Console installation automatically includes the MI Listener, so you can designate the Console as the MI Listener machine. For instructions, refer to the *ProTune Installation Guide*.

Preparing for Data Collection

In order to obtain information for the monitors configured inside the firewall, you must create a unique connection between the Console and the mediator machine, via the Mercury Interactive listener machine, MI Listener. You establish this connection by defining the mediator machine as a load generator.

To configure the Console for data collection:

1 You should already have configured the ProTune agent and the Console to operate over the firewall, as described in Chapter 12, “Working with Firewalls.”

2 Remember that in the Firewall tab of the Load Generator Information dialog box, you should enter the IP address of the MI Listener machine, and check **Enable Monitoring over Firewall**.

3 Connect to the load generator. Make sure that you obtain information for the monitors configured inside the firewall.

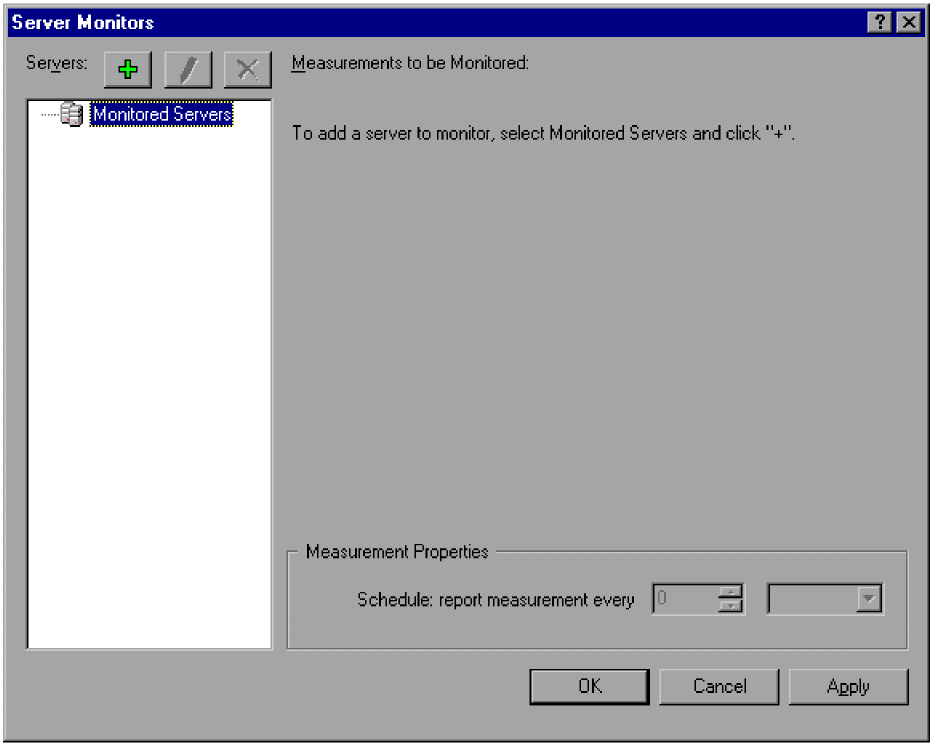
Configuring Server Monitor Properties

The next step is to add the server monitors. You configure server monitor properties (select the server whose resources you want to monitor, and the type of monitors to run), add the measurements to monitor for each server, and specify the frequency with which you want the monitored measurements to be reported.

To enable monitoring over the firewall, you need to configure server monitor properties.

To configure server monitor properties:

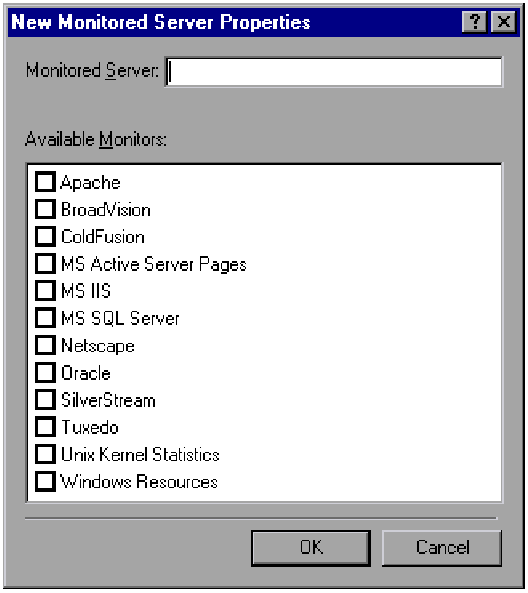
1 Select **Start** > **Programs** > **ProTune** > **Advanced Settings** > **Monitor Configuration**. For machines without the complete ProTune installation, select **Start** > **Programs** > **Server Monitor** > **Monitor Configuration.** The Monitor Configuration dialog box opens.



To add a server to monitor, select Monitored Servers and click "+".

Schedule: report measurement every 0

2 Click the **Add Server** button. The New Monitored Server Properties dialog box opens.



New Monitored Server Properties

Monitored Server:

Available Monitors:

- □ Apache
- □ BroadVision
- □ ColdFusion
- □ MS Active Server Pages
- □ MS IIS
- □ MS SQL Server
- □ Netscape
- □ Oracle
- □ SilverStream
- □ Tuxedo
- □ Unix Kernel Statistics
- □ Windows Resources

3 In the Monitored Server box, type the name or IP address of the server whose resources you want to monitor.

Note: To add several servers simultaneously, separate the server names or IP ranges with commas. For example: 255.255.255.0-255.255.255.5, server1, server2.

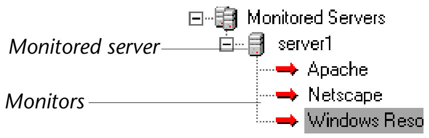
---

4 From the Available Monitors list, select the monitors appropriate for the server being monitored.

Note: Data can only be viewed for the monitors that are enabled with your ProTune license key. To preview your license key information, in the ProTune Console, select **Help** > **About ProTune**.

---

5 Click **OK** to close the New Monitored Server Properties dialog box to display the Monitored Servers list.



Monitored server

Monitors

Note that, for certain monitors, ProTune displays default measurements in the right pane. For details on selecting measurements, see “Adding and Removing Measurements” on page 192.

6 To add additional monitored servers to the list, repeat steps 1-5.

7 Click **Apply** to save your settings.

Adding and Removing Measurements

After you configure one or more server machines to monitor, you add measurements to monitor for each server. If ProTune added default measurements, you can edit them as required.

To add a measurement to monitor:

1 Select a server from the Monitored Servers list.

2 Click the **Add Measurement** button. Select the appropriate monitor. A dialog box opens, enabling you to choose measurements for the monitor you selected.

3 Select the measurements that you want to monitor, and click **OK**.

4 Click **Apply** to save your settings.

For information on configuring measurements for each server monitor, see the relevant chapter.

To remove a measurement from the measurements list:

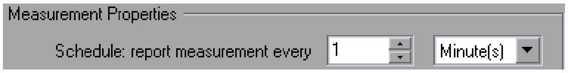
1 Select the measurement, and click the **Delete** button.

2 Click **Apply** to save your settings.

Configuring Measurement Frequency

Once you have configured monitor measurements, you configure measurement frequency.

In the Measurement Properties section, you set a measurement schedule for each measurement to be reported.



Measurement Properties

Schedule: report measurement every 1 Minute(s)

To set a measurement schedule for a measurement:

1 Select the configured server measurement you want to schedule.

2 Specify the frequency at which you want ProTune to report the measurement.

3 Click **Apply** to save your settings.

Configuring the Network Delay Monitor over a Firewall

To run the Network Delay Monitor when there are firewalls between the Console machine and the source machine, you must configure the Network Delay Monitor (see “Configuring the Network Monitor,” on page 248), and add the following to step 3 (on page 249):

In the **Monitor the network delay from machine** section, enter the server name or IP address of the source machine according to the following format: <*MI Listener machine*>:<*source machine local key*>.

where source machine local key is the unique key that you chose when configuring the ProTune Agent on the source machine.

For example: 12.12.12.3:vds

15

Run-Time and Transaction Monitoring

While running a session step, you can use ProTune's Run-Time and Transaction monitors to view graphs of run-time status and transaction performance.

This chapter describes:

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

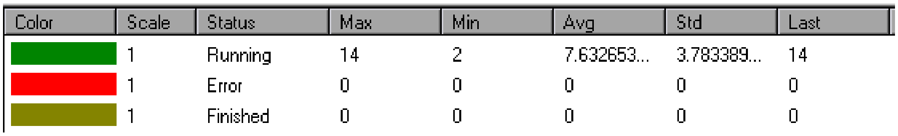
About Run-Time and Transaction Graphs

The *Run-Time* monitor provides information about the status of the Vusers participating in the session step, and the number and types of errors that the Vusers generate. In addition, the Run-Time monitor provides the User-Defined Data Points graph, which displays the real time values for user-defined points in a Vuser script.

The *Transaction* monitor displays the transaction rate and response time during session step execution. For more information about transactions, see "Adding Transactions to a Script" on page 200.

Run-Time Graphs

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



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

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

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

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

The **Vusers with Error Statistics** graph provides details about the number of Vusers that generate errors during session step execution. The errors are grouped by error source.

User-Defined Data Points Graph

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

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

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

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

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

Transaction Monitor Graphs

The *Transaction* monitor provides the following graphs:

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

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

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

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

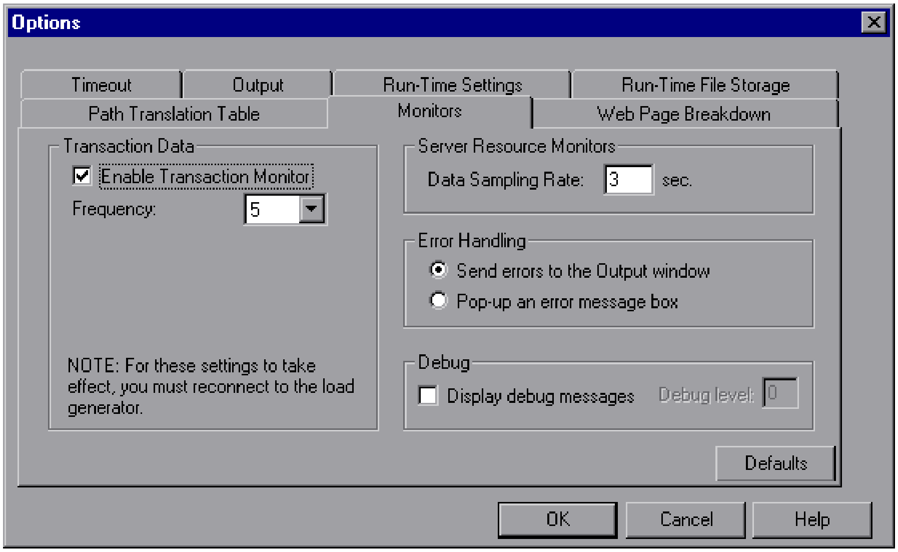
The **Total Transactions per Second (Passed)** graph shows the total number of completed, successful transactions per second (y-axis) as a function of the elapsed time in the session step (x-axis).

Enabling the Transaction Monitor

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

To enable the Transaction monitor:

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



Transaction Data

☑ Enable Transaction Monitor

Frequency: 5

NOTE: For these settings to take effect, you must reconnect to the load generator.

Server Resource Monitors

Data Sampling Rate: 3 sec.

Error Handling

● Send errors to the Output window

○ Pop-up an error message box

Debug

□ Display debug messages Debug level: 0

2 Enable transaction monitoring by selecting the **Enable Transaction Monitor** check box. To disable transaction monitoring, clear the **Enable Transaction Monitor** check box.

Adding Transactions to a Script

If there are no transactions defined in your Vuser script, no data will be displayed in the online graphs. To add transactions to an existing script, edit it using the appropriate tool. The following table shows the script generation tools for each script type:

Script type	Editing tool
GUI Windows	WinRunner
non-GUI Windows	VuGen (Vuser Generator)
SAP	QuickTest for SAP

To add a transaction to a script:

1. Click the **Design** tab to view the list of Vuser groups and scripts.
2. To edit a script for a Vuser group, select the group and click the **View Script** button to the right of the Session Groups window. The script generation tool opens.

   To edit a script for an individual Vuser, click **Vusers**. Right-click the Vuser whose script you want to edit, and select **View Script** to open the script generation tool.
3. Insert Start and End Transaction functions or markers throughout your script.

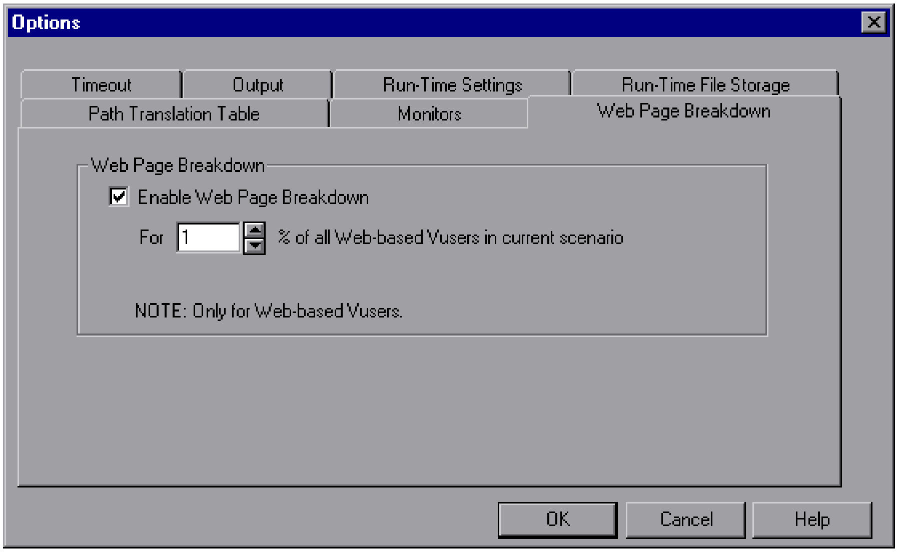
For more information, see the appropriate user's guide as described in the *Welcome* chapter.

Enabling Web Page Breakdown

In order for the Analysis to generate Web Page Breakdown graphs, which provide you with performance information for each transaction and sub-transaction defined in your script, you must enable the Web page breakdown feature in the Console before running your session step.

To enable Web page breakdown:

1 Choose **Tools** > **Options** and select the **Web Page Breakdown** tab.



Web Page Breakdown

☑ Enable Web Page Breakdown

For 1 % of all Web-based Vusers in current scenario

NOTE: Only for Web-based Vusers.

2 Select **Enable Web Page Breakdown**, and specify the percentage of Web Vusers for which you want Web page breakdown to be performed.

For more information about Web Page Breakdown graphs, see the *ProTune Analysis User's Guide*.

16

Web Resource Monitoring

You can obtain information about the performance of your Web server using ProTune's Web Resource monitor.

This chapter describes:

- Hits per Second Graph
- Throughput Graph
- HTTP Responses per Second Graph
- Pages Downloaded per Second Graph
- Retries per Second Graph

About Web Resource Monitoring

The Web Resource monitor enables you to analyze the throughput on the Web server, the number of hits per second that occurred during the session, the number of HTTP responses per second, the HTTP status codes (which indicate the status of HTTP requests, for example, "the request was successful," "the page was not found") returned from the Web server, the number of downloaded pages per second, and the number of server retries per second.

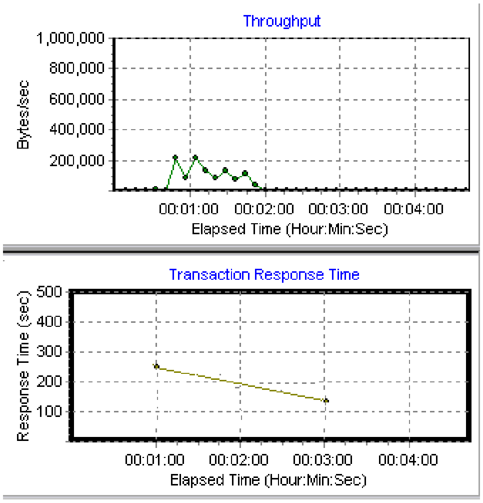
Hits per Second Graph

The **Hits Per Second** graph shows the number of hits (HTTP requests) to the Web server (y-axis) as a function of the elapsed time in the session (x-axis). This graph can display the whole session, or the last 60, 180, 600, or 3600 seconds. You can compare this graph to the Transaction Response Time graph to see how the number of hits affects transaction performance.

Throughput Graph

The **Throughput** graph shows the amount of throughput on the Web server (y-axis) during each second of the session run (x-axis). Throughput is measured in bytes and represents the amount of data that the Vusers received from the server at any given second. You can compare this graph to the Transaction Response Time graph to see how the throughput affects transaction performance.

In the following example, the Transaction Response time graph is compared with the Throughput graph. It is apparent from the graph that as the throughput decreases, the transaction response time also decreases. The peak throughput occurred at approximately 1 minute into the session. The highest response time also occurred at this time.



HTTP Responses per Second Graph

The **HTTP Responses per Second** graph shows the number of HTTP status codes—which indicate the status of HTTP requests, for example, “the request was successful,” “the page was not found”—(y-axis) returned from the Web server during each second of the session run (x-axis), grouped by status code. You can group the results shown in this graph by script (using the "Group By" function) to locate scripts which generated error codes.

The following table displays a list of HTTP status codes:

Code	Description
200	OK
201	Created
202	Accepted

Code	Description
203	Non-Authoritative Information
204	No Content
205	Reset Content
206	Partial Content
300	Multiple Choices
301	Moved Permanently
302	Found
303	See Other
304	Not Modified
305	Use Proxy
307	Temporary Redirect
400	Bad Request
401	Unauthorized
402	Payment Required
403	Forbidden
404	Not Found
405	Method Not Allowed
406	Not Acceptable
407	Proxy Authentication Required
408	Request Timeout
409	Conflict
410	Gone
411	Length Required
412	Precondition Failed

Code	Description
413	Request Entity Too Large
414	Request - URI Too Large
415	Unsupported Media Type
416	Requested range not satisfiable
417	Expectation Failed
500	Internal Server Error
501	Not Implemented
502	Bad Gateway
503	Service Unavailable
504	Gateway Timeout
505	HTTP Version not supported

For more information on the above status codes and their descriptions, see http://www.w3.org/Protocols/rfc2616/rfc2616-sec10.html#sec10.

Pages Downloaded per Second Graph

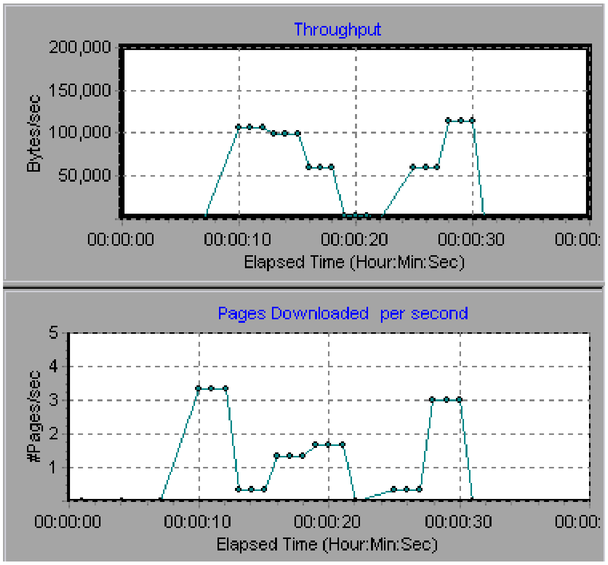
The **Pages Downloaded per Second** graph shows the number of Web pages (y-axis) downloaded from the server during each second of the session run (x-axis). This graph helps you evaluate the amount of load Vusers generate, in terms of the number of pages downloaded.

Note: In order to view the Pages Downloaded per Second graph, you must select **Pages per second (HMTL Mode only)** from the run-time settings Preferences tab before running your session.

Like throughput, downloaded pages per second is a representation of the amount of data that the Vusers received from the server at any given second.

- The Throughput graph takes into account each resource and its size (for example, the size of each *.gif* file, the size of each Web page).
- The Pages Downloaded per Second graph takes into account simply the number of pages.

In the following example, the Throughput graph is compared with the Pages Downloaded per Second graph. It is apparent from the graph that throughput is not proportional to the number of pages downloaded per second. For example, between 15 and 16 seconds into the session run, the throughput decreased while the number of pages downloaded per second increased.



Retries per Second Graph

The **Retries Per Second** graph shows the number of attempted Web server connections (y-axis) as a function of the elapsed time in the session (x-axis). A server connection is retried when the initial connection was unauthorized, when proxy authentication is required, when the initial connection was closed by the server, when the initial connection to the server could not be made, or when the server was initially unable to resolve the load generator's IP address.

17

System Resource Monitoring

You can monitor a machine's system resource usage during a session step run using ProTune's System Resource monitors.

This chapter describes:

- Configuring the Windows Resources Monitor
- Configuring the UNIX Resources Monitor
- Configuring an rstatd Daemon on UNIX
- Configuring the SNMP Resources Monitor
- Configuring the TUXEDO Monitor
- Configuring the Antara FlameThrower Monitor

About System Resource Monitoring

A primary factor in a transaction's response time is its system resource usage. Using the ProTune resource monitors, you can monitor the Windows, TUXEDO, UNIX, SNMP, or Antara FlameThrower resources on a machine during a session step run, and determine why a bottleneck occurred on a particular machine.

The Windows measurements correspond to the built-in counters available from the Windows Performance Monitor.

The UNIX measurements include those available by the *rstatd* daemon: average load, collision rate, context switch rate, CPU utilization, incoming packets error rate, incoming packets rate, interrupt rate, outgoing packets error rate, outgoing packets rate, page-in rate, page-out rate, paging rate, swap-in rate, swap-out rate, system mode CPU utilization, and user mode CPU utilization.

Note: You must configure an *rstatd* daemon on all UNIX machines being monitored. For information on how to configure an *rstatd* daemon, refer to the UNIX *man* pages, or see "Configuring an rstatd Daemon on UNIX," on page 221.

The TUXEDO monitor can monitor the server, load generator machine, workstation handler, and queue in a TUXEDO system. Note that in order to run the TUXEDO monitor, you must install the TUXEDO client libraries on the machine you want to monitor. For information on configuring the TUXEDO monitor, see "Configuring the TUXEDO Monitor," on page 225.

The SNMP monitor is available for monitoring machines using the Simple Network Management Protocol (SNMP). SNMP monitoring is platform independent.

The Antara FlameThrower monitor can measure the following performance counters: Layer, TCP, HTTP, SSL/HTTPS, Sticky SLB, FTP, SMPT, POP3, DNS, and Attacks.

The resource monitors are automatically enabled when you execute a session step. However, you must specify the machine you want to monitor and which resources to monitor for each machine. You can also add or remove machines and resources during the session step run.

Configuring the Windows Resources Monitor

Windows NT and Windows 2000 measurements correspond to the built-in counters available from the Windows Performance Monitor.

Note: To monitor a Windows NT or 2000 machine through a firewall, use TCP, port 139.

To configure the Windows Resources monitor:

1. Click the Windows Resources graph in the graph tree, and drag it into the right pane of the Run view.
2. Right-click the graph and choose **Add Measurement(s)**, or choose **Monitors** > **Add Online Measurement**.
3. In the Monitored Server Machines section of the Windows Resources dialog box, click **Add** to enter the server name or IP address of the machine you want to monitor. Select the platform on which the machine runs, and click **OK**.
4. In the Resource Measurements section of the Windows Resources dialog box, select the measurements that you want to monitor.

The following default measurements are available for Windows machines:

Object	Measurement	Description
System	**% Total Processor Time**	The average percentage of time that all the processors on the system are busy executing non-idle threads. On a multi-processor system, if all processors are always busy, this is 100%, if all processors are 50% busy this is 50% and if 1/4th of the processors are 100% busy this is 25%. It can be viewed as the fraction of the time spent doing useful work. Each processor is assigned an Idle thread in the Idle process which consumes those unproductive processor cycles not used by any other threads.
System	**File Data Operations/sec**	The rate at which the computer issues read and write operations to file system devices. This does not include File Control Operations.
Processor	**% Processor Time (Windows 2000)**	The percentage of time that the processor is executing a non-idle thread. This counter was designed as a primary indicator of processor activity. It is calculated by measuring the time that the processor spends executing the thread of the idle process in each sample interval, and subtracting that value from 100%. (Each processor has an idle thread which consumes cycles when no other threads are ready to run). It can be viewed as the percentage of the sample interval spent doing useful work. This counter displays the average percentage of busy time observed during the sample interval. It is calculated by monitoring the time the service was inactive, and then subtracting that value from 100%.

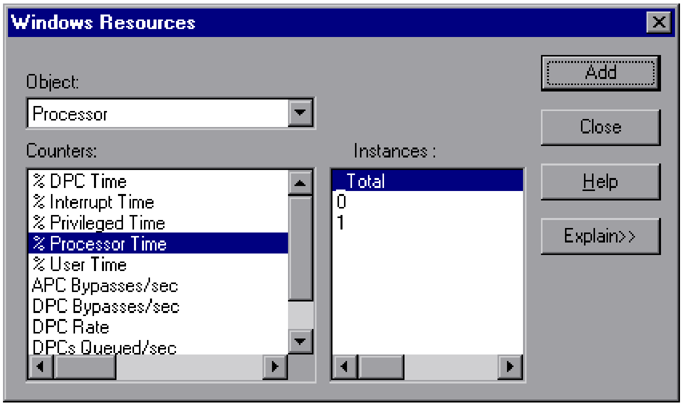
Object	Measurement	Description
System	**Processor Queue Length**	The instantaneous length of the processor queue in units of threads. This counter is always 0 unless you are also monitoring a thread counter. All processors use a single queue in which threads wait for processor cycles. This length does not include the threads that are currently executing. A sustained processor queue length greater than two generally indicates processor congestion. This is an instantaneous count, not an average over the time interval.
Memory	**Page Faults/sec**	This is a count of the page faults in the processor. A page fault occurs when a process refers to a virtual memory page that is not in its Working Set in the main memory. A page fault will not cause the page to be fetched from disk if that page is on the standby list (and hence already in main memory), or if it is in use by another process with which the page is shared.
PhysicalDisk	**% Disk Time**	The percentage of elapsed time that the selected disk drive is busy servicing read or write requests.
Memory	**Pool Nonpaged Bytes**	The number of bytes in the nonpaged pool, a system memory area where space is acquired by operating system components as they accomplish their appointed tasks. Nonpaged pool pages cannot be paged out to the paging file. They remain in main memory as long as they are allocated.

Object	Measurement	Description
Memory	**Pages/sec**	The number of pages read from the disk or written to the disk to resolve memory references to pages that were not in memory at the time of the reference. This is the sum of Pages Input/sec and Pages Output/sec. This counter includes paging traffic on behalf of the system cache to access file data for applications. This value also includes the pages to/from non-cached mapped memory files. This is the primary counter to observe if you are concerned about excessive memory pressure (that is, thrashing), and the excessive paging that may result.
System	**Total Interrupts/sec**	The rate at which the computer is receiving and servicing hardware interrupts. The devices that can generate interrupts are the system timer, the mouse, data communication lines, network interface cards, and other peripheral devices. This counter provides an indication of how busy these devices are on a computer-wide basis. See also Processor:Interrupts/sec.
Objects	**Threads**	The number of threads in the computer at the time of data collection. Notice that this is an instantaneous count, not an average over the time interval. A thread is the basic executable entity that can execute instructions in a processor.
Process	**Private Bytes**	The current number of bytes that the process has allocated that cannot be shared with other processes.

Note: To change the default counters for the Windows machine monitor, see “Changing a Monitor’s Default Counters,” on page 525.

If you are monitoring a Win2000 machine, some of the NT machine default counters may not be available (such as % Total CPU usage and Interrupts/sec). Proceed to step 5 in order to select counters appropriate for Win2000.

5 To select additional measurements, click **Add**. A dialog box displaying the available measurements and server properties opens.



Windows Resources

Object: Processor

Counters:
% DPC Time
% Interrupt Time
% Privileged Time
% Processor Time
% User Time
APC Bypasses/sec
DPC Bypasses/sec
DPC Rate
DPCs Queued/sec

Instances :
_Total
0
1

6 Select an object, a counter, and an instance. You can select multiple counters using the **Ctrl** key. The instance is relevant only if multiple instances of the highlighted counter are running. For a description of each counter, click **Explain>>** to expand the dialog box.

7 Click **Add** to place the selected counter on the resource list. Add all the desired resources to the list, and click **Close**.

8 Click **OK** in the Windows Resources dialog box to activate the monitor.

Note: If you want to monitor a remote Windows machine that does not use Windows domain security, you must authenticate the Console machine on the remote Windows machine. To authenticate the Console machine, create an account, or change the password of the account used to log on to the Console so that it matches the password and user name used to log on to the remote monitored Windows machine. When the remote Windows machine requests another machine's resources, it sends the logged-in user name and password of the machine requesting the resources.

Configuring the UNIX Resources Monitor

The UNIX kernel statistics measurements include those available by the *rstatd* daemon: average load, collision rate, context switch rate, CPU utilization, incoming packets error rate, incoming packets rate, interrupt rate, outgoing packets error rate, outgoing packets rate, page-in rate, page-out rate, paging rate, swap-in rate, swap-out rate, system mode CPU utilization, and user mode CPU utilization.

To configure the UNIX Resources monitor:

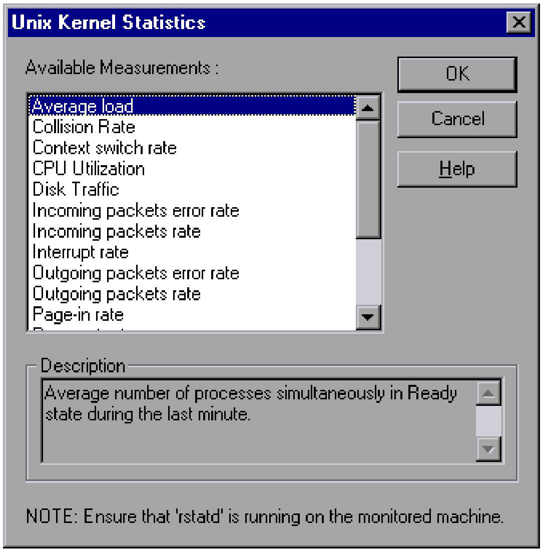
1. Click the UNIX Resources graph in the graph tree, and drag it into the right pane of the Run view.
2. Right-click the graph and choose **Add Measurement(s)**, or choose **Monitors** > **Add Online Measurement**.
3. In the Monitored Server Machines section of the UNIX Resources dialog box, click **Add** to enter the server name or IP address of the machine you want to monitor. Select **UNIX** from the list of platforms, and click **OK**.
4. In the Resource Measurements section of the UNIX Resources dialog box, select the default measurements you want to monitor.

The following default measurements are available for the UNIX machine:

Measurement	Description
Average load	Average number of processes simultaneously in READY state during the last minute
Collision rate	Collisions per second detected on the Ethernet
Context switches rate	Number of switches between processes or threads, per second
CPU utilization	Percent of time that the CPU is utilized
Disk rate	Rate of disk transfers
Incoming packets error rate	Errors per second while receiving Ethernet packets
Incoming packets rate	Incoming Ethernet packets per second
Interrupt rate	Number of device interrupts per second
Outgoing packets errors rate	Errors per second while sending Ethernet packets
Outgoing packets rate	Outgoing Ethernet packets per second
Page-in rate	Number of pages read to physical memory, per second
Page-out rate	Number of pages written to pagefile(s) and removed from physical memory, per second
Paging rate	Number of pages read to physical memory or written to pagefile(s), per second
Swap-in rate	Number of processes being swapped
Swap-out rate	Number of processes being swapped
System mode CPU utilization	Percent of time that the CPU is utilized in system mode
User mode CPU utilization	Percent of time that the CPU is utilized in user mode

Note: To change the default counters for the UNIX monitor, see "Changing a Monitor's Default Counters," on page 525.

5 To select additional measurements, click **Add**. The UNIX Kernel Statistics dialog box opens, displaying the available measurements and server properties.



Available Measurements :

- Average load
- Collision Rate
- Context switch rate
- CPU Utilization
- Disk Traffic
- Incoming packets error rate
- Incoming packets rate
- Interrupt rate
- Outgoing packets error rate
- Outgoing packets rate
- Page-in rate

Description

Average number of processes simultaneously in Ready state during the last minute.

NOTE: Ensure that 'rstatd' is running on the monitored machine.

6 To add UNIX measurements to the monitor list, select the desired measurements, and click **OK**.

7 Click **OK** in the UNIX Resources dialog box to activate the UNIX monitor.

Note: Ensure that the rstatd daemon is correctly configured and running on the monitored UNIX machine. For more information, see "Configuring an rstatd Daemon on UNIX," on page 221.

Configuring an rstatd Daemon on UNIX

To monitor UNIX resources, you must configure the rstatd daemon. Note that the rstatd daemon might already be configured, because when a machine receives an rstatd request, the inetd on that machine activates the rstatd automatically.

To verify whether the rstatd daemon is already configured:

The *rup* command reports various machine statistics, including rstatd configuration. Run the following command to view the machine statistics:

```
>rup host
```

You can also use **lr_host_monitor** and see if it returns any relevant statistics.

If the command returns meaningful statistics, the rstatd daemon is already configured and activated. If not, or if you receive an error message, the rstatd daemon is not configured.

To configure the rstatd daemon:

1. Run the command: *su root*
2. Go to */etc/inetd.conf* and look for the rstatd row (it begins with the word rstatd). If it is commented out (with a #), remove the comment directive, and save the file.
3. From the command line, run:

   ```
   kill -1 inet_pid
   ```

   where *inet_pid* is the pid of the inetd process. This instructs the inetd to rescan the */etc/inetd.conf* file and register all daemons which are uncommented, including the rstatd daemon.
4. Run *rup* again.

   If the command still does not indicate that the rstatd daemon is configured, contact your system administrator.

Note: To monitor a UNIX machine through a firewall, you must run a UNIX utility called rpcinfo and identify the rstatd's port number. By running rpcinfo -p <hostname>, you will receive a list of all RPC servers registered in the host's portmapper, along with the port number. This list will not change until rstatd is stopped and rerun.

Some firewalls allow you to open an RPC program number instead of a port. In such cases, open program 100001. If are prompted to include a version number, specify versions 3 and 4.

Configuring the SNMP Resources Monitor

The SNMP Resources monitor is available for monitoring any machine that runs an SNMP agent, using the Simple Network Management Protocol (SNMP).

Note: You can specify a port number in the *snmp.cfg* file. If you do not specify a port, ProTune connects to default SNMP port 161. You can also specify a machine name in the following format:
<server name>:*<port number>*

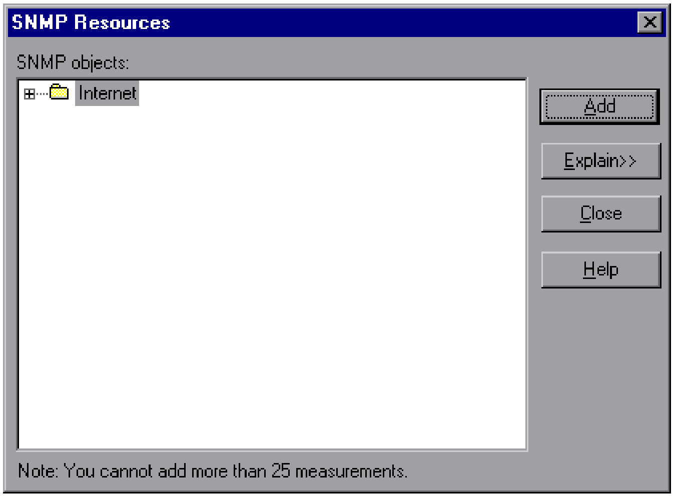
To monitor SNMP resources through a firewall, use ports 161 or 162.

To configure the SNMP Resources monitor:

1. Click the SNMP Resources graph in the graph tree, and drag it into the right pane of the Run view.
2. Right-click the graph and choose **Add Measurement(s)**, or choose **Monitors** > **Add Online Measurement**.
3. In the Monitored Server Machines section of the SNMP dialog box, click **Add** to enter the server name or IP address of the machine you want to monitor. Select the platform on which the machine runs, and click **OK**.

4 In the Resource Measurements section of the SNMP dialog box, click **Add** to select the measurements that you want to monitor.

The SNMP Resources dialog box opens.



Note: You cannot add more than 25 measurements.

5 Browse the SNMP Object tree.

6 To measure an object, select it, and click **Add**. For a description of each resource, click **Explain>>** to expand the dialog box. Add all the desired resources to the list, and click **Close**.

Note: The SNMP monitor can only monitor up to 25 measurements.

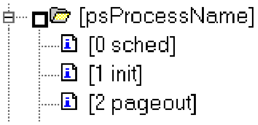
---

7 Click **OK** in the SNMP dialog box to activate the monitor.

You can modify the list of resources that you want to monitor at any point during the session step. Note that a session step does not have to be active in order for you to monitor the resources on a remote machine.

Note: You can improve the level of measurement information for the SNMP monitor by enabling measurements with string values to be listed in addition to measurements with numeric values, and by enabling the name modifier which displays the string value as an identifying part of the measurement name.

In the following example of a measurement using the name modifier, the string value of ProcessName (sched) is displayed in addition to its instance ID (0):



[psProcessName]
[0 sched]
[1 init]
[2 pageout]

To enable this feature, add the following line to the *<ProTune root folder>\dat\monitors\snmp.cfg* file:

```
SNMP_show_string_nodes=1
```

Usage Notes: You can select more than one name modifier, but the first in the hierarchy will be used. Each time the SNMP Add Measurements dialog box opens, the information is reread from the *snmp.cfg* file. You cannot add the same measurement twice (once with a name modifier and once without it). If you do so, an error message is issued.

Configuring the TUXEDO Monitor

The TUXEDO monitor allows you to measure and view your TUXEDO client's performance.

Note: If TUXEDO 7.1 or higher is installed on the Console machine, more than one TUXEDO application server can be monitored at a time. However, if TUXEDO 6.5 or below is installed on the Console machine, only one TUXEDO application server can be monitored at a time. Use a TUXEDO 6.x client if a TUXEDO 6.x server is used, and TUXEDO 7.1 or above if a TUXEDO 7.1 or above server is used.

Before you set up the monitor:

1 Ensure that a TUXEDO workstation client (not a native client) is installed on the Console machine.

Note: A TUXEDO workstation client communicates with the application server over the network, and is not required to run the TUXEDO application server on the same machine. A native client can only communicate with the TUXEDO application server if it is part of the relevant TUXEDO domain.

---

2 Define the TUXEDO environment variables on the Console machine—set the TUXDIR variable to the TUXEDO installation directory, and add the TUXEDO bin directory to the PATH variable.

3 Configure the TUXEDO application server so that the workstation listener (WSL) process is running. This enables the application server to accept requests from workstation clients. Note that the address and port number used to connect to the application server must match those dedicated to the WSL process.

To configure the TUXEDO monitor:

1 Click the TUXEDO graph in the graph tree, and drag it into the right pane of the Run view.

2 Right-click the graph and choose **Add Measurement(s)**, or choose **Monitors** > **Add Online Measurement**.

3 In the Monitored Server Machines section of the TUXEDO dialog box, click **Add** to enter the server name or IP address of the machine you want to monitor. Select the platform on which the machine runs, and click **OK**.

4 Click **Add** in the Resource Measurements section of the TUXEDO dialog box to select the measurements that you want to monitor. You will be prompted to enter information about the TUXEDO server: Login Name, Password, Server Name, Client Name.

Note: This information is located in the Logon section of the *tpinit.ini* file in the recorded script's directory. Click the **Browse** button and navigate to the *tpinit.ini* file of that ProTune script. You can also determine the client name from the **lrt_tpinitialize** statement in the recorded script.

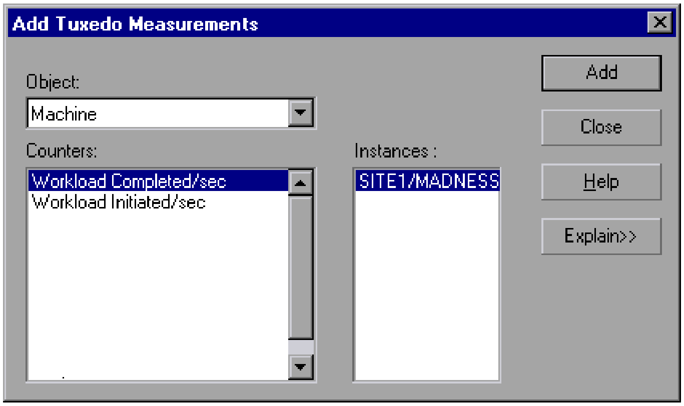
If you already know the required values, you can manually type them into the dialog box. The format of the server name is *//<machine name>:<port number>*. Alternatively, you can specify the IP address instead of the machine name. The hexadecimal format used by old versions of TUXEDO is also supported. Note that quotation marks should not be used.

In the following example of a *tpinit.ini* file, the TUXEDO monitor was configured for a server named URANUS using port 65535, and a client named bankapp. The logon user name was Smith and the password was mypasswd.

```
[Logon]
LogonServername=//URANUS:65535
LogonUsrName=Smith
LogonCltName=bankapp
LogonGrpName=
LogonPasswd=mypasswd
LogonData=
```

The TUXEDO monitor can only connect to one application server during a Console session. Once it connects to an application server, that server is the only one used by the monitor until the Console is closed. This applies even when all of the counters are deleted from the monitor.

5 Click **OK**. The Add TUXEDO Measurements dialog box opens.



Add Tuxedo Measurements

Object: Machine

Counters: Workload Completed/sec, Workload Initiated/sec

Instances : SITE1/MADNESS

6 Select a TUXEDO object from the Object list. Select the measurements and instances you want to monitor. The following table lists the available TUXEDO monitor measurements:

Monitor	Measurements
Server	**Requests per second** - How many server requests were handled per second
	Workload per second -The workload is a weighted measure of the server requests. Some requests could have a different weight than others. By default, the workload is always 50 times the number of requests.
Machine	**Workload completed per second** - The total workload on all the servers for the machine that was completed, per unit time
	Workload initiated per second - The total workload on all the servers for the machine that was initiated, per unit time
	Current Accessers - Number of clients and servers currently accessing the application either directly on this machine or through a workstation handler on this machine.
	Current Clients - Number of clients, both native and workstation, currently logged in to this machine.
	Current Transactions - Number of in use transaction table entries on this machine.
Queue	**Bytes on queue** - The total number of bytes for all the messages waiting in the queue
	Messages on queue - The total number of requests that are waiting on queue. By default this is 0.

Monitor	Measurements
Workstation Handler (WSH)	**Bytes received per second** - The total number of bytes received by the workstation handler, per unit time
	Bytes sent per second - The total number of bytes sent back to the clients by the workstation handler, per unit time
	Messages received per second - The number of messages received by the workstation handler, per unit time
	Messages sent per second - The number of messages sent back to the clients by the workstation handler, per unit time
	Number of queue blocks per second - The number of times the queue for the workstation handler blocked, per unit time. This gives an idea of how often the workstation handler was overloaded.

7 Click **Add** to place the selected object on the resource list. Add all the desired objects to the list, and click **Close**.

8 Click **OK** in the TUXEDO dialog box to activate the monitor.

Configuring the Antara FlameThrower Monitor

You select measurements to monitor the Antara FlameThrower server using the Antara FlameThrower Monitor Configuration dialog box.

To configure the Antara FlameThrower monitor:

1 Click the Antara FlameThrower graph in the graph tree, and drag it into the right pane of the Run view.

2 Right-click the graph and choose **Add Measurement(s)**, or choose **Monitors** > **Add Online Measurement**.

3 In the Monitored Server Machines section of the Antara FlameThrower dialog box, click **Add** to enter the server name or IP address of the machine you want to monitor. Enter the server name or IP address according to the following format: *<server name>:<port number>*.

For example: merc1:12135

Select the platform on which the machine runs, and click **OK**.

4 Click **Add** in the Resource Measurements section of the Antara FlameThrower dialog box to select the measurements that you want to monitor. The Antara FlameThrower Monitor Configuration dialog box opens.

5 Browse the Measured Components tree.

6 Check the required performance counters in the Antara FlameThrower Monitor Configuration window's right pane.

The following tables describe the counters that can be monitored:

Layer Performance Counters

Measurement	Description
TxBytes	The total number of Layer 2 data bytes transmitted.
TxByteRate(/sec)	The number of Layer 2 data bytes transmitted per second.
TxFrames	The total number of packets transmitted.
TxFrameRate(/sec)	The number of packets transmitted per second.
RxBytes	The total number of Layer 2 data bytes received.
RxByteRate(/sec)	The number of Layer 2 data bytes received per second.
RxFrames	The total number of packets received.
RxFrameRate(/sec)	The number of packets received per second.

TCP Performance Counters

Measurement	Description
ActiveTCPConns	Total number of currently active TCP connections.
SuccTCPConns	Total number of SYN ACK packets received.

Measurement	Description
SuccTCPConn Rate(/sec)	Number of SYN ACK packets received per second.
TCPConnLatency(milisec)	Interval between transmitting a SYN packet and receiving a SYN ACK reply packet in msec.
MinTCPConn Latency(milisec)	Minimum TCPConnectionLatency in msec.
MaxTCPConn Latency(milisec)	Maximum TCPConnectionLatency in msec.
TCPSndConnClose	Total number of FIN or FIN ACK packets transmitted (Client).
TCPRcvConnClose	Total number of FIN or FIN ACK packets received (Client).
TCPSndResets	Total number of RST packets transmitted.
TCPRcvResets	Total number of RST packets received.
SYNSent	Total number of SYN packets transmitted.
SYNSentRate(/sec)	Number of SYN packets transmitted per second.
SYNAckSent	Total number of SYN ACK packets transmitted.
SYNAckRate(/sec)	Number of SYN ACK packets transmitted per second.

HTTP Performance Counters

Measurement	Description
HTTPRequests	Total number of HTTP Request command packets transmitted.
HTTPRequestRate (/sec)	Number of HTTP Request packets transmitted per second.
AvgHTTPData Latency(milisecs)	The average HTTP Data Latency over the past second in msec.
HTTPData Latency(milisecs)	Interval between transmitting a Request packet and receiving a response in msec.

Measurement	Description
DataThroughput (bytes/sec)	The number of data bytes received from the HTTP server per second.
MinHTTPData Latency(milisecs)	Minimum HTTPDataLatency in msec.
MaxHTTPData Latency(milisecs)	Maximum HTTPDataLatency in msec.
MinData Throughput (bytes/sec)	Minimum HTTPDataThroughput in seconds.
MaxData Throughput (bytes/sec)	Maximum HTTPDataThroughput in seconds.
SuccHTTPRequests	Total number of successful HTTP Request Replies (200 OK) received.
SuccHTTPRequest Rate(/sec)	Number of successful HTTP Request Replies (200 OK) received per second.
UnSuccHTTP Requests	Number of unsuccessful HTTP Requests.

SSL/HTTPS Performance Counters

Measurement	Description
SSLConnections	Number of ClientHello messages sent by the Client.
SSLConnection Rate(/sec)	Number of ClientHello messages sent per second.
SuccSSL Connections	Number of successful SSL Connections. A successful connection is one in which the Client receives the Server's finished handshake message without any errors.
SuccSSLConnection Rate(/sec)	Number of successful SSL connections established per second.

Measurement	Description
SSLAlertErrors	Number of SSL alert messages received by the client (e.g. bad_record_mac, decryption_failed, handshake_failure, etc..)
SuccSSLResumed Sessions	Number of SSL Sessions that were successfully resumed.
FailedSSLResumed Sessions	Number of SSL Sessions that were unable to be resumed.

Sticky SLB Performance Counters

Measurement	Description
Cookie AuthenticationFail	The number of Cookie's that were not authenticated by the Server.
SuccCookie Authentication	The number of Cookie's authenticated by the server.
SSLClientHellos	The number of Client Hello packets sent to the server.
SSLServerHellos	The number of Server Hello packets sent to back to the client.
SSLSessionsFailed	The number of Session ID's that were not authenticated by the server.
SSLSessions Resumed	The number of Session ID's authenticated by the server.
succSSLClientHellos	The number of Client Hello replies received by the client or packets received by the server.
succSSLServerHellos	The number of Server Hello's received by the client.

FTP Performance Counters

Measurement	Description
TPUsers	Total number of Ftp User command packets transmitted.
FTPUserRate(/sec)	Number of Ftp User command packets transmitted per second.
FTPUserLatency (milisecs)	Interval between transmitting a Ftp User command packet and receiving a response in msec.
MinFTPUserLatency (milisecs)	Minimum FTPUsersLatency in msec.
MaxFTPUserLatency (milisecs)	Maximum FTPUsersLatency in msec.
SuccFTPUsers	Total number of successful Ftp User command replies received.
SuccFTPUserRate (/sec)	Number of successful Ftp User command replies received per second.
FTPPasses	Total number of FTP PASS packets transmitted.
FTPPassRate(/sec)	Number of FTP PASS packets transmitted per second.
FTPPassLatency (milisecs)	Interval between transmitting a Ftp PASS packet and receiving a response in msec.
MinFTPPassLatency (milisecs)	Minimum FTPPassLatency in msec.
MaxFTPPassLatency (milisecs)	Maximum FTPPassLatency in msec.
SuccFTPPasses	Total number of successful FTP PASS replies received.
SuccFTPPassRate (/sec)	Number of successful FTP PASS replies received per second.
FTPControl Connections	Total number of SYN packets transmitted by the FTP client.
FTPControl ConnectionRate (/sec)	Number of SYN packets transmitted by the FTP client per second.

Measurement	Description
SuccFTPControl Connections	Total number of SYN ACK packets received by the FTP client.
SuccFTPControl ConnectionRate (/sec)	Number of SYN ACK packets received by the FTP Client per second.
FTPData Connections	Number of SYN ACK packets received by the FTP client per second.
FTPDataConnection Rate(/sec)	Number of SYN ACK packets transmitted by the FTP Client or received by the FTP Server per second.
SuccFTPData Connections	Total number of SYN ACK packets transmitted by the FTP Client or received by the FTP Server.
SuccFTPData ConnectionRate (/sec)	Number of SYN ACK packets received by the FTP server per second.
FtpAuthFailed	Total number of error replies received by the FTP client.
FTPGets	Total number of client Get requests.
FTPPuts	Total number of client Put requests.
SuccFTPGets	Total number of successful Get requests (data has been successfully transferred from server to client).
SuccFTPPuts	Total number of successful Put requests (data has been successfully transferred from client to server) .

SMTP Performance Counters

Measurement	Description
SMTPHelos	Total number of HELO packets transmitted.
SMTPHeloRate(/sec)	Number of HELO packets transmitted per second.
SMTPHeloLatency (milisecs)	Interval between transmitting a HELO packet and receiving a response in msec.
MinSMTPHelo Latency(milisecs)	Minimum SMTPHeloLatency in msec.

Measurement	Description
MaxSMTPHelo Latency(milisecs)	Maximum SMTPHeloLatency in msec.
SuccSMTPHelos	Total number of successful HELO replies received.
SuccSMTPHelo Rate(/sec)	Number of successful HELO replies received per second.
SMTPMailFroms	Total number of Mail From packets transmitted.
SMTPMailFromRate (/sec)	Number of Mail From packets transmitted per second.
SMTPMailFrom Latency(milisecs)	Interval between transmitting a Mail From packet and receiving a response in msec.
MinSMTPMailFrom Latency(milisecs)	Minimum SMTPMailFromLatency in msec.
MaxSMTPMailFrom Latency(milisecs)	Maximum SMTPMailFromLatency in msec.
SuccSMTPMail Froms	Total number of successful Mail From replies received.
SuccSMTPMailFrom Rate(/sec)	Number of successful Mail From replies received per second.
SMTPRcptTos	Total number of RcptTo packets transmitted.
SMTPRcptToRate (/sec)	Number of RcptTo packets transmitted per second.
SMTPRcptTo Latency(milisecs)	Interval between transmitting a RcptTo packet and receiving a response in msec.
MinSMTPRcptTo Latency(milisecs)	Minimum SMTPRcptToLatency in msec.
MaxSMTPRcptTo Latency(milisecs)	Maximum SMTPRcptToLatency in msec.
SuccSMTPRcptTos	Total number of successful RcptTo replies received.
SuccSMTPRcptTo Rate(/sec)	Number of successful RcptTo replies received per second.

Measurement	Description
SMTPDatas	Total number of Data packets transmitted.
SMTPDataRate(/sec)	Number of Data packets transmitted per second.
SMTPDataLatency (milisecs)	Interval between transmitting a Data packet and receiving a response in msec.
MinSMTPData Latency(milisecs)	Minimum SMTPDataLatency in msec.
MaxSMTPData Latency(milisecs)	Maximum SMTPDataLatency in msec.
SuccSMTPDatas	Total number of successful Data replies received.
SuccSMTPDataRate (/sec)	Number of successful Data replies received per second.

POP3 Performance Counters

Measurement	Description
POP3Users	Total number of Pop3 User command packets transmitted.
POP3UserRate(/sec)	Number of Pop3 User command packets transmitted per second.
POP3UserLatency (milisecs)	Interval between transmitting a Pop3 User command packet and receiving a response in msec.
MinPOP3User Latency(milisecs)	Minimum POP3UserLatency in msec.
MaxPOP3User Latency(milisecs)	Maximum POP3UserLatency in msec.
SuccPOP3Users	Total number of successful Pop3 User replies received.
SuccPOP3UserRate (/sec)	Number of successful Pop3 User replies received per second.
POP3Passes	Total number of Pop3 Pass command packets transmitted.

Measurement	Description
POP3PassRate(/sec)	Number of Pop3 Pass command packets transmitted per second.
POP3PassLatency (milisecs)	Interval between transmitting a Pop3 Pass packet and receiving a response in msec.
MinPOP3Pass Latency(milisecs)	Minimum POP3PassLatency in msec.
MaxPOP3Pass Latency(milisecs)	Maximum POP3PassLatency in msec.
SuccPOP3Passes	Total number of successful Pop3 Pass replies received.
SuccPOP3PassRate (/sec)	Number of successful Pop3 Pass replies received per second.
POP3Stats	Total number of Pop3 Stat command packets sent.
POP3StatRate(/sec)	Number of Pop3 Stat command packets transmitted per second.
POP3StatLatency (milisecs)	Interval between transmitting a Pop3 Stat packet and receiving a response in msec.
MinPOP3Stat Latency(milisecs)	Minimum POP3StartLatency in msec.
MaxPOP3Stat Latency(milisecs)	Maximum POP3StartLatency in msec.
SuccPOP3Stats	Total number of successful Pop3 Stat replies received.
SuccPOP3StatRate (/sec)	Number of successful Pop3 Stat replies received per second.
POP3Lists	Total number of Pop3 List command packets transmitted.
POP3ListRate(/sec)	Number of Pop3 List command packets transmitted per second.
POP3ListLatency (milisecs)	Interval between transmitting a Pop3 List packet and receiving a response in msec.

Measurement	Description
MinPOP3List Latency(milisecs)	Minimum POP3ListLatency in msec.
MaxPOP3List Latency(milisecs)	Maximum POP3ListLatency in msec.
SuccPOP3Lists	Total number of successful Pop3Lists received.
SuccPOP3ListRate (/sec)	Number of successful Pop3Lists received per second.
POP3Retrs	Total number of Pop3 Retr packets transmitted.
POP3RetrRate(/sec)	Number of Pop3 Retr packets transmitted per second.
POP3RetrLatency (milisecs)	Interval between transmitting a Pop3 Retr packet and receiving a response in msec.
MinPOP3Retr Latency(milisecs)	Minimum POP3RetrLatency in msec.
MaxPOP3Retr Latency(milisecs)	Maximum POP3RetrLatency in msec.
SuccPOP3Retrs	Total number of successful Pop3Retrs received.
SuccPOP3RetrRate (/sec)	Number of successful Pop3Retrs received per second.

DNS Performance Counters

Measurement	Description
SuccPrimaryDNS Request	Total number of Successful DNS requests made to the Primary DNS server.
SuccSecondaryDNS Request	Total number of Successful DNS requests made to the Secondary DNS server.
SuccDNSData RequestRate(/sec)	Number of Successful DNS Request packets transmitted per second.
PrimaryDNSFailure	Total number of DNS requests failures received from the Primary DNS server.

Measurement	Description
PrimaryDNSRequest	Total number of DNS requests made to the Primary DNS server.
SecondaryDNS Failure	Total number of DNS requests failures received from the Secondary DNS server.
SecondaryDNS Request	Total number of DNS requests made to the Secondary DNS server.
MinDNSData Latency	Minimum DNS Data Latency in msec.
MaxDNSData Latency	Maximum DNS Data Latency in msec.
CurDNSData Latency	Interval between sending a DNS request packet and receiving a response in msec.
DNSDataRequest Rate(/sec)	Number of DNS Request packets transmitted per second.
NoOf ReTransmission	Total number of DNS Request packets re
NoOfAnswers	Total number of Answers to the DNS Request packets.

Attacks Performance Counters

Measurement	Description
Attacks	Total number of attack packets transmitted (All Attacks)
AttackRate(/sec)	Number of attack packets transmitted per second (ARP, Land, Ping, SYN, and Smurf)
Havoc Flood	Number of Havoc packets generated (Stacheldraht only)
Icmp Flood	Number of ICMP attack packets generated (TFN, TFN2K, & Stacheldraht)
Mix Flood	Number of Mix packets generated (TFN2K only)
Mstream Flood	Number of Mstream packets generated (Stacheldraht only)

Measurement	Description
Null Flood	Number of Null packets generated (Stacheldraht only)
Smurf Flood	Number of Smurf packets generated (TFN, TFN2K, & Stacheldraht)
Syn Flood	Number of SYN packets generated (TFN, TFN2K, & Stacheldraht)
Targa Flood	Number of Targa packets generated (TFN2K only)
Udp Flood	Number of UDP packets generated (All DDoS Attacks only)

7 Click **OK** in the Antara FlameThrower Monitor Configuration dialog box, and in the Antara FlameThrower dialog box, to activate the Antara FlameThrower monitor.

18

Network Monitoring

You can use Network monitoring to determine whether your network is causing a delay in the session step. You can also determine the problematic network segment.

Note: You must have administrator privileges on the Windows source machine in order to run the Network monitor (unless you are using the ICMP protocol).

This chapter describes:

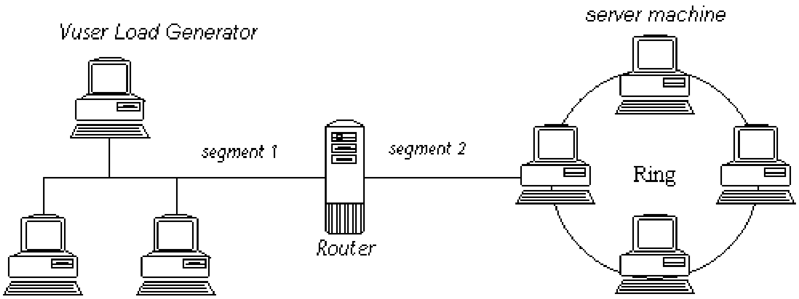
- Network Monitoring from a UNIX Source Machine
- Configuring the Network Monitor
- Viewing the Network Delay Time Graph

About Network Monitoring

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

In a true Web or client/server system, there are many network segments. A single network segment with poor performance can affect the entire system.

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



To measure network performance, the Network monitor sends packets of data across the network. When a packet returns, the monitor calculates the time it takes for the packet to go to the requested node and return. This time is the delay which appears in the Network Delay Time graph.

Using the online Network Delay Time graph, you can locate the network-related problem so that it can be fixed.

Note: The delays from the source machine to each of the nodes are measured concurrently, yet independently. It is therefore possible that the delay from the source machine to one of the nodes could be greater than the delay for the complete path between the source and destination machines.

Network Monitoring from a UNIX Source Machine

You can run the Network monitor on UNIX machines, using UDP or ICMP. Before running the Network monitor from a UNIX source machine:

- ➤ configure the source machine by assigning root permissions to the *merc_webtrace* process.
- ➤ make the necessary adjustments to either connect to the source machine through rsh, or through the agent.

Configuring the Source Machine

To configure the source machine, where ProTune is installed locally:

To assign root permissions to the *merc_webtrace* process, add an s-bit to *merc_webtrace*'s permissions, as follows:

1. Log in to the source machine as root.
2. Type: cd <ProTune>_installation/bin to change to the *bin* directory.
3. Type: chown root merc_webtrace to make the root user the owner of the *merc_webtrace* file.
4. Type: chmod +s merc_webtrace to add the s-bit to the file permissions.
5. To verify, type ls -l merc_webtrace. The permissions should look like: -rwsrwsr-x.

To configure the source machine, where ProTune is installed on the network:

In a ProTune network installation, the *merc_webtrace* process is on the network, not on the source machine disk. The following procedure copies the *merc_webtrace* file to the local disk, configures *mdrv.dat* to recognize the process, and assigns root permissions to *merc_webtrace*:

1. Copy *merc_webtrace* from *<ProTune>_installation/bin* to anywhere on the local disk of the source machine. For example, to copy the file to the */local/ProTune* directory, type: cp /net/tools/ProTune_installation/bin/merc_webtrace /local/ProTune

---

Note: All of the source machines that use the same network installation must copy *merc_webtrace* to the identical directory path on their local disk (for example, */local/ProTune*), since all of them use the same *mdrv.dat*.

---

2 Add the following line to the *ProTune_installation/dat/mdrv.dat* file, in the [monitors_server] section:

```
ExtCmdLine=-merc_webtrace_path /local/xxx
```

3 Log in to the source machine as root.

4 Type: cd ProTune_installation/bin to change to the *bin* directory.

5 Type: chown root merc_webtrace to make the root user the owner of the *merc_webtrace* file.

6 Type: chmod +s merc_webtrace to add the s-bit to the file permissions.

7 To verify, type ls -l merc_webtrace. The permissions should look like: -rwsrwsr-x.

Connecting to the Source Machine Through RSH

If the Console is connected to the source machine through rsh (default connection mode), then you don't need to activate the agent daemon. Before running the Network monitor the first time, you enter an encrypted user name and password in the Network monitor configuration file.

To create an encrypted user name and password:

1 On the Console machine, type: cd <ProTune>_installation/bin to change to the *bin* directory.

2 Run *CryptonApp.exe*.

3 Type your RSH user name and password, separated by a vertical bar symbol. For example, myname|mypw.

4 Copy the encrypted string to the clipboard (highlight the string and click **ctrl+c**).

5 Add the following line to the *<ProTune>_installation/dat/monitors/ndm.cfg* file, in the [hosts] section:

```
Host = <encrypted string copied from clipboard>
```

6 Close and open the current session step. ProTune will read the updated configuration file and recognize the source machine for monitoring.

Connecting to the Source Machine Through the Agent

If the Console is not connected to the source machine through RSH, then make sure that the agent daemon is active on the source machine before running the Network monitor. For more information about working without RSH, refer to the section titled “UNIX Shell” in Appendix A, “Troubleshooting the Console.”

To activate the agent daemon:

If you are not working in RSH, invoke the agent daemon on the source machine.

1 Type `m_daemon_setup -install` from the *<ProTune>_installation/bin* directory.

2 Make sure that the agent daemon is running whenever you activate the Network monitor.

Configuring the Network Monitor

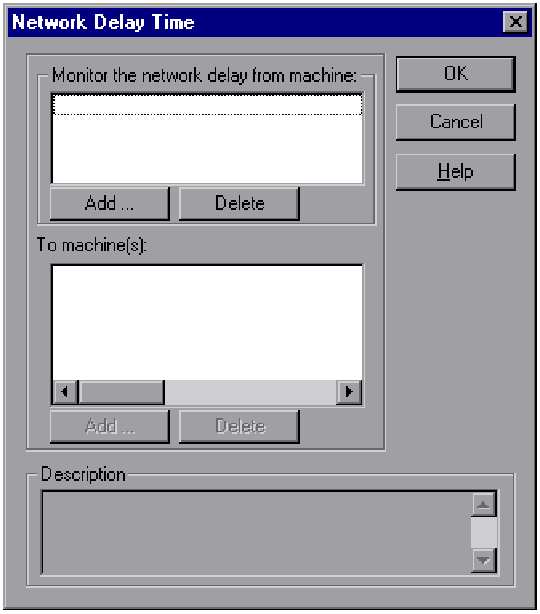
You configure the Network monitor from the Run view of the Console before you begin running a session step. Using the Network Delay Time and Add Destination Machines for Network Delay Monitoring dialog boxes, you select the network path you want to monitor.

Note: To enable network monitoring, you must install the ProTune agent on the source machine. You do not have to install the ProTune agent on the destination machine.

To configure the Network monitor:

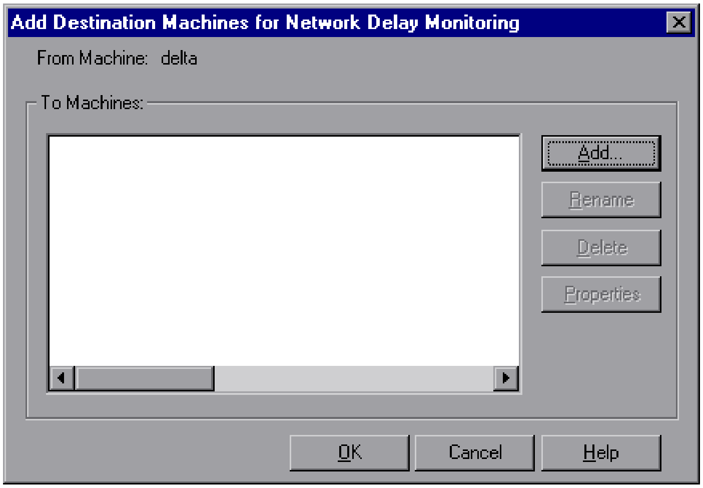
1 In the graph tree view, select the **Network Delay Time** graph and drag it into the right pane.

2 Right-click the graph and choose **Add Measurement(s)**, or choose **Monitors** > **Add Online Measurement**. The Network Delay Time dialog box opens.



3 In the **Monitor the network delay from machine** section, click **Add** to enter the server name or IP address of the source machine, from which you want the network path monitoring to begin. Select the platform on which the machine runs, and click **OK**.

4 In the **To machine(s)** section of the Network Delay Time dialog box, click **Add** to enter the name of the machine at the final destination of the path you want to monitor. The Add Destination Machines for Network Delay Monitoring dialog box opens.

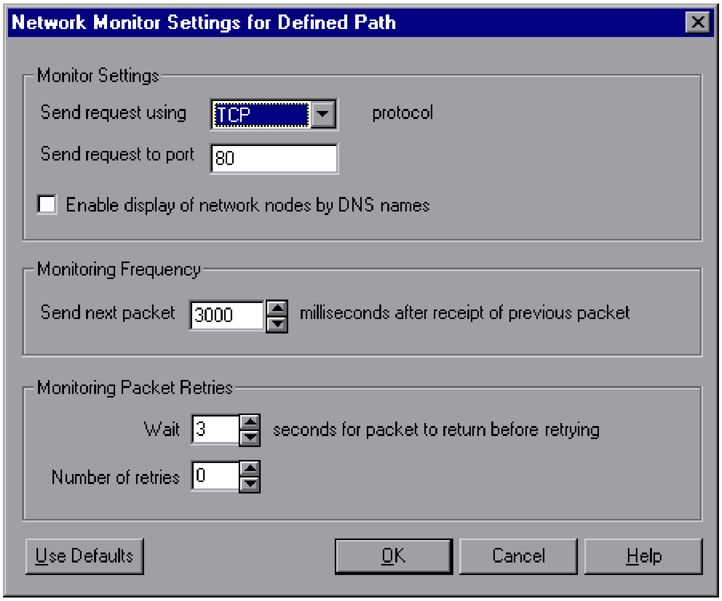


5 Click **Add**, enter the name of the destination machine, and click **OK**. The name of the machine appears in the Add Destination Machines for Network Delay Monitoring dialog box. Repeat this procedure for each path you want to monitor.

To rename a machine, click **Rename**, and enter a new name for the machine.

To delete a machine, select it and click **Delete**.

6 Click **Properties** to configure additional network monitor settings. The Network Monitor Settings for Defined Path dialog box opens.



Network Monitor Settings for Defined Path

Monitor Settings

Send request using TCP protocol

Send request to port 80

□ Enable display of network nodes by DNS names

Monitoring Frequency

Send next packet 3000 milliseconds after receipt of previous packet

Monitoring Packet Retries

Wait 3 seconds for packet to return before retrying

Number of retries 0

7 In the Monitor Settings box, select the protocol and enter the port number being used by the network path. The Network monitor supports three protocols: TCP, UDP, and ICMP. It is recommended that you use the default protocol. In Windows, the default is TCP, and in UNIX, the default is UDP.

8 Select **Enable display of network nodes by DNS names** if you want to view the DNS name of each node along the network path, in addition to its IP address. Note that selecting this option will decrease the speed of the Network monitor.

9 In the Monitoring Frequency box, select the number of milliseconds the monitor should wait between receiving a packet and sending out the next packet. The default value is 3000 milliseconds. If you have a long, steady session step, you can increase the interval by several seconds.

10 In the Monitoring Packet Retries box, select the maximum number of seconds that the monitor should wait for a packet to return before it retries to send the packet. The default value is 3 seconds. If your network is very large and loaded (an internet connection with a low capacity), you should increase the value by several seconds. If you have a small network (such as a LAN), you can decrease the value.

In addition, select the number of times the Network monitor should try resending a packet to a node if the packet is not initially returned. The default value is 0.

Network Monitoring over a Firewall

If you are monitoring a network in which there are firewalls between the source and the destination machines, you must configure the firewalls to allow the network data packets to reach their destinations.

➤ If you are using the TCP protocol, the firewall that protects the destination machine should not block outgoing ICMP_TIMEEXCEEDED packets (packets that are sent outside the firewall from the machine). In addition, the firewall protecting the source machine should allow ICMP_TIMEEXCEEDED packets to enter, as well as TCP packets to exit.

➤ If you are using the ICMP protocol, the destination machine's firewall should not block incoming ICMP_ECHO_REQUEST packets, or outgoing ICMP_ECHO_REPLY and ICMP_ECHO_TIMEEXCEEDED packets. In addition, the firewall protecting the source machine should allow ICMP_ECHO_REPLY and ICMP_ECHO_TIMEEXCEEDED packets to enter, and ICMP_ECHO_REQUEST packets to exit.

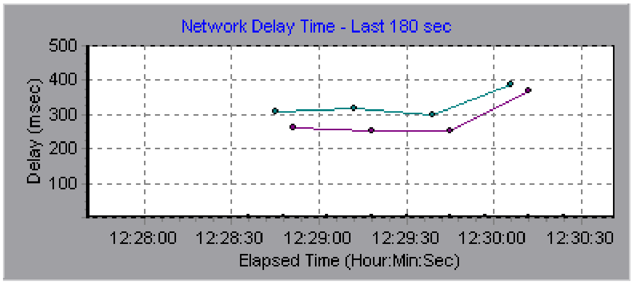
➤ If you are using the UDP protocol, ensure that the UDP protocol can access the destination machine from the source machine. The destination machine's firewall should not block outgoing ICMP_DEST_UNREACHABLE and ICMP_ECHO_TIMEEXCEEDED packets. In addition, the firewall protecting the source machine should allow ICMP_DEST_UNREACHABLE and ICMP_ECHO_TIMEEXCEEDED packets to enter.

Note: To run the Network Delay Monitor when there are firewalls between the Console machine and the source machine, you must configure the ProTune agent, MI Listener, and Network monitor for monitoring over a firewall. For more information see "Configuring the ProTune Agents in LAN1," on page 146, "Installing and Configuring the MI_Listener in LAN2," on page 154, and "Configuring the Network Delay Monitor over a Firewall," on page 193.

Viewing the Network Delay Time Graph

The **Network Delay Time** graph shows the delay for the complete path between the source and destination machines (y-axis) as a function of the elapsed session step time (x-axis).

Each path defined in the Add Destination Machines for Network Delay Monitoring dialog box is represented by a separate line with a different color in the graph.

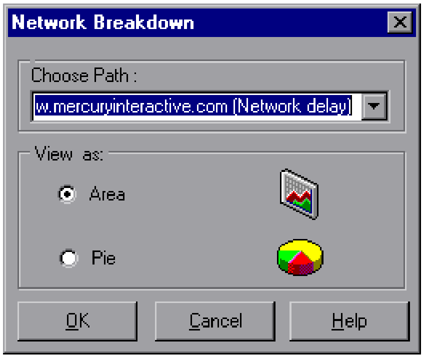


To view the DNS names of the measurements displayed in the legend, right-click the graph and select **View as DNS Name**.

In addition, you can view the delay from the source machine to each of the nodes along the network path.

To view the delay time for the network segments:

1 Right-click the Network Delay Time graph, and select **View Segments**. The Network Breakdown dialog box opens.



Choose Path : w.mercuryinteractive.com (Network delay)

View as: ● Area ○ Pie

2 Select the path that you want to break down.

3 Choose whether you want to view the network segments of the graph of the graph you chose as an area graph or a pie graph.

4 Click **OK** to close the Network Breakdown dialog box. The delay time for the network segments of the path you chose is displayed in the graph view area.

Note: The segment delays are measured approximately, and do not add up to the network path delay which is measured exactly. The delay for each segment of the path is estimated by calculating the delay from the source machine to one node and subtracting the delay from the source machine to another node. For example, the delay for segment B to C is calculated by measuring the delay from the source machine to point C, and subtracting the delay from the source machine to point B.

To return to the complete path delay time view, select **Hide Segments** from the right-click menu.

19

Firewall Server Performance Monitoring

During a session step run, you can monitor the firewall server in order to isolate server performance bottlenecks.

This chapter describes:

➤ Configuring the Check Point FireWall-1 Server Monitor

About the Firewall Server Monitor

The Firewall server online monitor measures the performance of a Firewall server during session step execution. In order to obtain performance data, you need to activate the Firewall server monitor before executing the session step, and indicate which statistics and measurements you want to monitor.

Configuring the Check Point FireWall-1 Server Monitor

To monitor the Check Point FireWall-1 server, you must select the counters you want the Check Point FireWall-1 server monitor to measure. You select these counters using the Check Point FireWall-1 SNMP Resources dialog box.

To configure the Check Point FireWall-1 server monitor:

1 Click the Check Point FireWall-1 graph in the graph tree, and drag it into the right pane of the Run view.

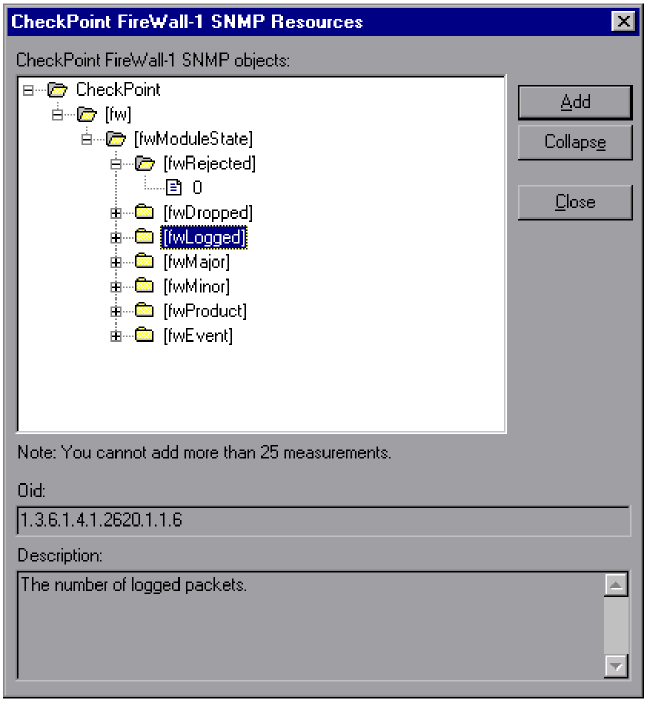
2 Right-click the graph and choose **Add Measurement(s)**, or choose **Monitors** > **Add Online Measurement**.

3 In the Monitored Server Machines section of the Check Point FireWall-1 dialog box, click **Add** to enter the server name or IP address of the machine you want to monitor. Select the platform on which the machine runs, and click **OK**.

Note: You can specify a port number in the *snmp.cfg* file. If you do not specify a port number, ProTune connects to port 260, the default port for the Check Point FireWall-1 SNMP agent. You can also specify a machine name and port number in the Add Machine dialog box using the following format:
<machine name>:*<port number>*

---

4 Click **Add** in the Resource Measurements section of the Check Point FireWall-1 dialog box. The Check Point FireWall-1 SNMP Resources dialog box opens.



Note: You cannot add more than 25 measurements.

Oid: 1.3.6.1.4.1.2620.1.1.6

Description: The number of logged packets.

5 Select the measurements you want to monitor. The following default counters can be monitored:

Measurement	**Description**
fwRejected	The number of rejected packets.
fwDropped	The number of dropped packets.
fwLogged	The number of logged packets.

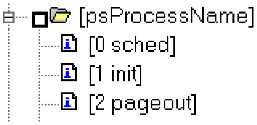
6 Click **Add** to place the selected counter on the resource list. Add all the desired resources to the list, and click **Close**.

Note: The Check Point FireWall-1 monitor can only monitor up to 25 measurements.

7 Click **OK** in the Check Point FireWall-1 dialog box to activate the monitor.

Note: You can improve the level of measurement information for the Check Point FireWall-1 monitor by enabling measurements with string values to be listed in addition to measurements with numeric values, and by enabling the name modifier which displays the string value as an identifying part of the measurement name.

In the following example of a measurement using the name modifier, the string value of ProcessName (sched) is displayed in addition to its instance ID (0):



[psProcessName]
[0 sched]
[1 init]
[2 pageout]

To enable this feature, add the following line to the *<ProTune root folder>\dat\monitors\snmp.cfg* file:

```
SNMP_show_string_nodes=1
```

Usage Notes: You can select more than one name modifier, but the first in the hierarchy will be used. Each time the Check Point FireWall-1 Add Measurements dialog box opens, the information is reread from the *snmp.cfg* file. You cannot add the same measurement twice (once with a name modifier and once without it). If you do so, an error message is issued.

20

Web Server Resource Monitoring

Using ProTune's Web Server Resource monitors, you can monitor the Apache, Microsoft IIS, and iPlanet/Netscape servers during a session step run and isolate server performance bottlenecks.

This chapter describes:

- Configuring the Apache Monitor
- Configuring the Microsoft IIS Monitor
- Configuring the iPlanet/Netscape Monitor
- Monitoring Using a Proxy Server

About Web Server Resource Monitors

Web Server Resource monitors provide you with information about the resource usage of the Apache, Microsoft IIS, and iPlanet/Netscape Web servers during session step execution. In order to obtain this data, you need to activate the online monitor for the server and specify which resources you want to measure before executing the session step.

The procedures for selecting monitor measurements and configuring the monitors vary according to server type. The following sections contain specific configuration instructions for each server type.

Note: Certain measurements or counters are especially useful for determining server performance and isolating the cause of a bottleneck during an initial stress test on a Web server. For more information about these counters, see "Useful Counters for Stress Testing" on page 526.

Configuring the Apache Monitor

To monitor an Apache server you need to know the server statistics information URL. A simple way to verify the statistics information URL is to try to view it through the browser.

The URL should be in the following format:

http://<*server name/IP address*>:<*port number*>/server-status?auto

For example:

http://stimpy:80/server-status?auto

To configure the Apache monitor:

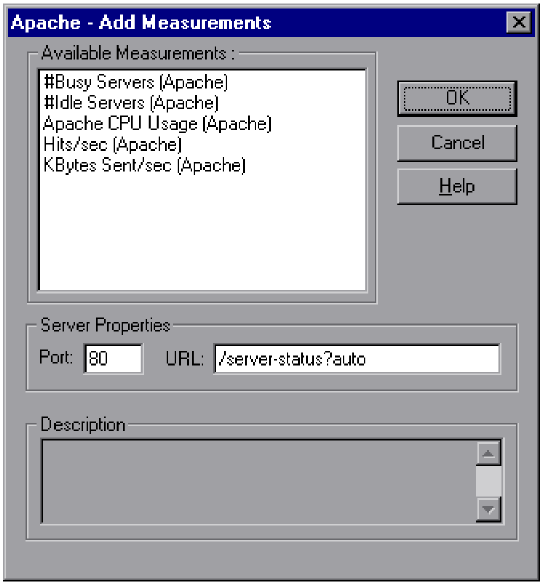
1 Click the Apache graph in the graph tree, and drag it into the right pane of the Run view.

2 Right-click the graph and choose **Add Measurement(s)**, or choose **Monitors** > **Add Online Measurement**.

3 In the Monitored Server Machines section of the Apache dialog box, click **Add** to enter the server name or IP address of the machine you want to monitor. Select the platform on which the machine runs, and click **OK**.

4 In the Resource Measurements section of the Apache dialog box, click **Add** to select the measurements that you want to monitor.

The Apache - Add Measurements dialog box opens, displaying the available measurements and server properties.



Select the required measurements. You can select multiple measurements using the **Ctrl** key.

The following table describes the measurements and server properties that can be monitored:

Measurement	Description
# Busy Servers	The number of servers in the Busy state
# Idle Servers	The number of servers in the Idle state
Apache CPU Usage	The percentage of time the CPU is utilized by the Apache server
Hits/sec	The HTTP request rate
KBytes Sent/sec	The rate at which data bytes are sent from the Web server

5. In the Server Properties section, enter the Port number and URL (without the server name), and click **OK**. The default URL is /server-status?auto.
6. Click **OK** in the Apache dialog box to activate the monitor.

Note: The default port number and URL can vary from one server to another. Please consult your Web server administrator.

To change the default server properties:

1. Open the *apache.cfg* file in the *<ProTune root folder>\dat\monitors\* directory.
2. Edit the following parameters after the Delimiter=: statement:

InfoURL server statistics information URL

ServerPort server port number

SamplingRate rate (milliseconds) at which the ProTune monitor will poll the server for the statistics information. If this value is greater than 1000, ProTune will use it as its sampling rate. Otherwise, it will use the sampling rate defined in the Monitors tab of the Options dialog box.

Note: To monitor an Apache server through a firewall, use the Web server port (by default, port 80).

Configuring the Microsoft IIS Monitor

You select measurements for the Microsoft IIS Server monitor using the MS IIS dialog box.

Note: To monitor an IIS server through a firewall, use TCP, port 139.

To configure the IIS server monitor:

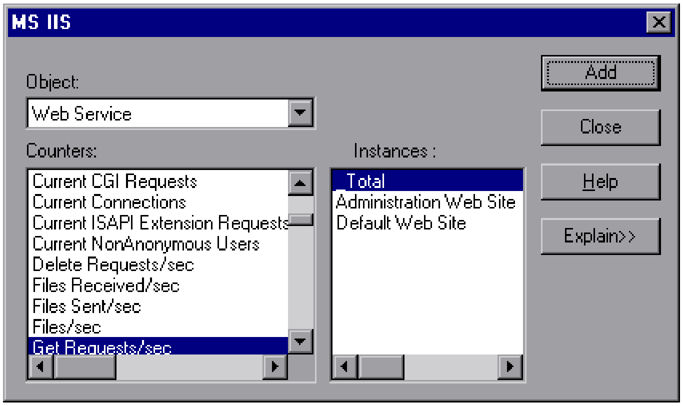
1. Click the MS IIS graph in the graph tree, and drag it into the right pane of the Run view.
2. Right-click the graph and choose **Add Measurement(s)**, or choose **Monitors** > **Add Online Measurement**.
3. In the Monitored Server Machines section of the MS IIS dialog box, click **Add** to enter the server name or IP address of the machine you want to monitor. Select the platform on which the machine runs, and click **OK**.
4. In the Resource Measurements section of the MS IIS dialog box, select the measurements you want to monitor. The following table describes the default measurements that can be monitored:

Object	Measurement	Description
Web Service	**Bytes Sent/sec**	The rate at which the data bytes are sent by the Web service
Web Service	**Bytes Received/sec**	The rate at which the data bytes are received by the Web service
Web Service	**Get Requests/sec**	The rate at which HTTP requests using the GET method are made. Get requests are generally used for basic file retrievals or image maps, though they can be used with forms.
Web Service	**Post Requests/sec**	The rate at which HTTP requests using the POST method are made. Post requests are generally used for forms or gateway requests.

Object	Measurement	Description
Web Service	**Maximum Connections**	The maximum number of simultaneous connections established with the Web service
Web Service	**Current Connections**	The current number of connections established with the Web service
Web Service	**Current NonAnonymous Users**	The number of users that currently have a non-anonymous connection using the Web service
Web Service	**Not Found Errors/sec**	The rate of errors due to requests that could not be satisfied by the server because the requested document could not be found. These are generally reported to the client as an HTTP 404 error code.
Process	**Private Bytes**	The current number of bytes that the process has allocated that cannot be shared with other processes.

Note: To change the default counters for the Microsoft IIS Server monitor, see "Changing a Monitor's Default Counters" on page 525.

5 To select additional measurements, click **Add.** A dialog box displaying the Web Service object, its counters, and instances opens.



Object: Web Service

Counters:
Current CGI Requests
Current Connections
Current ISAPI Extension Requests
Current NonAnonymous Users
Delete Requests/sec
Files Received/sec
Files Sent/sec
Files/sec
Get Requests/sec

Instances :
_Total
Administration Web Site
Default Web Site

6 Select a counter and an instance. You can select multiple counters using the **Ctrl** key. The instance is relevant only if multiple instances of the highlighted counter are running. For a description of each counter, click **Explain**>> to expand the dialog box.

7 Click **Add** to place the selected counter on the resource list. Add all the desired resources to the list, and click **Close**.

8 Click **OK** in the MS IIS dialog box to activate the monitor.

Configuring the iPlanet/Netscape Monitor

To monitor an iPlanet/Netscape server, you need to know the administration server URL. A simple way to verify the administration server URL, is to try to view it through the browser.

The URL should be in the following format:

http://<*admin_srv_name/IP address*>:<*port number*>/https-<*admin_srv_name/ IP address*>/bin/sitemon?doit

for example:

http://lazarus:12000/https-lazarus.mercury.co.il/bin/sitemon?doit

---

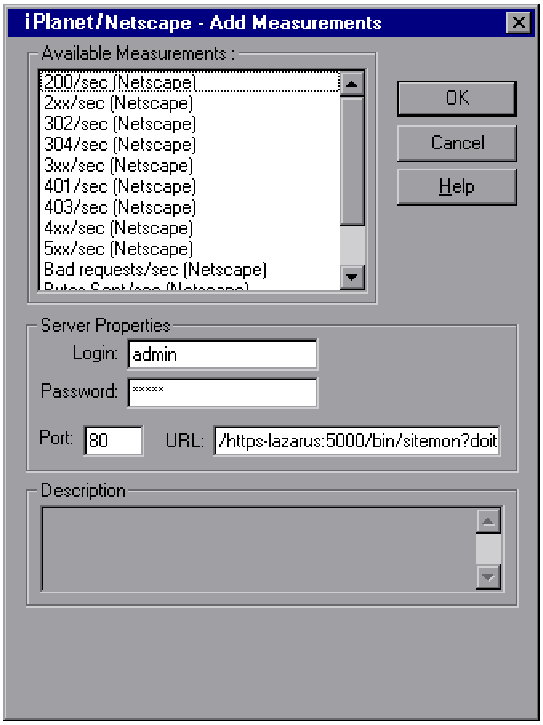
Note: In some server configurations, the URL must contain the administration server name and not the IP address.

In addition, the administration server name may differ from the iPlanet/Netscape server name.

To activate the iPlanet/Netscape monitor from the Console:

1. Click the iPlanet/Netscape graph in the graph tree, and drag it into the right pane of the Run view.
2. Right-click the graph and choose **Add Measurement(s)**, or choose **Monitors** > **Add Online Measurement**.
3. In the Monitored Server Machines section of the iPlanet/Netscape dialog box, click **Add** to enter the server name or IP address of the machine you want to monitor. Select the platform on which the machine runs, and click **OK**.
4. In the Resource Measurements section of the iPlanet/Netscape dialog box, click **Add** to select the measurements that you want to monitor.

Another iPlanet/Netscape - Add Measurements dialog box opens, displaying the available measurements and server properties:



Select the required measurements. You can select multiple measurements using the **Ctrl** key.

The following table describes the measurements and server properties that can be monitored:

Measurement	Description
200/sec	The rate of successful transactions being processed by the server
2xx/sec	The rate at which the server handles status codes in the 200 to 299 range
302/sec	The rate of relocated URLs being processed by the server

Measurement	Description
304/sec	The rate of requests for which the server tells the user to use a local copy of a URL instead of retrieving a newer version from the server
3xx/sec	The rate at which the server handles status codes in the 300 to 399 range
401/sec	The rate of unauthorized requests handled by the server
403/sec	The rate of forbidden URL status codes handled by the server
4xx/sec	The rate at which the server handles status codes in the 400 to 499 range
5xx/sec	The rate at which the server handles status codes 500 and higher
Bad requests/sec	The rate at which the server handles bad requests
Bytes sent/sec	The rate at which bytes of data are sent from the Web server
Hits/sec	The HTTP request rate
xxx/sec	The rate of all status codes (2xx-5xx) handled by the server, excluding timeouts and other errors that did return an HTTP status code

5 Fill in the Server Properties:

➤ Enter the user login name and password. The user must have administrator permissions on the server.

➤ Enter the port number and URL (without the server name), and click **OK**. The default URL is /https-<admin_server>/bin/sitemon?doit.

6 Click **OK** in the iPlanet/Netscape dialog box to activate the monitor.

Note: The default port number and URL can vary from one server to another. Please consult the Web server administrator. In some server configurations, the URL must contain the administration server name and not the IP address.

To change the default server properties:

1 Open the *Netscape.cfg* file in the *<ProTune root folder>\dat\monitors*\ directory.

2 Edit the following parameters in the [Netscape] section:

Counters number of counters that the ProTune iPlanet/Netscape monitor will show you. This value should match the number of counters defined in the file.

InfoURL server statistics information URL

ServerPort server port number

ServerLogin login name to the server

ServerPassword login password for the login name

SamplingRate rate (milliseconds) at which the ProTune monitor will poll the server for the statistics information. If this value is greater than 1000, ProTune will use it as its sampling rate. Otherwise, it will use the sampling rate defined in the Monitors tab of the Options dialog box.

Note: To monitor an iPlanet/Netscape server through a firewall, use the iPlanet/Netscape Administration server port. Configure this port during the server installation process.

Monitoring Using a Proxy Server

ProTune allows you to monitor using the Apache and Netscape monitors when there is a proxy server between the Console and the monitored server. To enable this, you must define settings in your configuration file: in *<LR root folder>\dat\monitors\apache.cfg* for the Apache monitor, or in *<LR root folder>\dat\monitors\Netscape.cfg* for the Netscape monitor.

Before defining settings, you need to determine whether you want ProTune to obtain proxy settings from your Internet Explorer connection configuration, or from the proxy settings in the configuration file.

To have ProTune read proxy settings from your Internet Explorer connection:

1. In the Proxy Settings section of the configuration file, assign **useProxy** a value of 1.
2. If the proxy requires a username, password, or domain, enter these parameters on the lines **proxyUsername**, **proxyPassword**, and **proxyDomain**.

To have ProTune read proxy settings from the configuration file:

1. In the Proxy Settings section of the configuration file, enter the proxy information on the **httpProxy** line. Use the format:
   [<protocol>=][<scheme>://]<proxy>[:<port>][[<protocol>=][<scheme>://]<proxy>[:<port>]]

   For example:
   httpProxy=http=http://my_http_proxy:8080 https=https://my_https_proxy:9000
2. If the proxy requires a username, password, or domain, enter these parameters on the lines **proxyUsername**, **proxyPassword**, and **proxyDomain**.

To have ProTune connect directly to the server (any proxy settings are ignored):

In the Proxy Settings section of the configuration file, assign **useProxy** a value of 0.

21

Web Application Server Resource Monitoring

You can monitor a Web application server during a session step run and isolate application server performance bottlenecks using ProTune's Web Application Server Resource monitors.

This chapter describes:

- Configuring the Ariba Monitor
- Configuring the ATG Dynamo Monitor
- Configuring the BroadVision Monitor
- Configuring the ColdFusion Monitor
- Configuring the Fujitsu INTERSTAGE Monitor
- Configuring the Microsoft Active Server Pages Monitor
- Configuring the Oracle9iAS HTTP Monitor
- Configuring the SilverStream Monitor
- Configuring the WebLogic (SNMP) Monitor
- Configuring the WebLogic (JMX) Monitor
- Configuring the WebSphere Monitor
- Configuring the WebSphere (EPM) Monitor

About Web Application Server Resource Monitors

Web Application Server Resource monitors provide you with information about the resource usage of the Ariba, ATG Dynamo, BroadVision, ColdFusion, Fujitsu INTERSTAGE, Microsoft ASP, Oracle9iAS HTTP, SilverStream, WebLogic (SNMP), WebLogic (JMX), and WebSphere application servers during session step execution. In order to obtain performance data, you need to activate the online monitor for the server and specify which resources you want to measure before executing the session step.

The procedures for selecting monitor measurements and configuring the monitors vary according to server type. The following sections contain specific configuration instructions for each server type.

Configuring the Ariba Monitor

You select measurements to monitor the Ariba server using the Ariba Monitor Configuration dialog box.

Note: The port you use to monitor an Ariba server through a firewall depends on the configuration of your server.

To configure the Ariba monitor:

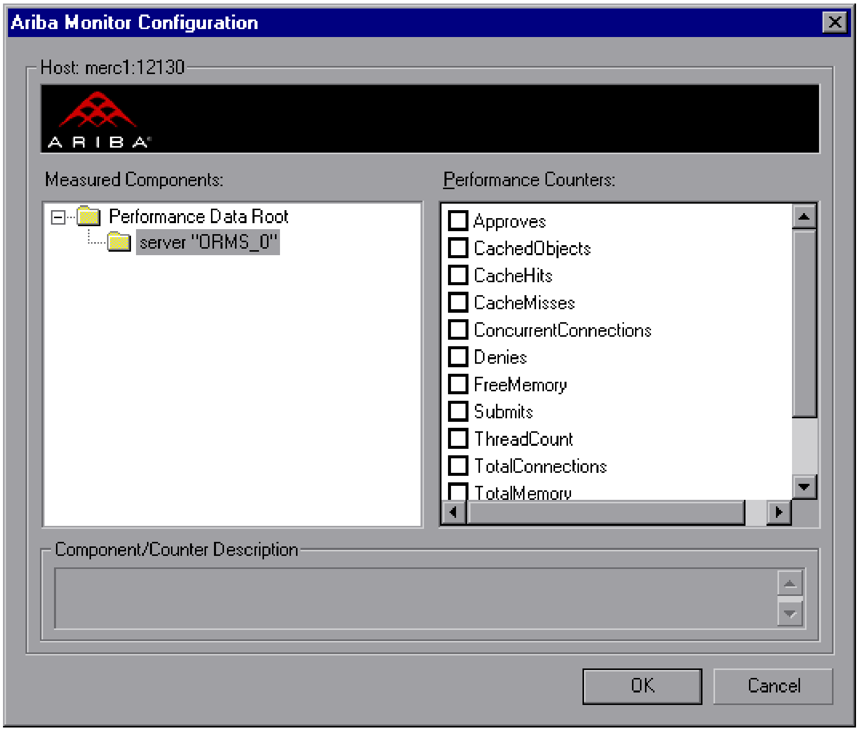
1. Click the Ariba graph in the graph tree, and drag it into the right pane of the Run view.
2. Right-click the graph and choose **Add Measurement(s)**, or choose **Monitors** > **Add Online Measurement**.
3. In the Monitored Server Machines section of the Ariba dialog box, click **Add** to enter the server name or IP address of the machine you want to monitor. Enter the server name or IP address according to the following format: *<server name>*:*<port number>*.

   For example: merc1:12130

Select the platform on which the machine runs, and click **OK**.

4 Click **Add** in the Resource Measurements section of the Ariba dialog box to select the measurements that you want to monitor. The Ariba Monitor Configuration dialog box opens.

5 Browse the Measured Components tree.



6 Check the required performance counters in the Ariba Monitor Configuration window’s right pane.

The following tables describe the counters that can be monitored:

Core Server Performance Counters

Measurement	Description
Requisitions Finished	The instantaneous reading of the length of the worker queue at the moment this metric is obtained. The longer the worker queue, the more user requests are delayed for processing.
Worker Queue Length	The instantaneous reading of the length of the worker queue at the moment this metric is obtained. The longer the worker queue, the more user requests are delayed for processing.
Concurrent Connections	The instantaneous reading of the number of concurrent user connections at the moment this metric is obtained
Total Connections	The cumulative number of concurrent user connections since Ariba Buyer was started.
Total Memory	The instantaneous reading of the memory (in KB) being used by Ariba Buyer at the moment this metric is obtained
Free Memory	The instantaneous reading of the reserved memory (in KB) that is not currently in use at the moment this metric is obtained
Up Time	The amount of time (in hours and minutes) that Ariba Buyer has been running since the previous time it was started
Number of Threads	The instantaneous reading of the number of server threads in existence at the moment this metric is obtained
Number of Cached Objects	The instantaneous reading of the number of Ariba Buyer objects being held in memory at the moment this metric is obtained
Average Session Length	The average length of the user sessions (in seconds) of all users who logged out since previous sampling time. This value indicates on average how long a user stays connected to server.

Measurement	Description
Average Idle Time	The average idle time (in seconds) for all the users who are active since previous sampling time. The idle time is the period of time between two consecutive user requests from the same user.
Approves	The cumulative count of the number of approves that happened during the sampling period. An Approve consists of a user approving one Approvable.
Submits	The cumulative count of the number of Approvables submitted since previous sampling time
Denies	The cumulative count of the number of submitted Approvables denied since previous sampling time
Object Cache Accesses	The cumulative count of accesses (both reads and writes) to the object cache since previous sampling time
Object Cache Hits	The cumulative count of accesses to the object cache that are successful (cache hits) since previous sampling time

System Related Performance Counters

Measurement	Description
Database Response Time	The average response time (in seconds) to the database requests since the previous sampling time
Buyer to DB server Traffic	The cumulative number of bytes that Ariba Buyer sent to DB server since the previous sampling time.
DB to Buyer server Traffic	The cumulative number of bytes that DB server sent to Ariba Buyer since the previous sampling time
Database Query Packets	The average number of packets that Ariba Buyer sent to DB server since the previous sampling time
Database Response Packets	The average number of packets that DB server sent to Ariba Buyer since the previous sampling time

7 Click **OK** in the Ariba Monitor Configuration dialog box, and in the Ariba dialog box, to activate the Ariba monitor.

XML Accessibility Verification

Only browsers that are XML-compatible will allow you to view the performance XML file.

To verify whether the XML file is accessible:

Display the XML file through the browser. The URL should be in the following format: http://<*server name*:*port number*>/metrics?query=getStats

For example: http://merc1:12130/metrics?query=getStats

Note: In some cases, although the browser is XML-compatible, it may still return the error: The XML page cannot be displayed. In these cases, the XML file can be accessed by the Ariba performance monitor, although it cannot be viewed by the browser.

Configuring the ATG Dynamo Monitor

The ATG Dynamo monitor uses SNMP to retrieve ATG Dynamo server statistics. You define the measurements for the ATG Dynamo monitor using the ATG Dynamo Resources dialog box.

To configure the ATG Dynamo server monitor:

1. Click the ATG Dynamo graph in the graph tree, and drag it into the right pane of the Run view.
2. Right-click the graph and choose **Add Measurement(s)**, or choose **Monitors** > **Add Online Measurement**.
3. In the Monitored Server Machines section of the ATG Dynamo dialog box, click **Add** to enter the server name or IP address of the machine you want to monitor. Select the platform on which the machine runs, and click **OK**.

Note: You need to define the port number if the ATG SNMP agent is running on a different port than the default ATG SNMP port 8870. You can define the default port for your ATG server in the configuration file, *snmp.cfg*, located in *<ProTune root folder>\dat\monitors*. For example, if the port used by the SNMP agent on your ATG system is 8888, you should edit the *snmp.cfg* file as follows:

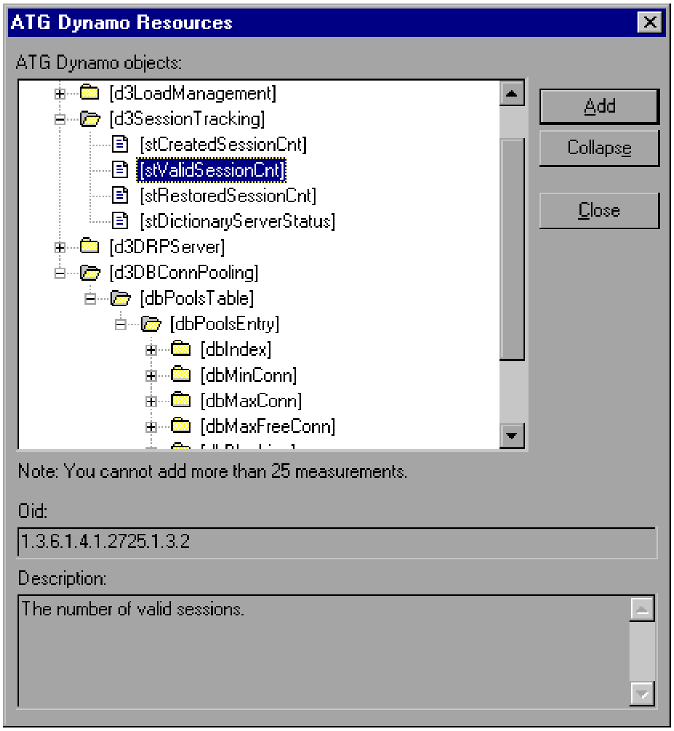
```
; ATG Dynamo
[cm_snmp_mon_atg]
port=8888
```

You can also specify a machine name and port number in the Add Machine dialog box using the following format:
<server name:port number>
For example: digi:8888

---

4 Click **Add** in the Resource Measurements section of the ATG Dynamo dialog box. The ATG Dynamo Resources dialog box opens.

5 Browse the ATG Dynamo Object tree, and select the measurements you want to monitor.



Note: You cannot add more than 25 measurements.

Oid:

1.3.6.1.4.1.2725.1.3.2

Description:

The number of valid sessions.

The following tables describe the measurements that can be monitored:

d3System

Measurement	Description
sysTotalMem	The total amount of memory currently available for allocating objects, measured in bytes
sysFreeMem	An approximation of the total amount of memory currently available for future allocated objects, measured in bytes
sysNumInfoMsgs	The number of system global info messages written

Measurement	Description
sysNumWarningMsgs	The number of system global warning messages written
sysNumErrorMsgs	The number of system global error messages written

d3LoadManagement

Measurement	Description
lmIsManager	True if the Dynamo is running a load manager
lmManagerIndex	Returns the Dynamo's offset into the list of load managing entities
lmIsPrimaryManager	True if the load manager is an acting primary manager
lmServicingCMs	True if the load manager has serviced any connection module requests in the amount of time set as the connection module polling interval
lmCMLDRPPort	The port of the connection module agent
lmIndex	A unique value for each managed entity
lmSNMPPort	The port for the entry's SNMP agent
lmProbability	The probability that the entry will be given a new session
lmNewSessions	Indicates whether or not the entry is accepting new sessions, or if the load manager is allowing new sessions to be sent to the entry. This value is inclusive of any override indicated by lmNewSessionOverride.
lmNewSessionOverride	The override set for whether or not a server is accepting new sessions

d3SessionTracking

Measurement	Description
stCreatedSessionCnt	The number of created sessions
stValidSessionCnt	The number of valid sessions

Measurement	Description
stRestoredSessionCnt	The number of sessions migrated to the server
StDictionaryServerStatus	d3Session Tracking

d3DRPServer

Measurement	Description
drpPort	The port of the DRP server
drpTotalReqsServed	Total number of DRP requests serviced
drpTotalReqTime	Total service time in msecs for all DRP requests
drpAvgReqTime	Average service time in msecs for each DRP request
drpNewessions	True if the Dynamo is accepting new sessions

d3DBConnPooling

Measurement	Description
dbPoolsEntry	A pooling service entry containing information about the pool configuration and current status
dbIndex	A unique value for each pooling service
dbPoolID	The name of the DB connection pool service
dbMinConn	The minimum number of connections pooled
dbMaxConn	The maximum number of connections pooled
dbMaxFreeConn	The maximum number of free pooled connections at a time
dbBlocking	Indicates whether or not the pool is to block out check outs
dbConnOut	Returns the number of connections checked out

Measurement	Description
dbFreeResources	Returns the number of free connections in the pool. This number refers to connections actually created that are not currently checked out. It does not include how many more connections are allowed to be created as set by the maximum number of connections allowed in the pool.
dbTotalResources	Returns the number of total connections in the pool. This number refers to connections actually created and is not an indication of how many more connections may be created and used in the pool.

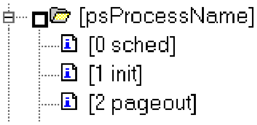
6 Click **Add** to place the selected counter on the resource list. Add all the desired resources to the list, and click **Close**.

Note: The ATG Dynamo monitor can only monitor up to 25 measurements.

7 Click **OK** in the ATG Dynamo dialog box to activate the monitor.

Note: You can improve the level of measurement information for the ATG Dynamo monitor by enabling measurements with string values to be listed in addition to measurements with numeric values, and by enabling the name modifier which displays the string value as an identifying part of the measurement name.

In the following example of a measurement using the name modifier, the string value of ProcessName (sched) is displayed in addition to its instance ID (0):



[psProcessName]
- [0 sched]
- [1 init]
- [2 pageout]

To enable this feature, add the following line to the *<ProTune root folder>\dat\monitors\snmp.cfg* file:

```
SNMP_show_string_nodes=1
```

Usage Notes: You can select more than one name modifier, but the first in the hierarchy will be used. Each time the ATG Dynamo Add Measurements dialog box opens, the information is reread from the *snmp.cfg* file. You cannot add the same measurement twice (once with a name modifier and once without it). If you do so, an error message is issued.

Configuring the BroadVision Monitor

To monitor a BroadVision server, you must grant the client permission to invoke or launch services on the server.

Note: The port you use to monitor a BroadVision server through a firewall depends on the configuration of your server.

To grant permission for a BroadVision server:

➤ Use the Iona Technologies (Orbix) command for setting user and access permission on a load generator machine:

```
chmodit [-h <host>] [-v] { <server> | -a <dir> }

{i{+,-}{user,group} | l{+,-}{user,group} }
```

➤ If you experience problems connecting to the BroadVision monitor, you may need to redefine the permissions to "all."

To invoke permission for all, enter the following command at the BroadVision server command prompt:

```
# chmodit <server> i+all
```

To launch permission for all, enter the following command at the BroadVision server command prompt:

```
# chmodit <server> l+all
```

➤ Alternatively, set ORBIX_ACL. Setting ORBIX_ACL=i+all l+all in the BroadVision/Orbix configuration file gives permission to all.

In addition, to monitor a BroadVision server, you need to have JDK 1.2 or higher installed on the Console machine.

You can install JDK 1.2 by following the download and installation instructions at the following Web site: http://java.sun.com/products/jdk/1.2/

Before activating the monitor, make sure that your Java environment is configured properly.

To configure your Java environment:

1 Open the Windows Registry.

2 The registry should contain the correct path to the Java executable (java.exe) under the JDK 1.2 installation directory. Verify the following registry key:

```
HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\App
Paths\java.exe
```

3 The registry should contain the correct path to the Java run-time environment (JRE) under the JRE 1.2 installation directory. Verify the following registry key:

HKEY_LOCAL_MACHINE\SOFTWARE\JavaSoft\Java Runtime Environment\1.2\JavaHome

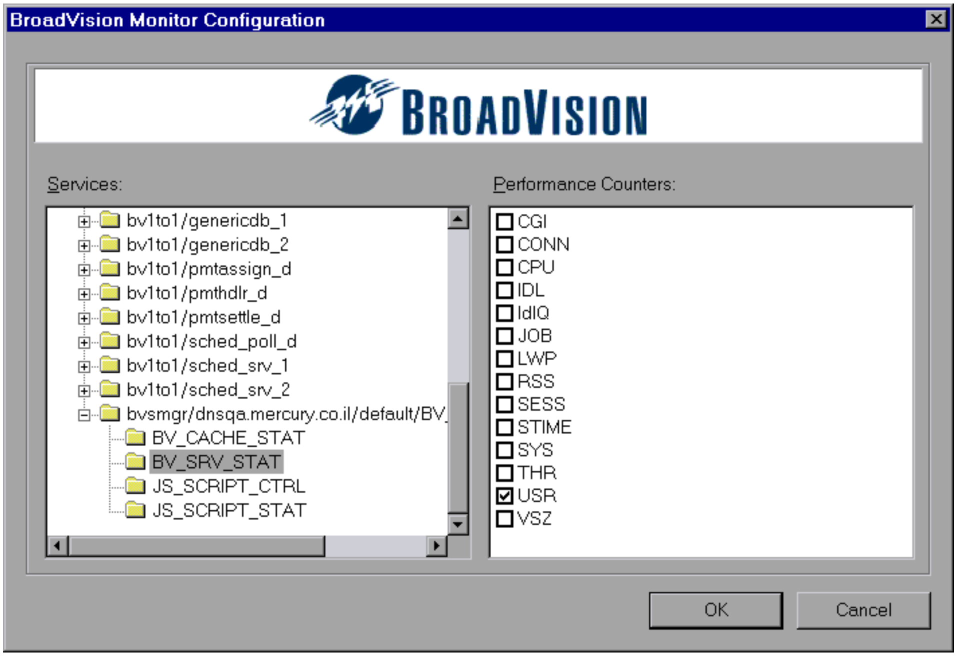
To configure the BroadVision online monitor:

1 Right-click the graph in the graph view area and choose **Add Measurement(s)**, or choose **Monitors** > **Add Online Measurement**.

2 In the Monitored Server Machines section of the BroadVision dialog box, click **Add** to enter the BroadVision server name or IP address with the port number according to the following format: *<server name>*:*<port number>*. For example: dnsqa:1221. Select the machine platform, and click **OK**.

3 Click **Add** in the Resource Measurements section of the BroadVision dialog box.

The BroadVision Monitor Configuration dialog box opens, displaying the available measurements:



Services:

- ⊞ bv1to1/genericdb_1
- ⊞ bv1to1/genericdb_2
- ⊞ bv1to1/pmtassign_d
- ⊞ bv1to1/pmthdlr_d
- ⊞ bv1to1/pmtsettle_d
- ⊞ bv1to1/sched_poll_d
- ⊞ bv1to1/sched_srv_1
- ⊞ bv1to1/sched_srv_2
- ⊟ bvsmgr/dnsqa.mercury.co.il/default/BV_
  - BV_CACHE_STAT
  - BV_SRV_STAT
  - JS_SCRIPT_CTRL
  - JS_SCRIPT_STAT

Performance Counters:

☐ CGI
☐ CONN
☐ CPU
☐ IDL
☐ IdlQ
☐ JOB
☐ LWP
☐ RSS
☐ SESS
☐ STIME
☐ SYS
☐ THR
☑ USR
☐ VSZ

4 Browse the Services tree and check the required performance counters in the BroadVision Monitor Configuration window's right pane. For a description of the performance counters, see the information below.

5 Click **OK** in the BroadVision Monitor Configuration dialog box, and in the BroadVision dialog box, to activate the BroadVision monitor.

The following table describes the servers/services that can be monitored:

Server	Multiple Instances	Description
adm_srv	No	One-To-One user administration server. There must be one.
alert_srv	No	Alert server handles direct IDL function calls to the Alert system.
bvconf_srv	No	One-To-One configuration management server. There must be one.
cmsdb	Yes	Visitor management database server.
cntdb	Yes	Content database server.
deliv_smtp_d	Yes	Notification delivery server for e-mail type messages. Each instance of this server must have its own ID, numbered sequentially starting with "1".
deliv_comp_d	No	Notification delivery completion processor.
extdbacc	Yes	External database accessor. You need at least one for each external data source.
genericdb	No	Generic database accessor handles content query requests from applications, when specifically called from the application. This is also used by the One-To-One Command Center.
hostmgr	Yes	Defines a host manager process for each machine that participates in One-To-One, but doesn't run any One-To-One servers. For example, you need a hostmgr on a machine that runs only servers. You don't need a separate hostmgr on a machine that already has one of the servers in this list.

Server	Multiple Instances	Description
g1_ofbe_srv	No	Order fulfillment back-end server.
g1_ofdb	Yes	Order fulfillment database server.
g1_om_srv	No	Order management server.
pmtassign_d	No	The payment archiving daemon routes payment records to the archives by periodically checking the invoices table, looking for records with completed payment transactions, and then moving those records into an archive table.
pmthdlr_d	Yes	For each payment processing method, you need one or more authorization daemons to periodically acquire the authorization when a request is made.
pmtsettle_d	Yes	Payment settlement daemon periodically checks the database for orders of the associated payment processing method that need to be settled, and then authorizes the transactions.
sched_poll_d	No	Notification schedule poller scans the database tables to determine when a notification must be run.
sched_srv	Yes	Notification schedule server runs the scripts that generate the visitor notification messages.

Performance Counters

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

The following section describes all the available counters under each group. Note that the same group can have a different number of counters, depending on the service.

Counter groups:

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

BV_DB_STAT

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

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

BV_SRV_CTRL

- SHUTDOWN

NS_STAT

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

- Bind
- List
- New
- Rebnd
- Rsolv
- Unbnd

BV_SRV_STAT

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

- **HOST -** Host machine running the process.
- **ID -** Instance of the process (of which multiple can be configured in the *bv1to1.conf* file), or engine ID of the Interaction Manager.
- **CGI -** Current count of CGI requests processed.
- **CONN -** Current count of connections.
- **CPU -** CPU percentage consumed by this process. If a process is using most of the CPU time, consider moving it to another host, or creating an additional process, possibly running on another machine. Both of these specifications are done in the *bv1to1.conf* file. The CPU % reported is against a single processor. If a server is taking up a whole CPU on a 4 processor machine, this statistic will report 100%, while the Windows Task Manager will report 25%. The value reported by this statistic is consistent with "% Processor Time" on the Windows Performance Monitor.
- **GROUP -** Process group (which is defined in the *bv1to1.conf* file), or Interaction Manager application name.

- **STIME -** Start time of server. The start times should be relatively close. Later times might be an indication that a server crashed and was automatically restarted.
- **IDL -** Total count of IDL requests received, not including those to the monitor.
- **IdlQ**
- **JOB**
- **LWP -** Number of light-weight processes (threads).
- **RSS -** Resident memory size of server process (in kilobytes).
- **STIME -** System start time.
- **SESS -** Current count of sessions.
- **SYS -** Accumulated system mode CPU time (seconds).
- **THR -** Current count of threads.
- **USR -** Accumulated user mode CPU time (seconds).
- **VSZ -** Virtual memory size of server process (in kilobytes). If a process is growing in size, it probably has a memory leak. If it is an Interaction Manager process, the culprit is most likely a component or dynamic object (though Interaction Manager servers do grow and shrink from garbage collection during normal use).

BV_CACHE_STAT

Monitors the request cache status.

The available counters for each request are:

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

Cache Metrics

Cache metrics are available for the following items:

- **AD**
- **ALERTSCHED -** Notification schedules are defined in the BV_ALERTSCHED and BV_MSGSCHED tables. They are defined by the One-To-One Command Center user or by an application.
- **CATEGORY_CONTENT**
- **DISCUSSION -** The One-To-One discussion groups provide moderated system of messages and threads of messages aligned to a particular topic. Use the Discussion group interfaces for creating, retrieving and deleting individual messages in a discussion group. To create, delete, or retrieve discussion groups, use the generic content management API. The BV_DiscussionDB object provides access to the threads and messages in the discussion group database.
- **EXT_FIN_PRODUCT**
- **EDITORIAL -** Using the Editorials content module, you can point cast and community cast personalized editorial content, and sell published text on your One-To-One site. You can solicit editorial content, such as investment reports and weekly columns, from outside authors and publishers, and create your own articles, reviews, reports, and other informative media. In addition to text, you can use images, sounds, music, and video presentations as editorial content.
- **INCENTIVE -** Contains sales incentives
- **MSGSCHED -** Contains the specifications of visitor-message jobs. Notification schedules are defined in the BV_ALERTSCHED and BV_MSGSCHED tables. They are defined by the One-To-One Command Center user or by an application.
- **MSGSCRIPT -** Contains the descriptions of the JavaScripts that generate visitor messages and alert messages. Contains the descriptions of the JavaScripts that generate targeted messages and alert messages. Use the Command Center to add message script information to this table by selecting the Visitor Messages module in the Notifications group. For more information, see the Command Center User's Guide.

- **PRODUCT -** BV_PRODUCT contains information about the products that a visitor can purchase.
- **QUERY -** BV_QUERY contains queries.
- **SCRIPT -** BV_SCRIPT contains page scripts.
- **SECURITIES**
- **TEMPLATE -** The Templates content module enables you to store in the content database any BroadVision page templates used on your One-To-One site. Combining BroadVision page templates with BroadVision dynamic objects in the One-To-One Design Center application is one way for site developers to create One-To-One Web sites. If your developers use these page templates, you can use the Command Center to enter and manage them in your content database. If your site doesn't use BroadVision page template, you will not use this content module.

JS_SCRIPT_CTRL

- CACHE
- DUMP
- FLUSH
- METER
- TRACE

JS_SCRIPT_STAT

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

Configuring the ColdFusion Monitor

You select measurements to monitor the ColdFusion server using the ColdFusion dialog box.

Note: The ColdFusion monitor works via HTTP and supports UNIX platforms. If you want to monitor the ColdFusion server on Windows platforms, you can also use the Windows Resource monitor.

To set up the ColdFusion monitor environment:

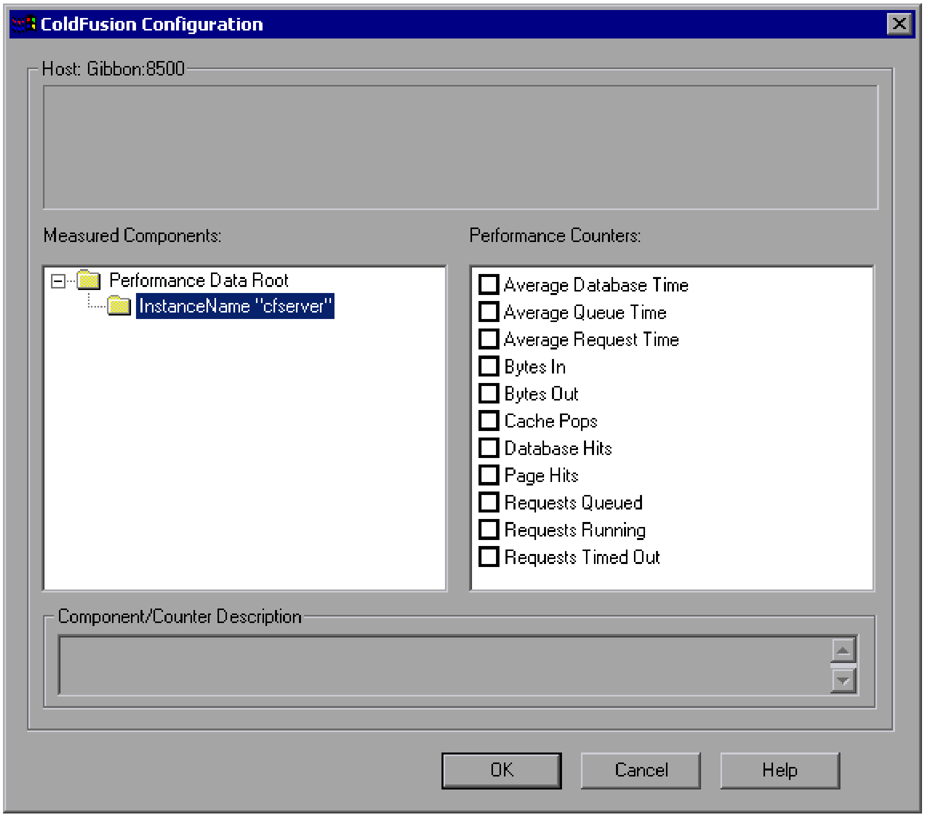
Copy the *<ProTune installation>\dat\monitors\perfmon.cfm* file into the *<ColdFusion Home>\cfide\administrator* directory. By default, the ColdFusion monitor checks for the *<ColdFusion Home>\cfide\administrator\perfmon.cfm* file.

Note: The port you use to monitor a ColdFusion server through a firewall depends on the configuration of your server.

To configure the ColdFusion monitor:

1. Click the ColdFusion graph in the graph tree, and drag it into the right pane of the Run view.
2. Right-click the graph and choose **Add Measurement(s)**, or choose **Monitors** > **Add Online Measurement**.
3. In the Monitored Server Machines section of the ColdFusion dialog box, click **Add** to enter the server name or IP address of the machine you want to monitor. Select the platform on which the machine runs, and click **OK**.
4. In the Resource Measurements section of the ColdFusion dialog box, click **Add** to select the measurements that you want to monitor. The ColdFusion Monitor Configuration dialog box displays the available measurements.

5 Browse the Measured Components tree.



ColdFusion Configuration

Host: Gibbon:8500

Measured Components:

- Performance Data Root
  - InstanceName "cfserver"

Performance Counters:

- □ Average Database Time
- □ Average Queue Time
- □ Average Request Time
- □ Bytes In
- □ Bytes Out
- □ Cache Pops
- □ Database Hits
- □ Page Hits
- □ Requests Queued
- □ Requests Running
- □ Requests Timed Out

Component/Counter Description

6 Check the required performance counters in the ColdFusion Monitor Configuration window’s right pane.

The following table describes the default counters that can be measured:

Measurement	Description
Avg. Database Time (msec)	The running average of the amount of time, in milliseconds, that it takes ColdFusion to process database requests.
Avg. Queue Time (msec)	The running average of the amount of time, in milliseconds, that requests spent waiting in the ColdFusion input queue before ColdFusion began to process the request.
Avg Req Time (msec)	The running average of the total amount of time, in milliseconds, that it takes ColdFusion to process a request. In addition to general page processing time, this value includes both queue time and database processing time.
Bytes In/sec	The number of bytes per second sent to the ColdFusion server.
Bytes Out/sec	The number of bytes per second returned by the ColdFusion server.
Cache Pops	Cache pops.
Database Hits/sec	This is the number of database hits generated per second by the ColdFusion server.
Page Hits/sec	This is the number of Web pages processed per second by the ColdFusion server.
Queued Requests	The number of requests currently waiting to be processed by the ColdFusion server.
Running Requests	The number of requests currently being actively processed by the ColdFusion server.
Timed Out Requests	The number of requests that timed out due to inactivity timeouts.

7 Click **OK** in the ColdFusion Monitor Configuration dialog box, and in the ColdFusion dialog box, to activate the ColdFusion monitor.

Configuring the Fujitsu INTERSTAGE Monitor

The Fujitsu INTERSTAGE monitor uses SNMP to retrieve Fujitsu INTERSTAGE server statistics. You define the measurements for the Fujitsu INTERSTAGE monitor using the Fujitsu INTERSTAGE SNMP Resources dialog box.

To configure the Fujitsu INTERSTAGE server monitor:

1 Click the Fujitsu INTERSTAGE graph in the graph tree, and drag it into the right pane of the Run view.

2 Right-click the graph and choose **Add Measurement(s)**, or choose **Monitors** > **Add Online Measurement**.

3 In the Monitored Server Machines section of the Fujitsu INTERSTAGE dialog box, click **Add** to enter the server name or IP address of the machine you want to monitor. Select the platform on which the machine runs, and click **OK**.

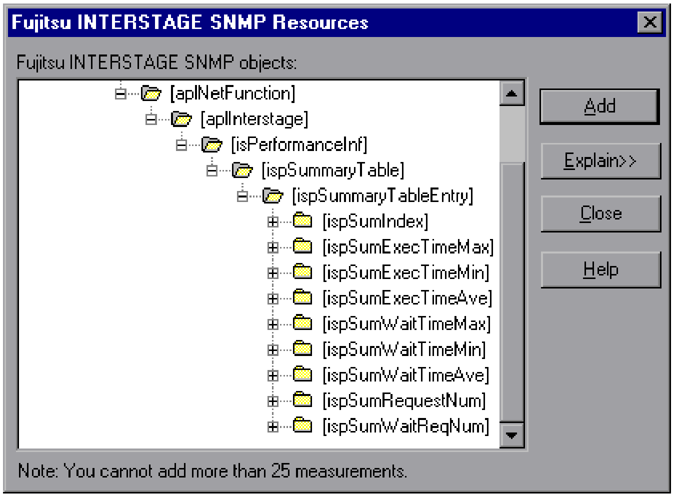
Note: You need to define the port number if the Fujitsu INTERSTAGE SNMP agent is running on a different port than the default SNMP port 161. Enter the following information in the Add Machine dialog box:
<*server name*:*port number*>
For example: digi:8888

In addition, you can define the default port for your Fujitsu INTERSTAGE server in the configuration file, *snmp.cfg*, located in *<ProTune root folder>\dat\monitors*. For example, if the port used by the SNMP agent on your Fujitsu INTERSTAGE system is 8888, you should edit the *snmp.cfg* file as follows:

```
; Fujitsu INTERSTAGE
[cm_snmp_mon_isp]
port=8888
```

4 Click **Add** in the Resource Measurements section of the Fujitsu INTERSTAGE dialog box. The Fujitsu INTERSTAGE SNMP Resources dialog box opens.

5 Browse the Fujitsu INTERSTAGE SNMP Object tree, and select the measurements you want to monitor.



Note: You cannot add more than 25 measurements.

The following tables describe the measurements that can be monitored:

Measurement	Description
IspSumObjectName	The object name of the application for which performance information is measured
IspSumExecTimeMax	The maximum processing time of the application within a certain period of time
IspSumExecTimeMin	The minimum processing time of the application within a certain period of time
IspSumExecTimeAve	The average processing time of the application within a certain period of time
IspSumWaitTimeMax	The maximum time required for INTERSTAGE to start an application after a start request is issued
IspSumWaitTimeMin	The minimum time required for INTERSTAGE to start an application after a start request is issued
IspSumWaitTimeAve	The average time required for INTERSTAGE to start an application after a start request is issued

Measurement	Description
IspSumRequestNum	The number of requests to start an application
IspSumWaitReqNum	The number of requests awaiting application activation

6 Click **Add** to place the selected counter on the resource list. Add all the desired resources to the list, and click **Close**.

Note: The Fujitsu INTERSTAGE monitor can only monitor up to 25 measurements.

7 Click **OK** in the Fujitsu INTERSTAGE dialog box to activate the monitor.

Configuring the Microsoft Active Server Pages Monitor

You select measurements to monitor the Microsoft ASP application server using the MS Active Server Pages dialog box.

Note: To monitor an ASP server through a firewall, use TCP, port 139.

To configure the ASP monitor:

1 Click the MS Active Server Pages graph in the graph tree, and drag it into the right pane of the Run view.

2 Right-click the graph and choose **Add Measurement(s)**, or choose **Monitors** > **Add Online Measurement**.

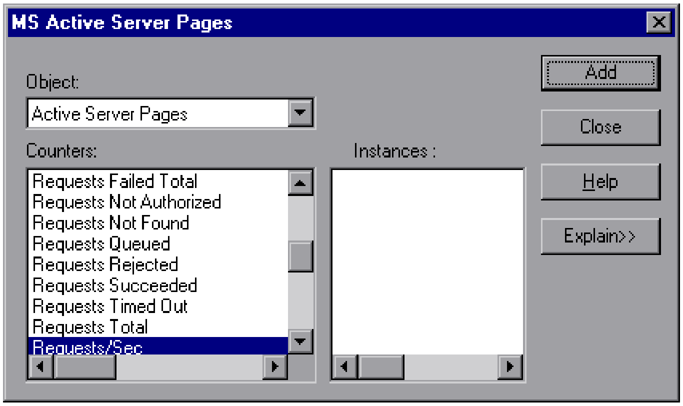
3 In the Monitored Server Machines section of the MS Active Server Pages dialog box, click **Add** to enter the server name or IP address of the machine you want to monitor. Select the platform on which the machine runs, and click **OK**.

4 In the Resource Measurements section of the MS Active Server Pages dialog box, select the measurements you want to monitor. The following table describes the default counters that can be monitored:

Measurement	Description
Errors per Second	The number of errors per second.
Requests Wait Time	The number of milliseconds the most recent request was waiting in the queue.
Requests Executing	The number of requests currently executing.
Requests Queued	The number of requests waiting in the queue for service.
Requests Rejected	The total number of requests not executed because there were insufficient resources to process them.
Requests Not Found	The number of requests for files that were not found.
Requests/sec	The number of requests executed per second.
Memory Allocated	The total amount of memory, in bytes, currently allocated by Active Server Pages.
Errors During Script Run-Time	The number of failed requests due to run-time errors.
Sessions Current	The current number of sessions being serviced.
Transactions/sec	The number of transactions started per second.

Note: To change the default counters for the Microsoft ASP monitor, see "Changing a Monitor's Default Counters" on page 525.

5 To select additional measurements, click **Add.** A dialog box displaying the Active Server Pages object, its counters, and instances opens.



Object: Active Server Pages

6 Select a counter and instance. You can select multiple counters using the **Ctrl** key. The instance is relevant only if multiple instances of the highlighted counter are running. For a description of each counter, click **Explain**>> to expand the dialog box.

7 Click **Add** to place the selected counter on the resource list. Add all the desired resources to the list, and click **Close**.

8 Click **OK** in the MS Active Server Pages dialog box to activate the monitor.

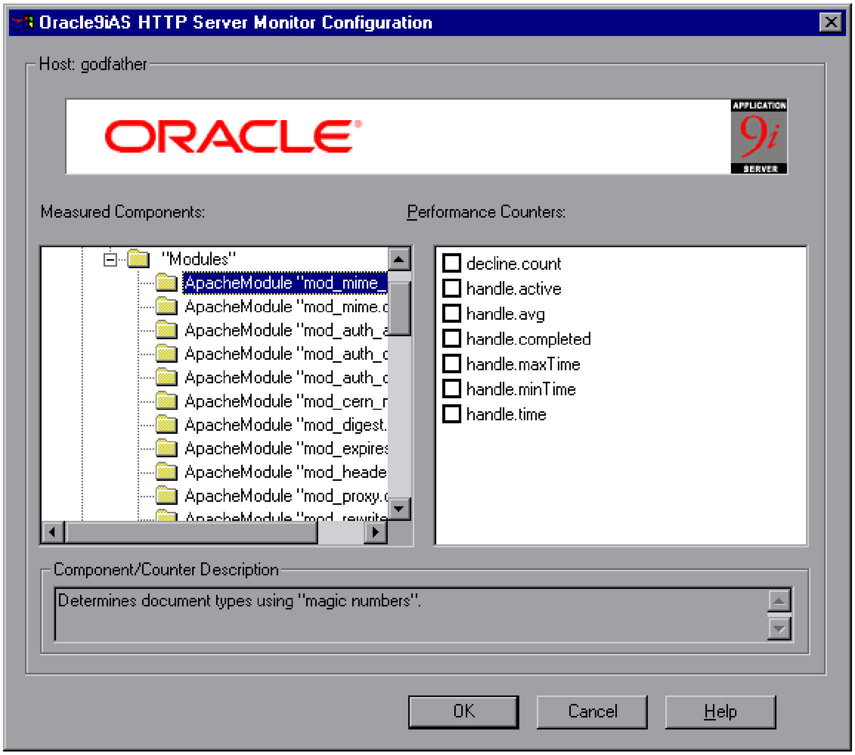
Configuring the Oracle9iAS HTTP Monitor

You select measurements to monitor the Oracle9iAS HTTP server using the Oracle HTTP Server Monitor Configuration dialog box. Note that you must start running the Oracle9iAS HTTP server before you begin selecting the measurements you want to monitor.

Note: The port you use to monitor an Oracle9iAS HTTP server through a firewall depends on the configuration of your server.

To configure the Oracle9iAS HTTP monitor:

1. Click the Oracle9iAS HTTP graph in the graph tree, and drag it into the right pane of the Run view.
2. Right-click the graph and choose **Add Measurement(s)**, or choose **Monitors** > **Add Online Measurement**.
3. In the Monitored Server Machines section of the Oracle9iAS HTTP Server dialog box, click **Add** to enter the server name or IP address of the machine you want to monitor. Select any platform, and click **OK**.
4. Click **Add** in the Resource Measurements section of the Oracle9iAS HTTP Server dialog box to select the measurements that you want to monitor. The Oracle HTTP Server Monitor Configuration dialog box opens, displaying the counters that can be monitored.
5. Browse the Measured Components tree.



Oracle9iAS HTTP Server Monitor Configuration

Host: godfather

Measured Components:

"Modules"
ApacheModule "mod_mime_
ApacheModule "mod_mime.c
ApacheModule "mod_auth_a
ApacheModule "mod_auth_c
ApacheModule "mod_auth_c
ApacheModule "mod_cern_r
ApacheModule "mod_digest.
ApacheModule "mod_expires
ApacheModule "mod_heade
ApacheModule "mod_proxy.c
ApacheModule "mod_rewrite

Performance Counters:

□ decline.count
□ handle.active
□ handle.avg
□ handle.completed
□ handle.maxTime
□ handle.minTime
□ handle.time

Component/Counter Description

Determines document types using "magic numbers".

6 Check the required machine processing counters, or application server performance counters and modules, in the Oracle HTTP Server Monitor Configuration window's right pane.

The following table describes some of the modules that can be monitored:

Measurement	Description
mod_mime.c	Determines document types using file extensions
mod_mime_magic.c	Determines document types using "magic numbers"
mod_auth_anon.c	Provides anonymous user access to authenticated areas
mod_auth_dbm.c	Provides user authentication using DBM files
mod_auth_digest.c	Provides MD5 authentication
mod_cern_meta.c	Supports HTTP header metafiles
mod_digest.c	Provides MD5 authentication (deprecated by mod_auth_digest)
mod_expires.c	Applies Expires: headers to resources
mod_headers.c	Adds arbitrary HTTP headers to resources
mod_proxy.c	Provides caching proxy abilities
mod_rewrite.c	Provides powerful URI-to-filename mapping using regular expressions
mod_speling.c	Automatically corrects minor typos in URLs
mod_info.c	Provides server configuration information
mod_status.c	Displays server status
mod_usertrack.c	Provides user tracking using cookies
mod_dms.c	Provides access to DMS Apache statistics
mod_perl.c	Allows execution of perl scripts
mod_fastcgi.c	Supports CGI access to long-lived programs
mod_ssl.c	Provides SSL support
mod_plsql.c	Handles requests for Oracle stored procedures

Measurement	Description
mod_isapi.c	Provides Windows ISAPI extension support
mod_setenvif.c	Sets environment variables based on client information
mod_actions.c	Executes CGI scripts based on media type or request method
mod_imap.c	Handles imagemap files
mod_asis.c	Sends files that contain their own HTTP headers
mod_log_config.c	Provides user-configurable logging replacement for mod_log_common
mod_env.c	Passes environments to CGI scripts
mod_alias.c	Maps different parts of the host file system in the document tree, and redirects URLs
mod_userdir.c	Handles user home directories
mod_cgi.c	Invokes CGI scripts
mod_dir.c	Handles the basic directory
mod_autoindex.c	Provides automatic directory listings
mod_include.c	Provides server-parsed documents
mod_negotiation.c	Handles content negotiation
mod_auth.c	Provides user authentication using text files
mod_access.c	Provides access control based on the client hostname or IP address
mod_so.c	Supports loading modules (.so on UNIX, .dll on Win32) at run-time
mod_oprocmgr.c	Monitors JServ processes and restarts them if they fail
mod_jserv.c	Routes HTTP requests to JServ server processes. Balances load across multiple JServs by distributing new requests in round-robin order

Measurement	Description
mod_ose.c	Routes requests to the JVM embedded in Oracle's database server
http_core.c	Handles requests for static Web pages

The following table describes the counters that can be monitored:

Measurement	Description
handle.minTime	The minimum time spent in the module handler
handle.avg	The average time spent in the module handler
handle.active	The number of threads currently in the handle processing phase
handle.time	The total amount of time spent in the module handler
handle.completed	The number of times the handle processing phase was completed
request.maxTime	The maximum amount of time required to service an HTTP request
request.minTime	The minimum amount of time required to service an HTTP request
request.avg	The average amount of time required to service an HTTP request
request.active	The number of threads currently in the request processing phase
request.time	The total amount of time required to service an HTTP request
request.completed	The number of times the request processing phase was completed
connection.maxTime	The maximum amount of time spent servicing any HTTP connection
connection.minTime	The minimum amount of time spent servicing any HTTP connection

Measurement	Description
connection.avg	The average amount of time spent servicing HTTP connections
connection.active	The number of connections with currently open threads
connection.time	The total amount of time spent servicing HTTP connections
connection.completed	The number of times the connection processing phase was completed
numMods.value	The number of loaded modules
childFinish.count	The number of times the Apache parent server started a child server, for any reason
childStart.count	The number of times "children" finished "gracefully."There are some ungraceful error/crash cases that are not counted in childFinish.count
Decline.count	The number of times each module declined HTTP requests
internalRedirect.count	The number of times that any module passed control to another module using an "internal redirect"
cpuTime.value	The total CPU time utilized by all processes on the Apache server (measured in CPU milliseconds)
heapSize.value	The total heap memory utilized by all processes on the Apache server (measured in kilobytes)
pid.value	The process identifier of the parent Apache process
upTime.value	The amount of time the server been running (measured in milliseconds)

7 Click **OK** in the Oracle HTTP Server Monitor Configuration dialog box, and in the Oracle9iAS HTTP Server dialog box, to activate the Oracle9iAS HTTP monitor.

Configuring the SilverStream Monitor

To monitor a SilverStream server you need to know the server statistics information URL. A simple way to verify the statistics URL is to access it from a browser.

The URL should be in the following format:

http://<*server_name/IP_address*>:<*port_number*>/SilverStream/Statistics

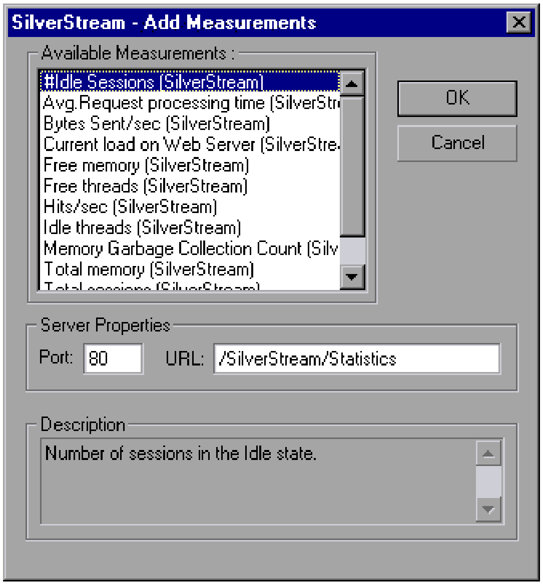
for example:

http://199.203.78.57:80/SilverStream/Statistics

To configure the SilverStream monitor:

1. Click the SilverStream graph in the graph tree, and drag it into the right pane of the Run view.
2. Right-click the graph and choose **Add Measurement(s)**, or choose **Monitors** > **Add Online Measurement**.
3. In the Monitored Server Machines section of the SilverStream dialog box, click **Add** to enter the server name or IP address of the machine you want to monitor. Select the platform on which the machine runs, and click **OK**.
4. In the Resource Measurements section of the SilverStream dialog box, click **Add** to select the measurements that you want to monitor.

A dialog box displaying the available measurements and server properties opens.



Available Measurements :

- #Idle Sessions (SilverStream)
- Avg.Request processing time (SilverStr
- Bytes Sent/sec (SilverStream)
- Current load on Web Server (SilverStre
- Free memory (SilverStream)
- Free threads (SilverStream)
- Hits/sec (SilverStream)
- Idle threads (SilverStream)
- Memory Garbage Collection Count (Silv
- Total memory (SilverStream)
- Total sessions (SilverStream)

Server Properties

Port: 80 URL: /SilverStream/Statistics

Description

Number of sessions in the Idle state.

Select the required measurements. You can select multiple measurements using the **Ctrl** key.

The following table describes the measurements and server properties that can be monitored:

Measurement	Description
#Idle Sessions	The number of sessions in the Idle state.
Avg. Request processing time	The average request processing time.
Bytes Sent/sec	The rate at which data bytes are sent from the Web server.
Current load on Web Server	The percentage of load utilized by the SilverStream server, scaled at a factor of 25.
Hits/sec	The HTTP request rate.
Total sessions	The total number of sessions.

Measurement	Description
Free memory	The total amount of memory in the Java Virtual Machine currently available for future allocated objects.
Total memory	The total amount of memory in the Java Virtual Machine.
Memory Garbage Collection Count	The total number of times the JAVA Garbage Collector has run since the server was started.
Free threads	The current number of threads not associated with a client connection and available for immediate use.
Idle threads	The number of threads associated with a client connection, but not currently handling a user request.
Total threads	The total number of client threads allocated.

5 In the Server Properties section, enter the Port number and URL (without the server name), and click **OK**. The default URL is /SilverStream/Statistics.

6 Click **OK** in the SilverStream dialog box to activate the monitor.

Note: The default port number and URL can vary from one server to another. Please consult the Web server administrator.

To change the default server properties:

1. Open the *SilverStream.cfg* file in the *<ProTune root folder>\dat\ monitors* directory.

2. Edit the following parameters at the end of the file:

InfoURL	server statistics information URL
ServerPort	server port number
SamplingRate	rate (milliseconds) at which the ProTune monitor will poll the server for the statistics information. If this value is greater than 1000, ProTune will use it as its sampling rate. Otherwise, it will use the sampling rate defined in the Monitors tab of the Options dialog box.

Note: To monitor a SilverStream server through a firewall, use the Web server port (by default, port 80).

Configuring the WebLogic (SNMP) Monitor

The WebLogic (SNMP) monitor uses SNMP to retrieve server statistics. To use this monitor, you must make sure that a version prior to WebLogic 6.0 is installed on your server, and that the SNMP agent is installed and activated on the server. For instructions on installing the SNMP agent, see http://www.weblogic.com/docs51/admindocs/snmpagent.html.

Note: To monitor a WebLogic (SNMP) server, use port 161 or 162, depending on the configuration of the agent.

To configure the WebLogic (SNMP) monitor:

1. Click the WebLogic (SNMP) graph in the graph tree, and drag it into the right pane of the Run view.

2 Right-click the graph and choose **Add Measurement(s)**, or choose **Monitors** > **Add Online Measurement**.

3 In the Monitored Server Machines section of the WebLogic (SNMP) dialog box, click **Add** to enter the server name or IP address of the machine you want to monitor. Select the platform on which the machine runs, and click **OK**.

Note: You need to define the port number if the WebLogic SNMP agent is running on a different port than the default SNMP port. Enter the following information in the Add Machine dialog box:
<*server name*:*port number*>
For example: digi:8888

In addition, you can define the default port for your WebLogic server in the configuration file, *snmp.cfg*, located in *<ProTune root folder>\dat\monitors*. For example, if the port used by the SNMP agent on your WebLogic server is 8888, you should edit the *snmp.cfg* file as follows:

```
; WebLogic
[cm_snmp_mon_isp]
port=8888
```

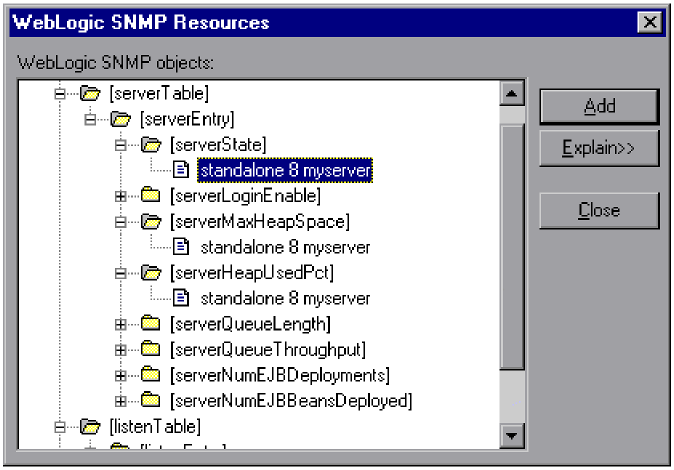
---

4 In the Resource Measurements section of the WebLogic (SNMP) dialog box, click **Add** to select the measurements that you want to monitor. The WebLogic SNMP Resources dialog box displays the available measurements.

Note: The WebLogic (SNMP) monitor can only monitor up to 25 measurements.

---

5 Browse the WebLogic SNMP Objects tree.



6 To measure an object, select it, and click **Add**. The following tables describe the measurements and server properties that can be monitored:

Server Table

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

Measurement	Description
ServerState	The state of the WebLogic server, as inferred by the SNMP agent. *Up* implies that the agent can contact the server. *Down* implies that the agent cannot contact the server.
ServerLoginEnable	This value is true if client logins are enabled on the server.
ServerMaxHeapSpace	The maximum heap size for this server, in KB
ServerHeapUsedPct	The percentage of heap space currently in use on the server

Measurement	Description
ServerQueueLength	The current length of the server execute queue
ServerQueueThroughput	The current throughput of execute queue, expressed as the number of requests processed per second
ServerNumEJBDeployment	The total number of EJB deployment units known to the server
ServerNumEJBBeansDeployed	The total number of EJB beans actively deployed on the server

Listen Table

The Listen Table is the set of protocol, IP address, and port combinations on which servers are listening. There will be multiple entries for each server: one for each protocol, ipAddr, port combination. If clustering is used, the clustering-related MIB objects will assume a higher priority.

Measurement	Description
ListenPort	Port number.
ListenAdminOK	True if admin requests are allowed on this (protocol, ipAddr, port); otherwise false
ListenState	Listening if the (protocol, ipAddr, port) is enabled on the server; not Listening if it is not. The server may be listening but not accepting new clients if its server Login Enable state is false. In this case, existing clients will continue to function, but new ones will not.

ClassPath Table

The ClassPath Table is the table of classpath elements for Java, WebLogic (SNMP) server, and servlets. There are multiple entries in this table for each server. There may also be multiple entries for each path on a server. If clustering is used, the clustering-related MIB objects will assume a higher priority.

Measurement	Description
CPType	The type of CP element: Java, WebLogic, servlet. A Java CPType means the cpElement is one of the elements in the normal Java classpath. A WebLogic CPType means the cpElement is one of the elements in weblogic.class.path. A servlet CPType means the cpElement is one of the elements in the dynamic servlet classpath.
CPIndex	The position of an element within its path. The index starts at 1.

7 After selecting and adding the required objects, click **Close**.

8 Click **OK** in the WebLogic (SNMP) dialog box to activate the monitor.

Configuring the WebLogic (JMX) Monitor

The BEA WebLogic (JMX) monitor uses the Java JMX interface to access run-time MBeans on the server. An MBean is a container that holds the performance data.

Before using the WebLogic (JMX) monitor, you must install Java 1.3 or later on the Console machine. If Java 1.3 or later is already installed, but is not the default Java version being used, specify the full path to the updated version. You specify the path in the *<ProTune root folder>\dat\monitors\WebLogicMon.ini* file. Edit the JVM entry in the [WebLogicMon] section. For example:

```
JVM="E:\Program Files\JavaSoft\JRE\1.3.1\bin\javaw.exe
```

Note: To use the WebLogic (JMX) monitor, you must make sure that WebLogic 6.0 or above is installed on your server.

Setting Permissions for Monitoring

You must set certain permissions for a user to be able to monitor MBeans.

To set permissions:

1. Open the WebLogic console (http://<host:port>/console).
2. In the tree on the left, select **Security** > **ACLs**.

   If you are working with the WebLogic 6.1 console, click **Create a new ACL...** in the screen on the right.
3. In the New ACL Name box, type weblogic.admin.mbean, and click **Create**.

   If you are working with the WebLogic 6.1 console, click **Add a new Permission...** in the screen on the right.
4. In the New Permission box (or Permission box, in the WebLogic 6.1 console), type access. In the WebLogic 6.0 console, click **Create**.
5. In the Users box and Groups box, enter the name of any user or group you want to use for monitoring.
6. Click **Grant Permission** in the WebLogic 6.0 console. In the WebLogic 6.1 console, click **Apply**.

Loading Classes from the Server

The WebLogic (JMX) monitor utilizes a built-in server called the ClasspathServlet to load classes directly and automatically from the server. The advantages of this are easy installation and version independence. The disadvantages are a slight decrease in performance when loading classes for the first time (due to the size of the servlet), and the possibility of the servlet becoming disabled.

If the servlet is disabled, or if you do not want to use the servlet, you can load classes directly from the file system.

To load classes directly from the file system:

1. Copy the *weblogic.jar* file from the application server install folder (under the lib folder) to *<ProTune root folder>\classes*.
2. If the classes file is not located in the default *<ProTune root folder>* folder, you need to specify the full path to it in the *<ProTune root folder>\dat\monitors\WebLogicMon.ini* file. In this file, change the line Weblogic=weblogic.jar to Weblogic=*<full path to weblogic.jar>*.

Configuring the WebLogic (JMX) Monitor

You select measurements to monitor the WebLogic (JMX) application server using the BEA WebLogic Monitor Configuration dialog box.

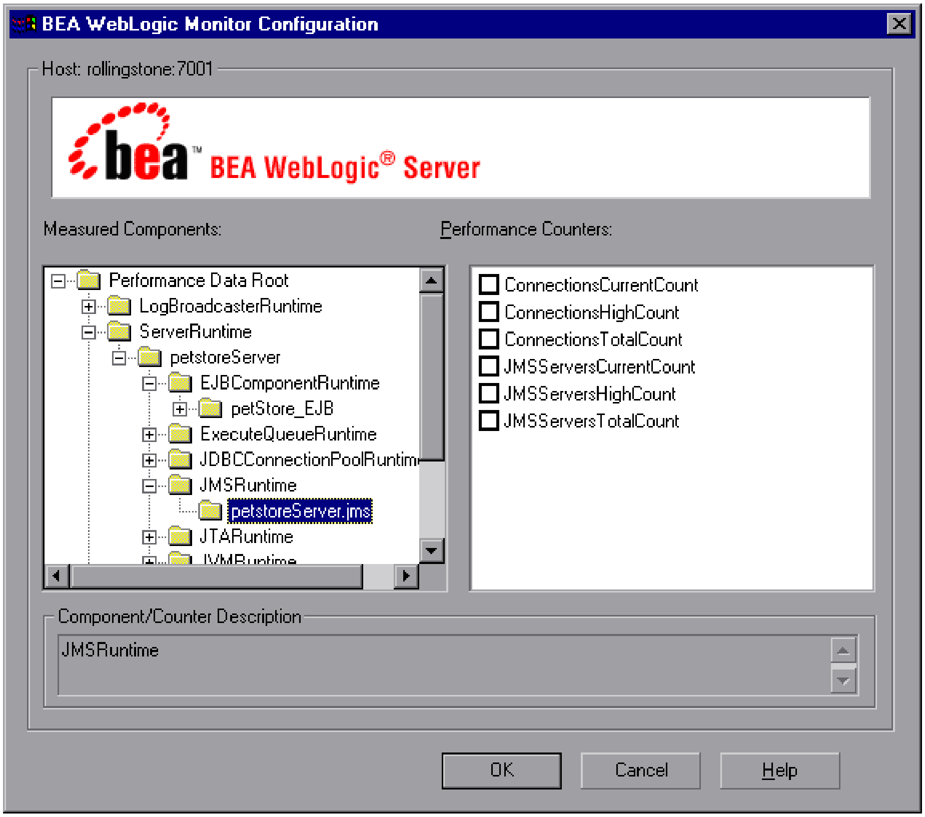
To configure the WebLogic (JMX) Monitor:

1. Click the WebLogic (JMX) graph in the graph tree, and drag it into the right pane of the Run view.
2. Right-click the graph and choose **Add Measurement(s)**, or choose **Monitors** > **Add Online Measurement**.
3. In the Monitored Server Machines section of the WebLogic (JMX) dialog box, click **Add** to enter the server name or IP address of the machine you want to monitor. Enter the server name or IP address according to the following format: *<server name>*:*<port number>*.

   For example: mercury:8111

   Select the platform on which the machine runs, and click **OK**.
4. Click **Add** in the Resource Measurements section of the WebLogic (JMX) dialog box. In the Enter Login Information dialog box, enter the username and password of a user with administrative privileges to the WebLogic server. The BEA WebLogic Monitor Configuration dialog box opens. For details on creating user permissions, see "Setting Permissions for Monitoring" on page 313.

5 Browse the Measured Components tree.



BEA WebLogic Monitor Configuration
bea™ BEA WebLogic® Server
Measured Components:
Performance Counters:
Component/Counter Description
OK
Cancel
Help

Host: rollingstone:7001

- Performance Data Root
  - LogBroadcasterRuntime
  - ServerRuntime
    - petstoreServer
      - EJBComponentRuntime
        - petStore_EJB
      - ExecuteQueueRuntime
      - JDBCConnectionPoolRuntim
      - JMSRuntime
        - petstoreServer.jms
      - JTARuntime
      - JVMRuntime

□ ConnectionsCurrentCount
□ ConnectionsHighCount
□ ConnectionsTotalCount
□ JMSServersCurrentCount
□ JMSServersHighCount
□ JMSServersTotalCount

JMSRuntime

6 Check the required performance counters in the BEA WebLogic Monitor Configuration window's right pane.

7 Click **OK** in the BEA WebLogic Monitor Configuration dialog box, and in the WebLogic (JMX) dialog box, to activate the WebLogic (JMX) monitor.

The following measurements are available for the WebLogic (JMX) server:

LogBroadcasterRuntime

Measurement	Description
MessagesLogged	The number of total log messages generated by this instance of the WebLogic server.
Registered	Returns "false" if the MBean represented by this object has been unregistered.
CachingDisabled	Private property that disables caching in proxies.

ServerRuntime

For more information on the measurements contained in each of the following measurement categories, see Mercury Interactive's Load Testing Monitors Web site (http://www-svca.mercuryinteractive.com/resources/library/technical/loadtesting_monitors/supported.html).

- ServletRuntime
- WebAppComponentRuntime
- EJBStatefulHomeRuntime
- JTARuntime
- JVMRuntime
- EJBEntityHomeRuntime.
- DomainRuntime
- EJBComponentRuntime
- DomainLogHandlerRuntime
- JDBCConnectionPoolRuntime
- ExecuteQueueRuntime
- ClusterRuntime
- JMSRuntime
- TimeServiceRuntime

- EJBStatelessHomeRuntime
- WLECConnectionServiceRuntime

ServerSecurityRuntime

Measurement	Description
UnlockedUsersTotalCount	Returns the number of times a user has been unlocked on the server
InvalidLoginUsersHighCount	Returns the high-water number of users with outstanding invalid login attempts for the server
LoginAttemptsWhileLockedTotalCount	Returns the cumulative number of invalid logins attempted on the server while the user was locked
Registered	Returns “false” if the MBean represented by this object has been unregistered.
LockedUsersCurrentCount	Returns the number of currently locked users on the server
CachingDisabled	Private property that disables caching in proxies.
InvalidLoginAttemptsTotalCount	Returns the cumulative number of invalid logins attempted on the server
UserLockoutTotalCount	Returns the cumulative number of user lockouts done on the server

Configuring the WebSphere Monitor

The WebSphere 3.x application server and the WebSphere 4.0 application server have different monitor installation requirements.

To monitor versions 3.02, 3.5, and 3.5.x of the IBM WebSphere application server, you must first install the appropriate IBM WebSphere servlet patch on the WebSphere machine.

Note: WebSphere version 4.0 contains the performance servlet within its default installation and therefore there is no need for a patch.

To install the IBM WebSphere servlet patch for the WebSphere 3.x server:

1 Open Mercury Interactive's Customer Support site, and select **Downloads** > **Patches** from the tree on the left.

Please make sure to install the appropriate patch according to your WebSphere version:

WebSphere version 3.02	IBM_WebSphere3.02_Servlet.zip
WebSphere version 3.5	IBM_WebSphere3.5_Servlet.zip
WebSphere version 3.5.x	IBM_WebSphere3.5.x_Servlet.zip

2 Unzip the *IBM_WebSphere<version#>_Servlet.zip* file in the ProTune Performance Monitors section.

3 Copy xml4j.jar, performance.dtd and perf.jar (version 3.02), or perf35.jar (version 3.5) or perf35x.jar (version 3.5.2 and 3.5.3) into the *default_host\default_app servlets* directory on the monitored machine.

To find the *default_app servlets* directory of the Web application, check the Web application's classpath. From the admin console, select the Web application in the tree and click on the Advanced tab. You should see the classpath for the Web application.

For example:

- **Microsoft Windows Platforms:** if the IBM WebSphere directory is installed under drive E, then the files should be copied to: *E:\WebSphere\AppServer\hosts\default_host\default_app\servlets*
- **IBM iSeries Platforms:** files should be copied to: */QIBM/UserDatat/WebASAdv/<instance>/hosts/default_host/default_app/servlets*
- **UNIX/Linux Platforms:** files should be copied to: */opt/IBMWebAS/hosts/default_host/default_app/servlets*

Note: If you want to monitor additional Web applications that are not on the same machine, copy the above files to the Servlets folder of the application you want to monitor.
Add the com.ibm.ivb.epm.servlet.PerformanceServlet to the classpath configuration in the WebSphere console for each Web application.

---

4 After copying the files, verify that the servlet is running properly and that the performance data is being generated. A simple way to verify that the performance data is accessible is to display it in a Web browser. The URL must be in the following format:

http://<*server name*:*port number*>/<*servlet_folder*>/com.ibm.ivb.epm.servlet.PerformanceServlet

For example: http://websphere.mercury.co.il:81/servlet/com.ibm.ivb.epm.servlet.PerformanceServlet

---

Note: Only browsers that are XML-compatible will allow you to view the performance XML file.

---

5 Open the *<ProTune root folder>\dat\monitors\xmlmonitorshared.ini* file and add the following line to the [WebSphereMonitor] section:

```
QueryLoginInfo=1
```

To set up the environment to monitor the WebSphere 4.0 server:

1. Open the < *root folder*>*dat**monitors**xmlmonitorshared.ini* file.

2. Add the following lines to the [WebSphereMonitor] section:

   ```
   ServletName=com.ibm.ws.pmi.perfServlet.PerformanceServlet

   ServletAlias=wasPerfTool/servlet

   QueryLoginInfo=1
   ```

Note: After making this change to the *xmlmonitorshared.ini* file, the Console will only be able to monitor WebSphere version 4.0. Previous versions of WebSphere will not be supported.

To configure the WebSphere monitor:

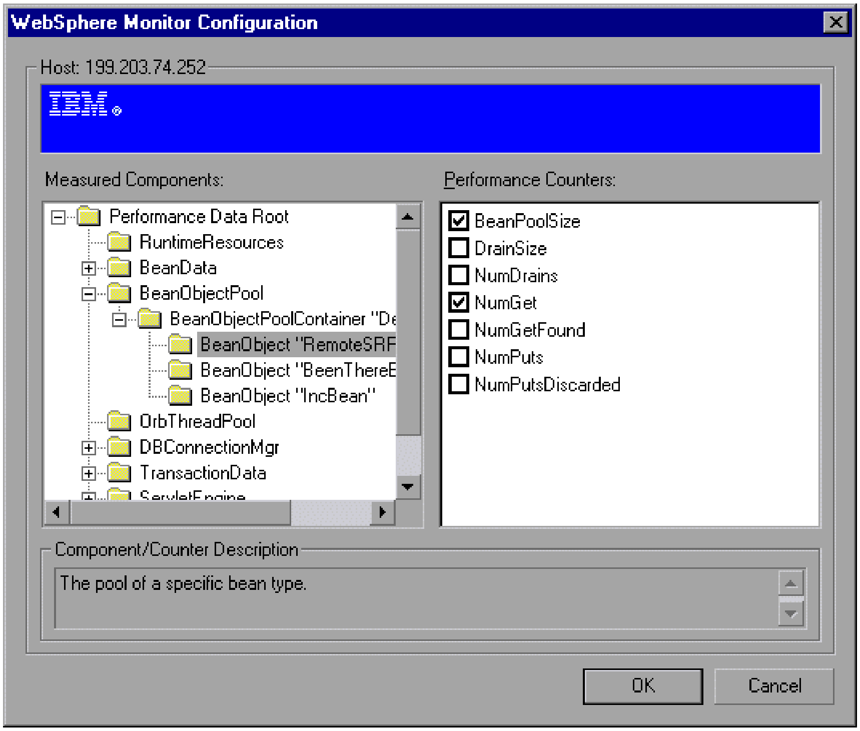
1. Click the WebSphere graph in the graph tree, and drag it into the right pane of the Run view.

2. Right-click the graph and choose **Add Measurement(s)**, or choose **Monitors** > **Add Online Measurement**.

3. In the Monitored Server Machines section of the WebSphere dialog box, click **Add** to enter the server name or IP address of the machine you want to monitor. Select the platform on which the machine runs, and click **OK**.

4. In the Resource Measurements section of the WebSphere dialog box, click **Add** to select the measurements that you want to monitor. The WebSphere Monitor Configuration dialog box displays the available measurements.

5 Browse the Measured Components tree.



WebSphere Monitor Configuration

Host: 199.203.74.252

Measured Components:

- Performance Data Root
  - RuntimeResources
  - BeanData
  - BeanObjectPool
    - BeanObjectPoolContainer "De
      - BeanObject "RemoteSRF
      - BeanObject "BeenThereE
      - BeanObject "IncBean"
  - OrbThreadPool
  - DBConnectionMgr
  - TransactionData
  - ServletEngine

Performance Counters:

☑ BeanPoolSize
☐ DrainSize
☐ NumDrains
☑ NumGet
☐ NumGetFound
☐ NumPuts
☐ NumPutsDiscarded

Component/Counter Description

The pool of a specific bean type.

6 Check the required performance counters in the WebSphere Monitor Configuration window's right pane. For a list of the available performance counters, see page 323.

7 Click **OK** in the WebSphere Monitor Configuration dialog box, and in the WebSphere dialog box, to activate the WebSphere monitor.

Note: The port you use to monitor a WebSphere server through a firewall depends on the configuration of your server.

To specify another Web alias for the servlet directory:

By default, ProTune uses the alias servlet as the servlet directory Web alias. For example, if the WebSphere Server machine is named mercury and the path for the servlets directory is: E:\AppServer\hosts\default_host\default_app\servlets, ProTune will request the XML file in the following URL: http:/mercury/servlet/com.ibm.ivb.epm.servlet.PerformanceServlet, where servlet is the default web alias for the servlet directory.

If the Web alias for the servlet directory is not servlet, you must specify the servlet directory Web alias in the Add Machine dialog box according to the following format:

http://<*server name*:*port number*>/<*servlet_dir_alias*>

For example: http://mercury/servlet2

Using this method, you can monitor as many application servers as you want—whether they are installed on the same machine, or on different machines.

To monitor other applications, in addition to the default application:

You can monitor as many applications as you want, regardless of whether they are installed on the same machine or different machines.

1 Copy the same files that you copied to the Servlets directory for the Default application to the Servlets directories for any other Web applications that you want to monitor.

2 Add the com.ibm.ivb.epm.servlet.PerformanceServlet to the configuration in the WebSphere Console for each Web application.

3 Add the Web application to be monitored to the WebSphere Performance Monitor using the following format:

   http://<*server:port_number*>/<*servlet_dir_alias*>/*servlet*

   For example: http://mercury/servlet3/servlet

To work with WebSphere version 3.5.x

1 The EPM counters in 3.5.x are by default set to "none". To enable the counters, choose the application server you are monitoring in the WebSphere Administrator's Console browser.

2 Right-click the application server and select **Performance**. Select Performance Modules from the pop-up window.

3 Right-click Performance Modules to choose a performance level. Selecting various levels of counters enables the application server to manage varying levels of performance data.

4 Click the **Set** button.

5 In versions 3.5.2 and 3.5.3 the Servlet counters have been disabled. To enable the Servlet counters, you need to modify the contents of the com/ibm/servlet/appserver.properties file located in "<WAS_HOME>\lib\ibmwebas.jar".

Extract the *jar* file and modify the appserver.properties as follows:

```
#listeners.application=com.ibm.servlet.engine.EPMApplicationListener
com.ibm.servlet.debug.OLTServletManager
listeners.application=
```

Should be:

```
listeners.application=com.ibm.servlet.engine.EPMApplicationListener
com.ibm.servlet.debug.OLTServletManager
#listeners.application=
```

6 Repackage the *jar* file.

WebSphere Counters

The following tables describe the counters that can be monitored:

➤ **Run-Time Resources**

Contains resources related to the Java Virtual Machine run-time, as well as the ORB.

Measurement	Description
MemoryFree	The amount of free memory remaining in the Java Virtual Machine
MemoryTotal	The total memory allocated for the Java Virtual Machine
MemoryUse	The total memory in use within the Java Virtual Machine

➤ BeanData

Every home on the server provides performance data, depending upon the type of bean deployed in the home. The top level bean data holds an aggregate of all the containers.

Measurement	Description
BeanCreates	The number of beans created. Applies to an individual bean that is either 'stateful' or 'entity'
EntityBeanCreates	The number of entity beans created
BeanRemoves	The number of entity beans pertaining to a specific bean that have been removed. Applies to an individual bean that is either 'stateful' or 'entity'
EntityBeanRemoves	The number of entity beans removed
StatefulBeanCreates	The number of stateful beans created
StatefulBeanRemoves	The number of stateful bean removed
BeanPassivates	The number of bean passivates pertaining to a specific bean. Applies to an individual bean that is either 'stateful' or 'entity'
EntityBeanPassivates	The number of entity bean passivates
StatefulBeanPassivates	The number of stateful bean passivates
BeanActivates	The number of bean activates pertaining to a specific bean. Applies to an individual bean that is either 'stateful' or 'entity'
EntityBeanActivates	The number of entity bean activates
StatefulBeanActivates	The number of stateful bean activates
BeanLoads	The number of times the bean data was loaded. Applies to entity
BeanStores	The number of times the bean data was stored in the database. Applies to entity

Measurement	Description
BeanInstantiates	The number of times a bean object was created. This applies to an individual bean, regardless of its type.
StatelessBeanInstantiates	The number of times a stateless session bean object was created
StatefulBeanInstantiates	The number of times a stateful session bean object was created
EntityBeanInstantiates	The number of times an entity bean object was created
BeanDestroys	The number of times an individual bean object was destroyed. This applies to any bean, regardless of its type
StatelessBeanDestroys	The number of times a stateless session bean object was destroyed
StatefulBeanDestroys	The number of times a stateful session bean object was destroyed
EntityBeanDestroys	The number of times an entity bean object was destroyed
BeansActive	The average number of instances of active beans pertaining to a specific bean. Applies to an individual bean that is either 'stateful' or 'entity'
EntityBeansActive	The average number of active entity beans
StatefulBeansActive	The average number of active session beans
BeansLive	The average number of bean objects of this specific type that are instantiated but not yet destroyed. This applies to an individual bean, regardless of its type.
StatelessBeansLive	The average number of stateless session bean objects that are instantiated but not yet destroyed
StatefulBeansLive	The average number of stateful session bean objects that are instantiated but not yet destroyed

Measurement	Description
EntityBeansLive	The average number of entity bean objects that are instantiated but not yet destroyed
BeanMethodRT	The average method response time for all methods defined in the remote interface to this bean. Applies to all beans
BeanMethodActive	The average number of methods being processed concurrently. Applies to all beans
BeanMethodCalls	The total number of method calls against this bean

➤ BeanObjectPool

The server holds a cache of bean objects. Each home has a cache and there is therefore one BeanObjectPoolContainer per container. The top level BeanObjectPool holds an aggregate of all the containers data.

Measurement	Description
BeanObjectPoolContainer	The pool of a specific bean type
BeanObject	The pool specific to a home
NumGet	The number of calls retrieving an object from the pool
NumGetFound	The number of calls to the pool that resulted in finding an available bean
NumPuts	The number of beans that were released to the pool
NumPutsDiscarded	The number of times releasing a bean to the pool resulted in the bean being discarded because the pool was full
NumDrains	The number of times the daemon found the pool was idle and attempted to clean it

Measurement	Description
DrainSize	The average number of beans discarded by the daemon during a clean
BeanPoolSize	The average number of beans in the pool

➤ OrbThreadPool

These are resources related to the ORB thread pool that is on the server.

Measurement	Description
ActiveThreads	The average number of active threads in the pool
TotalThreads	The average number of threads in the pool
PercentTimeMaxed	The average percent of the time that the number of threads in the pool reached or exceeded the desired maximum number
ThreadCreates	The number of threads created
ThreadDestroys	The number of threads destroyed
ConfiguredMaxSize	The configured maximum number of pooled threads

➤ DBConnectionMgr

These are resources related to the database connection manager. The manager consists of a series of data sources, as well as a top-level aggregate of each of the performance metrics.

Measurement	Description
DataSource	Resources related to a specific data source specified by the "name" attribute
ConnectionCreates	The number of connections created
ConnectionDestroys	The number of connections released
ConnectionPoolSize	The average size of the pool, i.e., number of connections
ConnectionAllocates	The number of times a connection was allocated
ConnectionWaiters	The average number of threads waiting for a connection
ConnectionWaitTime	The average time, in seconds, of a connection grant
ConnectionTime	The average time, in seconds, that a connection is in use
ConnectionPercentUsed	The average percentage of the pool that is in use
ConnectionPercentMaxed	The percentage of the time that all connections are in use

➤ TransactionData

These are resources that pertain to transactions.

Measurement	Description
NumTransactions	The number of transactions processed
ActiveTransactions	The average number of active transactions
TransactionRT	The average duration of each transaction

Measurement	Description
BeanObjectCount	The average number of bean object pools involved in a transaction
RolledBack	The number of transactions rolled back
Commited	The number of transactions committed
LocalTransactions	The number of transactions that were local
TransactionMethodCount	The average number of methods invoked as part of each transaction
Timeouts	The number of transactions that timed out due to inactivity timeouts
TransactionSuspended	The average number of times that a transaction was suspended

➤ ServletEngine

These are resources that are related to servlets and JSPs.

Measurement	Description
ServletsLoaded	The number of servlets currently loaded
ServletRequests	The number of requests serviced
CurrentRequests	The number of requests currently being serviced
ServletRT	The average response time for each request
ServletsActive	The average number of servlets actively processing requests
ServletIdle	The amount of time that the server has been idle (i.e., time since last request)
ServletErrors	The number of requests that resulted in an error or an exception
ServletBeanCalls	The number of bean method invocations that were made by the servlet

Measurement	Description
ServletBeanCreates	The number of bean references that were made by the servlet
ServletDBCalls	The number of database calls made by the servlet
ServletDBConAlloc	The number of database connections allocated by the servlet
SessionLoads	The number of times the servlet session data was read from the database
SessionStores	The number of times the servlet session data was stored in the database
SessionSize	The average size, in bytes, of a session data
LoadedSince	The time that has passed since the server was loaded (UNC time)

➤ Sessions

These are general metrics regarding the HTTP session pool.

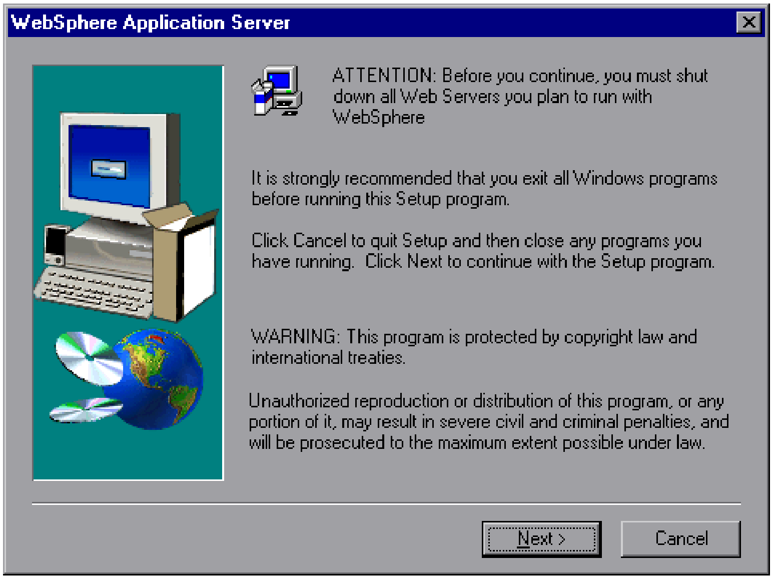
Measurement	Description
SessionsCreated	The number of sessions created on the server
SessionsActive	The number of currently active sessions
SessionsInvalidated	The number of invalidated sessions. May not be valid when using sessions in the database mode
SessionLifetime	Contains statistical data of sessions that have been invalidated. Does not include sessions that are still alive

Configuring the WebSphere (EPM) Monitor

To monitor the IBM WebSphere application server (3.5.x), you must first install the IBM WebSphere Administrator's Console on the Console machine. You may also need to copy the security keyring.

To install the IBM WebSphere Administrator's Console:

1 Start the WebSphere installation program from the WebSphere 3.5 Windows NT distribution CD-ROM. The WebSphere Application Server dialog box opens.



WebSphere Application Server

ATTENTION: Before you continue, you must shut down all Web Servers you plan to run with WebSphere

It is strongly recommended that you exit all Windows programs before running this Setup program.

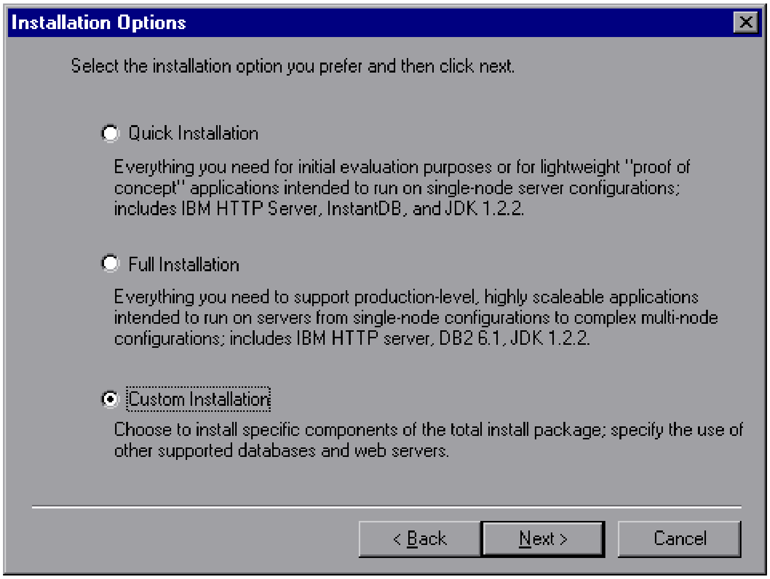
Click Cancel to quit Setup and then close any programs you have running. Click Next to continue with the Setup program.

WARNING: This program is protected by copyright law and international treaties.

Unauthorized reproduction or distribution of this program, or any portion of it, may result in severe civil and criminal penalties, and will be prosecuted to the maximum extent possible under law.

2 Disregard the instruction to shut down all Web servers that you plan to run with WebSphere. This is not relevant to the Administrator's Console installation. Follow the remaining instructions.

3 Click **Next** to proceed. The Installation Options dialog box opens.



Installation Options

Select the installation option you prefer and then click next.

○ Quick Installation

Everything you need for initial evaluation purposes or for lightweight "proof of concept" applications intended to run on single-node server configurations; includes IBM HTTP Server, InstantDB, and JDK 1.2.2.

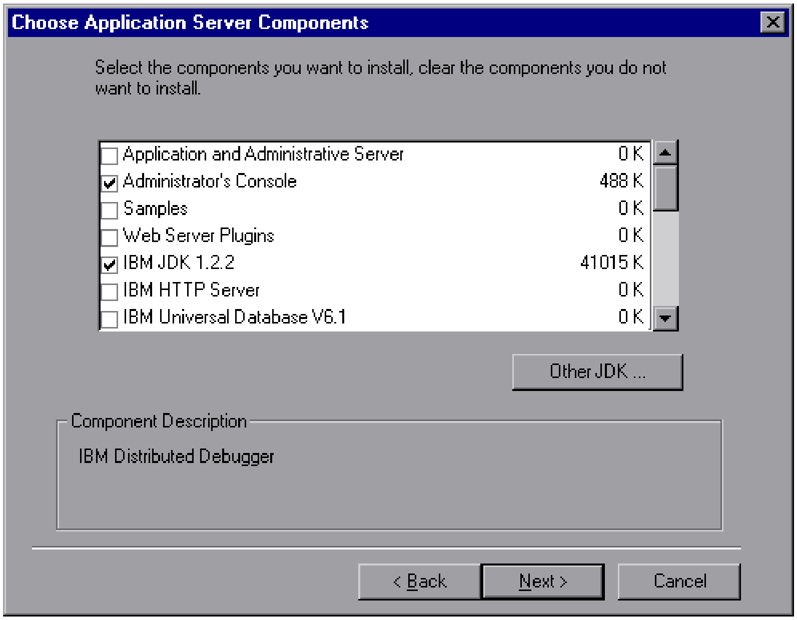
○ Full Installation

Everything you need to support production-level, highly scaleable applications intended to run on servers from single-node configurations to complex multi-node configurations; includes IBM HTTP server, DB2 6.1, JDK 1.2.2.

● Custom Installation

Choose to install specific components of the total install package; specify the use of other supported databases and web servers.

4 Select **Custom Installation**, and click **Next**. The Choose Application Server Components dialog box opens.



Choose Application Server Components

Select the components you want to install, clear the components you do not want to install.

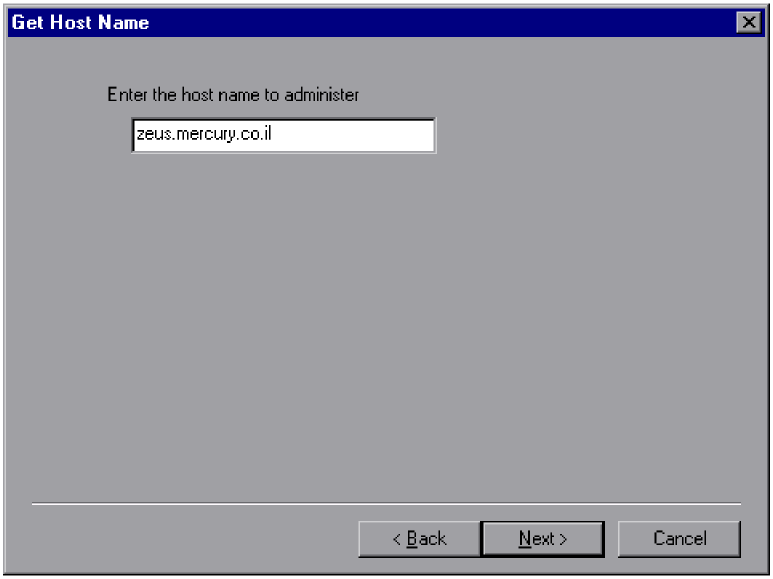
□	Application and Administrative Server	0 K
☑	Administrator's Console	488 K
□	Samples	0 K
□	Web Server Plugins	0 K
☑	IBM JDK 1.2.2	41015 K
□	IBM HTTP Server	0 K
□	IBM Universal Database V6.1	0 K

Component Description

IBM Distributed Debugger

5 Select **Administrator's Console** and **IBM JDK 1.2.2**. Clear all the other options.

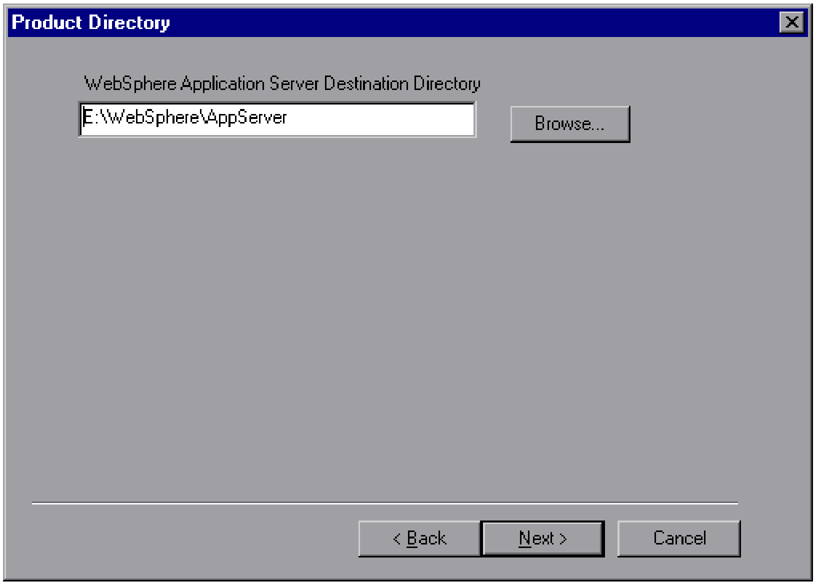
6 Click **Next**. The Get Host Name dialog box opens.



zeus.mercury.co.il

7 Type the name of the machine that you want to monitor.

8 Click **Next**. The Product Directory dialog box opens.

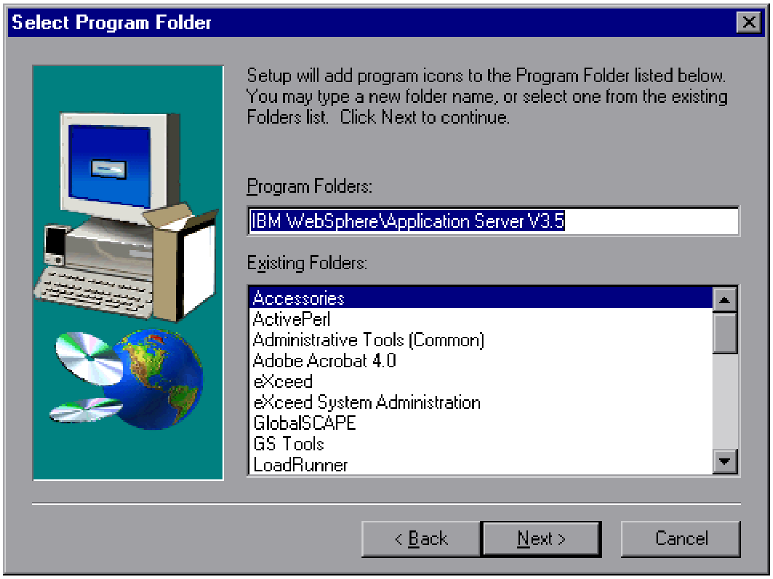


WebSphere Application Server Destination Directory

E:\WebSphere\AppServer

9 Specify the folder in which to install the Administrator's Console. To select a different location, click **Browse**, choose a folder other than the default folder, and click **OK**.

10 Click **Next**. The Select Program Folder dialog box opens.



Select Program Folder

Setup will add program icons to the Program Folder listed below. You may type a new folder name, or select one from the existing Folders list. Click Next to continue.

Program Folders:

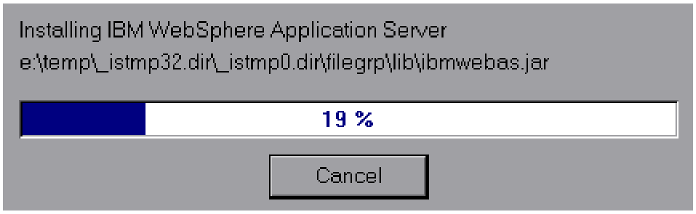
IBM WebSphere\Application Server V3.5

Existing Folders:

Accessories
ActivePerl
Administrative Tools (Common)
Adobe Acrobat 4.0
eXceed
eXceed System Administration
GlobalSCAPE
GS Tools
LoadRunner

11 Specify a program folder, or accept the default folder, IBM WebSphere\Application Server V3.5.

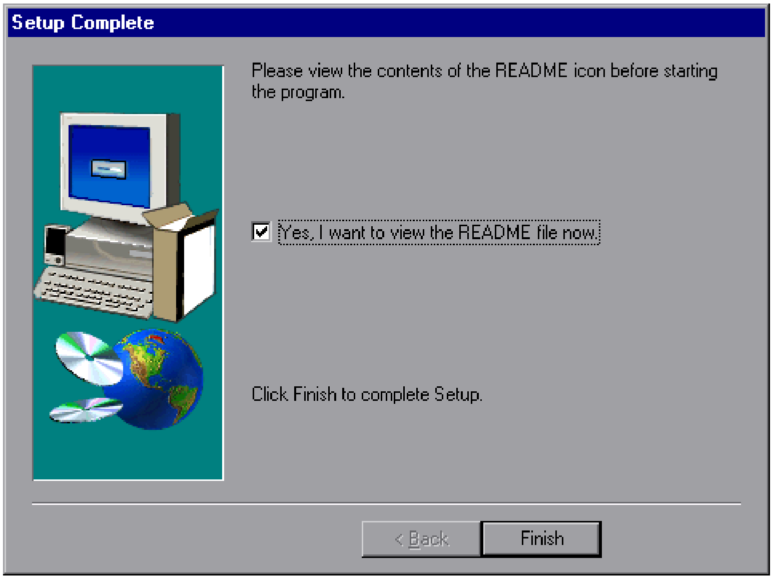
12 Click **Next**. The installation process begins. To pause or quit the installation, click **Cancel**.



Installing IBM WebSphere Application Server
e:\temp_istmp32.dir_istmp0.dir\filegrp\lib\ibmwebas.jar

19 %

When the installation is complete, the Setup Complete dialog box opens.



Setup Complete

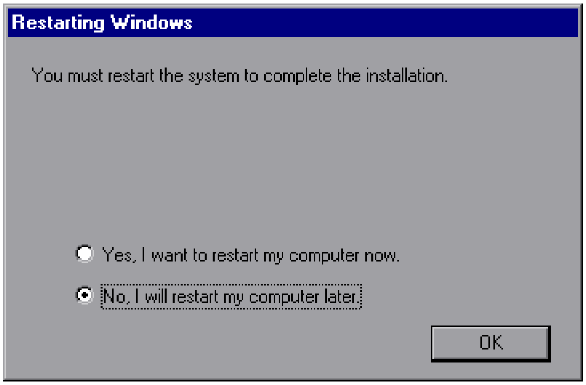
Please view the contents of the README icon before starting the program.

☑ Yes, I want to view the README file now.

Click Finish to complete Setup.

13 In the Setup Complete dialog box, select the check box to view the readme file before starting the program. You can view the readme file at any time by selecting **Start** > **Programs** > **Application Server V3.5** > **IBM WebSphere** > **README**.

14 Click **Finish** to complete the installation program. The Restarting Windows dialog box opens.



Restarting Windows

You must restart the system to complete the installation.

○ Yes, I want to restart my computer now.

● No, I will restart my computer later.

15 Select either to restart your computer and complete the installation now (recommended) or to wait and complete the installation later.

16 Click **OK** to complete the installation of the Administrator's Console.

Copying the Security Keyring

If you enabled security on the WebSphere server, you must copy the security keyring from the server to the admin client. (One way to tell whether security is enabled is to see whether the Administrator's Console can connect to the admin server.) A keyring is a certification used by the server to identify the client.

You need to copy the *jar* file containing the keyring from the server lib folder to the client lib folder. You also need to add the *jar* file containing the keyring to the monitoring client command line.

Note: The keyring used in this file (*353Keyring.jar*) is the IBM dummy keyring that must be installed on servers using versions 3.52 and below. If your server is using the IBM dummy keyring and is version 3.52 or below, you do not need to change the line. If you are using the dummy keyring and are running version 3.53 or later, you do not need to do anything.

To copy the keyring:

1 Copy the keyring *jar* file from the server to the admin client lib folder (by default, C:\Websphere\Appserver\lib):

The *jar* file containing the keyring, *xxxKeyring.jar*, is located by default in the following location:

NT Server	C:\Websphere\Appserver\lib
UNIX Server	OPT/websphere/Appserver/lib

2. Open the *<ProTune root folder>\dat\monitors\WebSphere35Mon.ini* file in a text editor.
3. Locate the following line:
   ```
   JVM_CLASSES4=C:\WebSphere\AppServer\lib\353Keyring.jar
   ```

   Note: If you did not use the default location for the WebSphere installation, the line will be different.

4. Change *353Keyring.jar* to the keyring you are using.

Enabling EPM Counters on the WebSphere 3.5.x Server

To enable the EPM counters, which are by default set to "none," right-click the application you are monitoring in the WebSphere Administrator's Console browser, and select **Performance**. Expand the Performance Modules tree in the dialog box that opens. In order to manage different levels of performance data, right-click the performance modules and choose a performance level. Click the **Set** button.

Alternatively, ensure that the application server is started, select the **Advanced** tab in the WebSphere Administrator's Console browser, and in the EPM Specification box, type:

```
epm=high:epm.beanMethodData=none
```

Activating the WebSphere (EPM) Monitor

Once you have installed the WebSphere Administrator's Console and enabled the EPM counters, you can activate the WebSphere (EPM) monitor.

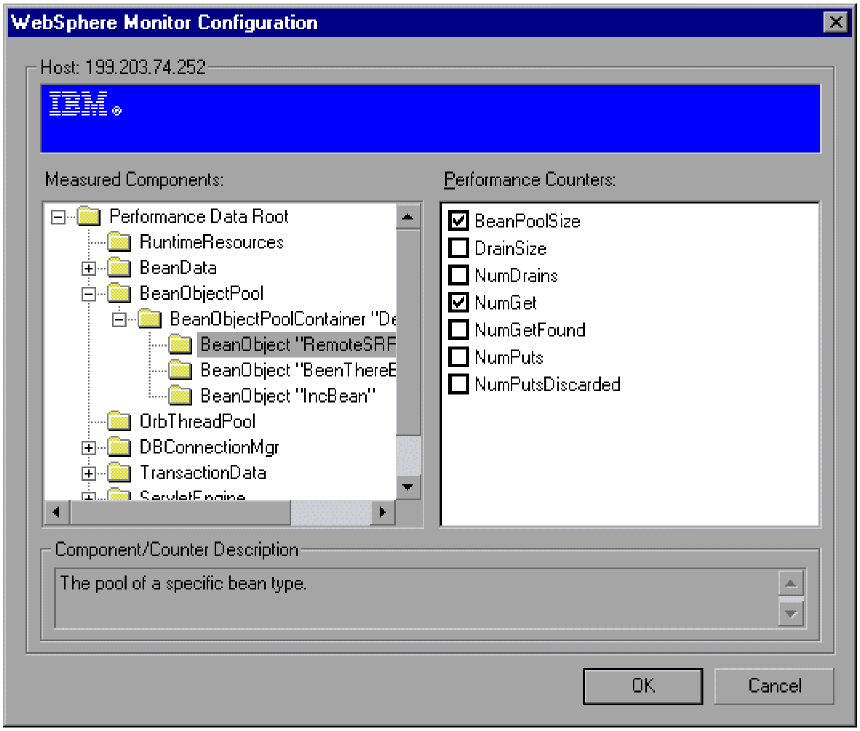
To activate the WebSphere EPM monitor:

1. Click the WebSphere (EPM) graph in the graph tree, and drag it into the right pane of the Run view.
2. Right-click the graph and choose **Add Measurement(s)**, or choose **Monitors** > **Add Online Measurement**.

3 In the Monitored Server Machines section of the WebSphere (EPM) dialog box, click **Add** to enter the server name or IP address of the machine you want to monitor. Select the platform on which the machine runs, and click **OK**.

4 In the Resource Measurements section of the WebSphere (EPM) dialog box, click **Add** to select the measurements that you want to monitor. The WebSphere Monitor Configuration dialog box displays the available measurements.

5 Browse the Measured Components tree.



WebSphere Monitor Configuration

Host: 199.203.74.252

Measured Components:

```
Performance Data Root
  RuntimeResources
  BeanData
  BeanObjectPool
    BeanObjectPoolContainer "De
      BeanObject "RemoteSRF
      BeanObject "BeenThereE
      BeanObject "IncBean"
  OrbThreadPool
  DBConnectionMgr
  TransactionData
  ServletEngine
```

Performance Counters:

☑ BeanPoolSize
☐ DrainSize
☐ NumDrains
☑ NumGet
☐ NumGetFound
☐ NumPuts
☐ NumPutsDiscarded

Component/Counter Description

The pool of a specific bean type.

6 Check the required performance counters in the WebSphere Monitor Configuration window's right pane. For a list of the available performance counters, see page 323.

7 Click **OK** in the WebSphere Monitor Configuration dialog box, and in the WebSphere (EPM) dialog box, to activate the WebSphere (EPM) monitor.

22

Database Resource Monitoring

You can monitor DB2, Oracle, SQL Server, or Sybase database resource usage during a session step run using ProTune's Database Server Resource monitors.

This chapter describes:

- Configuring the DB2 Monitor
- Configuring the Oracle Monitor
- Configuring the SQL Server Monitor
- Configuring the Sybase Monitor

About Database Resource Monitoring

The DB2, Oracle, SQL Server, or Sybase database server resource monitors measure statistics for DB2, Oracle, SQL Server, or Sybase database servers. During a session step run, you use these monitors to isolate database server performance bottlenecks.

For each database server, you configure the measurements you want to monitor before running your session step. Note that in order to run the DB2, Oracle, and Sybase monitors, you must also install the client libraries on the database server you want to monitor.

Configuring the DB2 Monitor

The DB2 database server monitor measures the resource usage on a DB2 database during a session step run.

Note: If there is no application working with a database, you can only monitor the database manager instance.

Before you can monitor a DB2 database server, you must set up the DB2 monitor environment.

To set up the DB2 monitor environment:

1. Install all the client files and libraries on the Console machine. It is recommended that you install DB2 Connect Enterprise Edition 7.2, if you are working on a 390 operating system.
2. Select **Start** > **Programs** > **DB2 for Windows NT** > **Control Center**. Enter your DB2 server username and password (with administrative privileges).
3. In the console that opens, right-click **Systems**, and select **Add**.
4. Enter the following settings in the dialog box:

   System Name: *<server name>*

   Remote Instance: DB2

   Host Name: *<server name>*

   Service Name: the DB2 server port. The default value is 50000.

5. Click **Retrieve**, and then **OK**.

Note: If you receive an error message after clicking **Retrieve**, repeat steps 3 and 4, and click **OK**.

---

6. Expand the *<server name>* node in the console tree.

7 Right-click **Instance**, and select **Add**.

8 Enter the following settings in the dialog box:

Remote Instance: DB2

Instance Name: the database instance to be called from the Console

Host Name: <*server name*>

Service Name: the DB2 server port. The default value is 50000.

9 Click **OK** and close the Control Center.

Note: You can only work with a single Database Manager instance during each monitoring session.

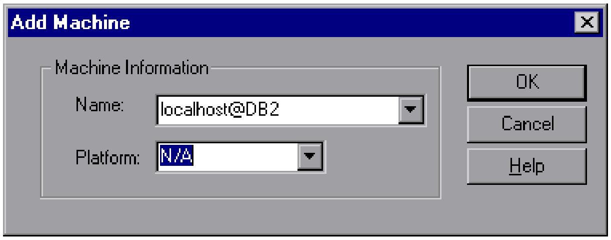
To configure the DB2 monitor:

1 Click the DB2 graph in the graph tree, and drag it into the right pane of the Run view.

2 Right-click the graph and choose **Add Measurement(s)**, or choose **Monitors** > **Add Online Measurement**. The DB2 dialog box opens.

3 Click the **Add** button in the Monitored Server Machines section of the dialog box. The Add Machine dialog box opens.

4 In the Name box, enter the DB2 server machine name followed by the @ sign and the database instance you specified in the DB2 Control Center. In the Platform box, select **N/A**.


Add Machine

Machine Information

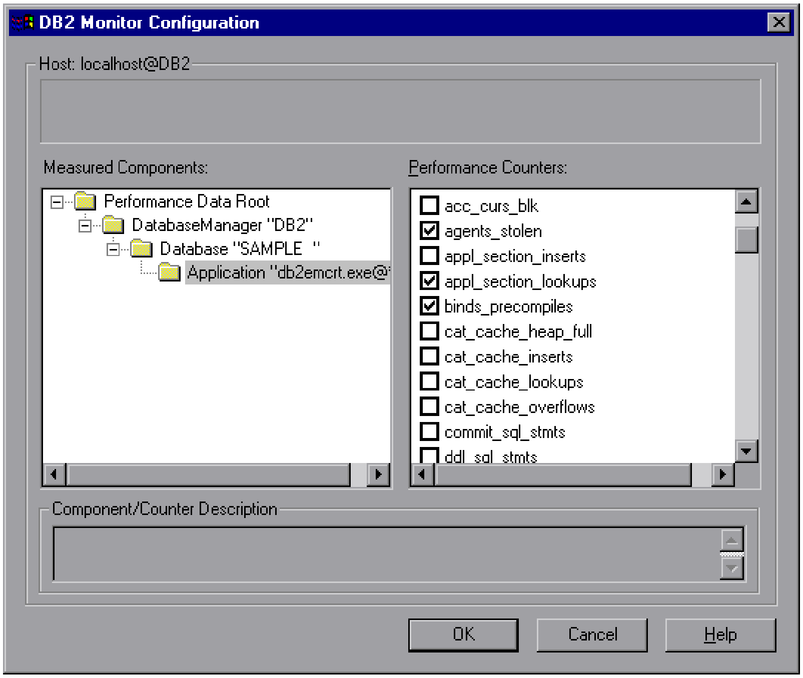
Name: localhost@DB2

Platform: N/A

Click **OK** to save the information you entered and close the dialog box.

5 Click **Add** in the Resource Measurements section of the DB2 dialog box. In the dialog box that opens, enter your DB2 server username and password, and click **OK**. The DB2 Monitor Configuration dialog box opens.

6 Expand the Measured Components tree and select the methods and counters you want to monitor.



Host: localhost@DB2

Measured Components:

- Performance Data Root
  - DatabaseManager "DB2"
    - Database "SAMPLE "
      - Application "db2emcrt.exe@

Performance Counters:

□ acc_curs_blk
☑ agents_stolen
□ appl_section_inserts
☑ appl_section_lookups
☑ binds_precompiles
□ cat_cache_heap_full
□ cat_cache_inserts
□ cat_cache_lookups
□ cat_cache_overflows
□ commit_sql_stmts
□ ddl_sql_stmts

The following tables describe the default counters that can be monitored.

DatabaseManager

Measurement	Description
rem_cons_in	The current number of connections initiated from remote clients to the instance of the database manager that is being monitored.
rem_cons_in_exec	The number of remote applications that are currently connected to a database and are currently processing a unit of work within the database manager instance being monitored.
local_cons	The number of local applications that are currently connected to a database within the database manager instance being monitored.
local_cons_in_exec	The number of local applications that are currently connected to a database within the database manager instance being monitored and are currently processing a unit of work.
con_local_dbases	The number of local databases that have applications connected.
agents_registered	The number of agents registered in the database manager instance that is being monitored (coordinator agents and subagents).
agents_waiting_on_token	The number of agents waiting for a token so they can execute a transaction in the database manager.
idle_agents	The number of agents in the agent pool that are currently unassigned to an application and are therefore "idle".
agents_from_pool	The number of agents assigned from the agent pool
agents_created_empty_pool	The number of agents created because the agent pool was empty.

Measurement	Description
agents_stolen	The number of times that agents are stolen from an application. Agents are stolen when an idle agent associated with an application is reassigned to work on a different application.
comm_private_mem	The amount of private memory that the instance of the database manager has currently committed at the time of the snapshot.
inactive_gw_agents	The number of DRDA agents in the DRDA connections pool that are primed with a connection to a DRDA database, but are inactive.
num_gw_conn_switches	The number of times that an agent from the agents pool was primed with a connection and was stolen for use with a different DRDA database.
sort_heap_allocated	The total number of allocated pages of sort heap space for all sorts at the level chosen and at the time the snapshot was taken.
post_threshold_sorts	The number of sorts that have requested heaps after the sort heap threshold has been reached.
piped_sorts_requested	The number of piped sorts that have been requested.
piped_sorts_accepted	The number of piped sorts that have been accepted.

Database

Measurement	Description
appls_cur_cons	Indicates the number of applications that are currently connected to the database.
appls_in_db2	Indicates the number of applications that are currently connected to the database, and for which the database manager is currently processing a request.

Measurement	Description
total_sec_cons	The number of connections made by a sub-agent to the database at the node.
num_assoc_agents	At the application level, this is the number of sub-agents associated with an application. At the database level, it is the number of sub-agents for all applications.
sort_heap_allocated	The total number of allocated pages of sort heap space for all sorts at the level chosen and at the time the snapshot was taken.
total_sorts	The total number of sorts that have been executed.
total_sort_time	The total elapsed time (in milliseconds) for all sorts that have been executed.
sort_overflows	The total number of sorts that ran out of sort heap and may have required disk space for temporary storage.
active_sorts	The number of sorts in the database that currently have a sort heap allocated.
total_hash_joins	The total number of hash joins executed.
total_hash_loops	The total number of times that a single partition of a hash join was larger than the available sort heap space.
hash_join_overflows	The number of times that hash join data exceeded the available sort heap space
hash_join_small_overflows	The number of times that hash join data exceeded the available sort heap space by less than 10%.
pool_data_l_reads	Indicates the number of logical read requests for data pages that have gone through the buffer pool.
pool_data_p_reads	The number of read requests that required I/O to get data pages into the buffer pool.

Measurement	Description
pool_data_writes	Indicates the number of times a buffer pool data page was physically written to disk.
pool_index_l_reads	Indicates the number of logical read requests for index pages that have gone through the buffer pool.
pool_index_p_reads	Indicates the number of physical read requests to get index pages into the buffer pool.
pool_index_writes	Indicates the number of times a buffer pool index page was physically written to disk.
pool_read_time	Provides the total amount of elapsed time spent processing read requests that caused data or index pages to be physically read from disk to buffer pool.
pool_write_time	Provides the total amount of time spent physically writing data or index pages from the buffer pool to disk.
files_closed	The total number of database files closed.
pool_async_data_reads	The number of pages read asynchronously into the buffer pool.
pool_async_data_writes	The number of times a buffer pool data page was physically written to disk by either an asynchronous page cleaner, or a pre-fetcher. A pre-fetcher may have written dirty pages to disk to make space for the pages being pre-fetched.
pool_async_index_writes	The number of times a buffer pool index page was physically written to disk by either an asynchronous page cleaner, or a pre-fetcher. A pre-fetcher may have written dirty pages to disk to make space for the pages being pre-fetched.
pool_async_index_reads	The number of index pages read asynchronously into the buffer pool by a pre-fetcher.
pool_async_read_time	The total elapsed time spent reading by database manager pre-fetchers.

Measurement	Description
pool_async_write_time	The total elapsed time spent writing data or index pages from the buffer pool to disk by database manager page cleaners.
pool_async_data_read_reqs	The number of asynchronous read requests.
pool_lsn_gap_clns	The number of times a page cleaner was invoked because the logging space used had reached a predefined criterion for the database.
pool_drty_pg_steal_clns	The number of times a page cleaner was invoked because a synchronous write was needed during the victim buffer replacement for the database.
pool_drty_pg_thrsh_clns	The number of times a page cleaner was invoked because a buffer pool had reached the dirty page threshold criterion for the database.
prefetch_wait_time	The time an application spent waiting for an I/O server (pre-fetcher) to finish loading pages into the buffer pool.
pool_data_to_estore	The number of buffer pool data pages copied to extended storage.
pool_index_to_estore	The number of buffer pool index pages copied to extended storage.
pool_data_from_estore	The number of buffer pool data pages copied from extended storage.
pool_index_from_estore	The number of buffer pool index pages copied from extended storage.
direct_reads	The number of read operations that do not use the buffer pool.
direct_writes	The number of write operations that do not use the buffer pool.
direct_read_reqs	The number of requests to perform a direct read of one or more sectors of data.
direct_write_reqs	The number of requests to perform a direct write of one or more sectors of data.

Measurement	Description
direct_read_time	The elapsed time (in milliseconds) required to perform the direct reads.
direct_write_time	The elapsed time (in milliseconds) required to perform the direct writes.
cat_cache_lookups	The number of times that the catalog cache was referenced to obtain table descriptor information.
cat_cache_inserts	The number of times that the system tried to insert table descriptor information into the catalog cache.
cat_cache_overflows	The number of times that an insert into the catalog cache failed due the catalog cache being full.
cat_cache_heap_full	The number of times that an insert into the catalog cache failed due to a heap-full condition in the database heap.
pkg_cache_lookups	The number of times that an application looked for a section or package in the package cache. At a database level, it indicates the overall number of references since the database was started, or monitor data was reset.
pkg_cache_inserts	The total number of times that a requested section was not available for use and had to be loaded into the package cache. This count includes any implicit prepares performed by the system.
pkg_cache_num_overflows	The number of times that the package cache overflowed the bounds of its allocated memory.
appl_section_lookups	Lookups of SQL sections by an application from its SQL work area.
appl_section_inserts	Inserts of SQL sections by an application from its SQL work area.
sec_logs_allocated	The total number of secondary log files that are currently being used for the database.

Measurement	Description
log_reads	The number of log pages read from disk by the logger.
log_writes	The number of log pages written to disk by the logger.
total_log_used	The total amount of active log space currently used (in bytes) in the database.
locks_held	The number of locks currently held.
lock_list_in_use	The total amount of lock list memory (in bytes) that is in use.
deadlocks	The total number of deadlocks that have occurred.
lock_escals	The number of times that locks have been escalated from several row locks to a table lock.
x_lock_escals	The number of times that locks have been escalated from several row locks to one exclusive table lock, or the number of times an exclusive lock on a row caused the table lock to become an exclusive lock.
lock_timeouts	The number of times that a request to lock an object timed-out instead of being granted.
lock_waits	The total number of times that applications or connections waited for locks.
lock_wait_time	The total elapsed time waited for a lock.
locks_waiting	Indicates the number of agents waiting on a lock.
rows_deleted	The number of row deletions attempted.
rows_inserted	The number of row insertions attempted.
rows_updated	The number of row updates attempted.
rows_selected	The number of rows that have been selected and returned to the application.

Measurement	Description
int_rows_deleted	The number of rows deleted from the database as a result of internal activity.
int_rows_updated	The number of rows updated from the database as a result of internal activity.
int_rows_inserted	The number of rows inserted into the database as a result of internal activity caused by triggers.
static_sql_stmts	The number of static SQL statements that were attempted.
dynamic_sql_stmts	The number of dynamic SQL statements that were attempted.
failed_sql_stmts	The number of SQL statements that were attempted, but failed.
commit_sql_stmts	The total number of SQL COMMIT statements that have been attempted.
rollback_sql_stmts	The total number of SQL ROLLBACK statements that have been attempted.
select_sql_stmts	The number of SQL SELECT statements that were executed.
uid_sql_stmts	The number of SQL UPDATE, INSERT, and DELETE statements that were executed.
ddl_sql_stmts	This element indicates the number of SQL Data Definition Language (DDL) statements that were executed.
int_auto_rebinds	The number of automatic rebinds (or recompiles) that have been attempted.
int_commits	The total number of commits initiated internally by the database manager.
int_rollbacks	The total number of rollbacks initiated internally by the database manager.

Measurement	Description
int_deadlock_rollbacks	The total number of forced rollbacks initiated by the database manager due to a deadlock. A rollback is performed on the current unit of work in an application selected by the database manager to resolve the deadlock.
binds_precompiles	The number of binds and pre-compiles attempted.

Application

Measurement	Description
agents_stolen	The number of times that agents are stolen from an application. Agents are stolen when an idle agent associated with an application is reassigned to work on a different application.
num_assoc_agents	At the application level, this is the number of sub-agents associated with an application. At the database level, it is the number of sub-agents for all applications.
total_sorts	The total number of sorts that have been executed.
total_sort_time	The total elapsed time (in milliseconds) for all sorts that have been executed.
sort_overflows	The total number of sorts that ran out of sort heap and may have required disk space for temporary storage.
total_hash_joins	The total number of hash joins executed.
total_hash_loops	The total number of times that a single partition of a hash join was larger than the available sort heap space.
hash_join_overflows	The number of times that hash join data exceeded the available sort heap space

Measurement	Description
hash_join_small_overflows	The number of times that hash join data exceeded the available sort heap space by less than 10%.
pool_data_l_reads	Indicates the number of logical read requests for data pages that have gone through the buffer pool.
pool_data_p_reads	The number of read requests that required I/O to get data pages into the buffer pool.
pool_data_writes	Indicates the number of times a buffer pool data page was physically written to disk.
pool_index_l_reads	Indicates the number of logical read requests for index pages that have gone through the buffer pool.
pool_index_p_reads	Indicates the number of physical read requests to get index pages into the buffer pool.
pool_index_writes	Indicates the number of times a buffer pool index page was physically written to disk.
pool_read_time	Provides the total amount of elapsed time spent processing read requests that caused data or index pages to be physically read from disk to buffer pool.
prefetch_wait_time	The time an application spent waiting for an I/O server (pre-fetcher) to finish loading pages into the buffer pool.
pool_data_to_estore	The number of buffer pool data pages copied to extended storage.
pool_index_to_estore	The number of buffer pool index pages copied to extended storage.
pool_data_from_estore	The number of buffer pool data pages copied from extended storage.
pool_index_from_estore	The number of buffer pool index pages copied from extended storage.

Measurement	Description
direct_reads	The number of read operations that do not use the buffer pool.
direct_writes	The number of write operations that do not use the buffer pool.
direct_read_reqs	The number of requests to perform a direct read of one or more sectors of data.
direct_write_reqs	The number of requests to perform a direct write of one or more sectors of data.
direct_read_time	The elapsed time (in milliseconds) required to perform the direct reads.
direct_write_time	The elapsed time (in milliseconds) required to perform the direct writes.
cat_cache_lookups	The number of times that the catalog cache was referenced to obtain table descriptor information.
cat_cache_inserts	The number of times that the system tried to insert table descriptor information into the catalog cache.
cat_cache_overflows	The number of times that an insert into the catalog cache failed due the catalog cache being full.
cat_cache_heap_full	The number of times that an insert into the catalog cache failed due to a heap-full condition in the database heap.
pkg_cache_lookups	The number of times that an application looked for a section or package in the package cache. At a database level, it indicates the overall number of references since the database was started, or monitor data was reset.
pkg_cache_inserts	The total number of times that a requested section was not available for use and had to be loaded into the package cache. This count includes any implicit prepares performed by the system.

Measurement	Description
appl_section_lookups	Lookups of SQL sections by an application from its SQL work area.
appl_section_inserts	Inserts of SQL sections by an application from its SQL work area.
uow_log_space_used	The amount of log space (in bytes) used in the current unit of work of the monitored application.
locks_held	The number of locks currently held.
deadlocks	The total number of deadlocks that have occurred.
lock_escals	The number of times that locks have been escalated from several row locks to a table lock.
x_lock_escals	The number of times that locks have been escalated from several row locks to one exclusive table lock, or the number of times an exclusive lock on a row caused the table lock to become an exclusive lock.
lock_timeouts	The number of times that a request to lock an object timed-out instead of being granted.
lock_waits	The total number of times that applications or connections waited for locks.
lock_wait_time	The total elapsed time waited for a lock.
locks_waiting	Indicates the number of agents waiting on a lock.
uow_lock_wait_time	The total amount of elapsed time this unit of work has spent waiting for locks.
rows_deleted	The number of row deletions attempted.
rows_inserted	The number of row insertions attempted.
rows_updated	The number of row updates attempted.
rows_selected	The number of rows that have been selected and returned to the application.

Measurement	Description
rows_written	The number of rows changed (inserted, deleted or updated) in the table.
rows_read	The number of rows read from the table.
int_rows_deleted	The number of rows deleted from the database as a result of internal activity.
int_rows_updated	The number of rows updated from the database as a result of internal activity.
int_rows_inserted	The number of rows inserted into the database as a result of internal activity caused by triggers.
open_rem_curs	The number of remote cursors currently open for this application, including those cursors counted by 'open_rem_curs_blk'.
open_rem_curs_blk	The number of remote blocking cursors currently open for this application.
rej_curs_blk	The number of times that a request for an I/O block at server was rejected and the request was converted to non-blocked I/O.
acc_curs_blk	The number of times that a request for an I/O block was accepted.
open_loc_curs	The number of local cursors currently open for this application, including those cursors counted by 'open_loc_curs_blk'.
open_loc_curs_blk	The number of local blocking cursors currently open for this application.
static_sql_stmts	The number of static SQL statements that were attempted.
dynamic_sql_stmts	The number of dynamic SQL statements that were attempted.
failed_sql_stmts	The number of SQL statements that were attempted, but failed.

Measurement	Description
commit_sql_stmts	The total number of SQL COMMIT statements that have been attempted.
rollback_sql_stmts	The total number of SQL ROLLBACK statements that have been attempted.
select_sql_stmts	The number of SQL SELECT statements that were executed.
uid_sql_stmts	The number of SQL UPDATE, INSERT, and DELETE statements that were executed.
ddl_sql_stmts	This element indicates the number of SQL Data Definition Language (DDL) statements that were executed.
int_auto_rebinds	The number of automatic rebinds (or recompiles) that have been attempted.
int_commits	The total number of commits initiated internally by the database manager.
int_rollbacks	The total number of rollbacks initiated internally by the database manager.
int_deadlock_rollbacks	The total number of forced rollbacks initiated by the database manager due to a deadlock. A rollback is performed on the current unit of work in an application selected by the database manager to resolve the deadlock.
binds_precompiles	The number of binds and pre-compiles attempted.

7 Click **Add** to place the selected counter on the resource list. Add all the desired resources to the list, and click **Close**.

8 Click **OK** in the DB2 dialog box to activate the monitor.

Configuring the Oracle Monitor

The Oracle server measures information from the V$SESSTAT and V$SYSSTAT Oracle V$ tables, and other table counters defined by the user in the custom query. In order to monitor the Oracle server, you must set up the monitoring environment as described below.

Note: The port you use to monitor an Oracle server through a firewall depends on the configuration of the Oracle server. Configuration information for the connection between the client and server is located in the Oracle client *tnsnames.ora* file.

To set up the Oracle monitor environment:

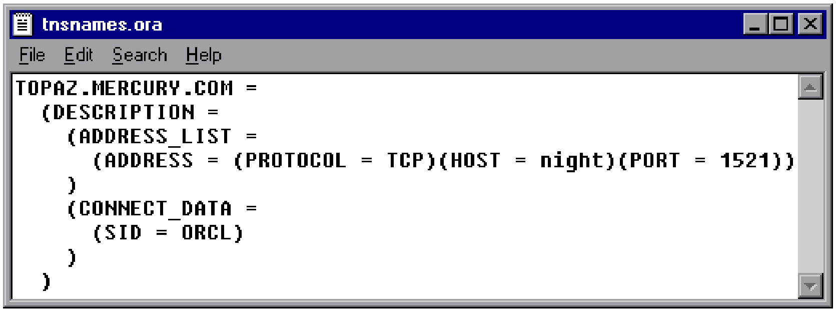
1. Ensure that the Oracle client libraries are installed on the Console machine.
2. Verify that *%OracleHome%\bin* is included in the path environment variable. If it is not, add it.
3. Configure the *tnsnames.ora* file on the Console machine so that the Oracle client can communicate with the Oracle server(s) you plan to monitor.

   You can configure connection parameters either manually, by editing the *tnsnames.ora* file in a text editor, or using the Oracle service configuration tool (for example, select **Start** > **Programs** > **Oracle for Windows NT** > **Oracle Net8 Easy Config**).

   You specify:

   - a new service name (TNS name) for the Oracle instance
   - TCP protocol
   - the host name (name of monitored server machine)
   - the port number (usually 1521)
   - the database SID (the default SID is ORCL).

For example:



```
TOPAZ.MERCURY.COM =
  (DESCRIPTION =
    (ADDRESS_LIST =
      (ADDRESS = (PROTOCOL = TCP)(HOST = night)(PORT = 1521))
    )
    (CONNECT_DATA =
      (SID = ORCL)
    )
  )
```

4 Obtain a username and password for the service from your database administrator, and ensure that the Console has database administrator privileges for the Oracle V$ tables (V$SESSTAT, V$SYSSTAT, V$STATNAME, V$INSTANCE, V$SESSION).

5 Verify connection with the Oracle server by performing *tns ping* from the Console machine. Note that there may be a problem connecting if the Oracle server is behind a DMZ/firewall that limits its communication to application servers accessing it.

6 Ensure that the registries are updated for the version of Oracle that you are using and that they have the following key:
HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE

7 Verify that the Oracle server you want to monitor is up and running.

Note: It is possible to monitor several Oracle database servers concurrently.

---

8 Run SQL*Plus from the Console and attempt to log in to the Oracle server(s) with the desired username/password/server combination.

9 Type SELECT * FROM V$SYSSTAT to verify that you can view the V$SYSSTAT table on the Oracle server. Use similar queries to verify that you can view the V$SESSTAT, V$SESSION, V$INSTANCE, V$STATNAME, and V$PROCESS tables on the server. Make sure that the Oracle bin directory is in the search path.

10 To change the length of each monitoring sample (in seconds), you need to edit the *dat\monitors\vmon.cfg* file in the ProTune root folder. The default rate is 10 seconds.

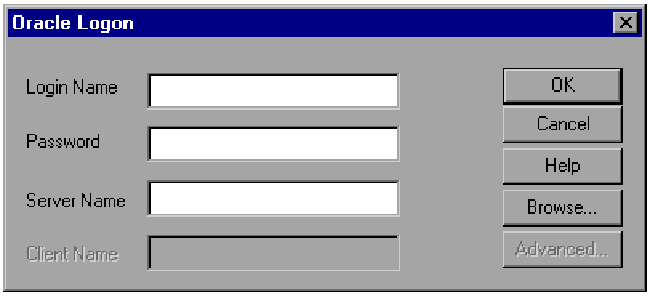
Note: The minimum sampling rate for the Oracle Monitor is 10 seconds. If you set the sampling rate at less than 10 seconds, the Oracle Monitor will continue to monitor at 10 second intervals.

Note: If a problem occurs in setting up the Oracle environment, view the error message issued by the Oracle server.

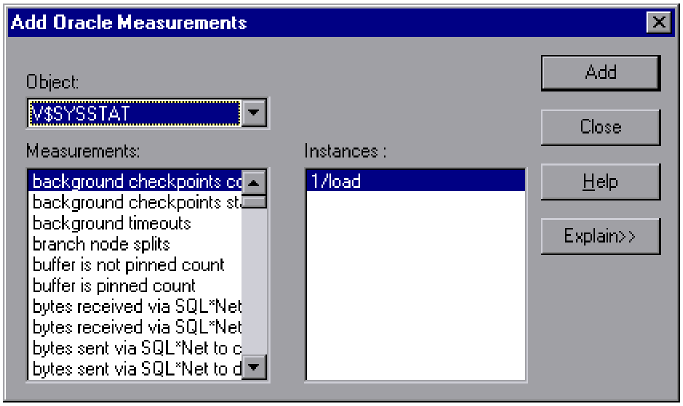
To configure the Oracle monitor:

1. Click the Oracle graph in the graph tree, and drag it into the right pane of the Run view.
2. Right-click the graph and choose **Add Measurement(s)**, or choose **Monitors** > **Add Online Measurement**.
3. In the Monitored Server Machines section of the Oracle dialog box, click **Add** to enter the server name or IP address of the machine you want to monitor. Select any platform, and click **OK**.
4. In the Resource Measurements section of the Oracle dialog box, click **Add** to select the measurements that you want to monitor.

   The Oracle Logon dialog box opens.



5 Enter your Login Name, Password, and Server Name, and click **OK**. The Add Oracle Measurements dialog box opens.



Object: V$SYSSTAT

Measurements:
background checkpoints co
background checkpoints st
background timeouts
branch node splits
buffer is not pinned count
buffer is pinned count
bytes received via SQL*Net
bytes received via SQL*Net
bytes sent via SQL*Net to c
bytes sent via SQL*Net to d

Instances : 1/load

6 Select an object, a measurement, and an instance. You can select multiple measurements using the **Ctrl** key. The instance is relevant only if multiple instances of the highlighted measurement are running. For a description of each measurement, click **Explain>>** to expand the dialog box. For instructions on creating custom queries, see "Custom Queries," on page 364.

The following measurements are most commonly used when monitoring the Oracle server (from the V$SYSSTAT table):

Measurement	Description
CPU used by this session	This is the amount of CPU time (in 10s of milliseconds) used by a session between the time a user call started and ended. Some user calls can be completed within 10 milliseconds and, as a result, the start and end user-call time can be the same. In this case, 0 milliseconds are added to the statistic. A similar problem can exist in the operating system reporting, especially on systems that suffer from many context switches.
Bytes received via SQL*Net from client	The total number of bytes received from the client over Net8

Measurement	Description
Logons current	The total number of current logons
Opens of replaced files	The total number of files that needed to be reopened because they were no longer in the process file cache
User calls	Oracle allocates resources (Call State Objects) to keep track of relevant user call data structures every time you log in, parse, or execute. When determining activity, the ratio of user calls to RPI calls gives you an indication of how much internal work gets generated as a result of the type of requests the user is sending to Oracle.
SQL*Net roundtrips to/from client	The total number of Net8 messages sent to, and received from, the client
Bytes sent via SQL*Net to client	The total number of bytes sent to the client from the foreground process(es)
Opened cursors current	The total number of current open cursors
DB block changes	Closely related to consistent changes, this statistic counts the total number of changes that were made to all blocks in the SGA that were part of an update or delete operation. These are changes that are generating redo log entries and hence will be permanent changes to the database if the transaction is committed. This statistic is a rough indication of total database work and indicates (possibly on a per-transaction level) the rate at which buffers are being dirtied.
Total file opens	The total number of file opens being performed by the instance. Each process needs a number of files (control file, log file, database file) in order to work against the database.

7 Click **Add** to place the selected measurement on the resource list. Add all the desired resources to the list, and click **Close**.

8 Click **OK** in the Oracle dialog box to activate the monitor.

Note: By default, the database returns the absolute value of a counter. However, by changing the IsRate setting in the *dat\monitors\vmon.cfg* file to 1, you can instruct the database to report a counter's rate value—the change in the counter per unit time.

Custom Queries

Using the custom query feature, you can define your own query to the Oracle database and view the result of this query—a single numerical value—in the Oracle online monitor graph. By defining your own query, you can monitor not only the V$SYSSTAT and V$SESSTAT table counters that are currently provided by the Oracle monitor, but other tables that contain useful performance information as well.

To create a custom query:

1. In the third line of the *vmon.cfg* file, CustomCounters=, indicate the number of custom counters you want to create.
2. Create a new section in the *vmon.cfg* file for the new counter. Each section has the following format:

```
[Custom2]

Name=Number of sessions

Description=This counter returns the number of sessions active.

Query=SELECT COUNT(*) FROM V$SESSION

IsRate=1
```

3. In the [Custom#] line, assign the next number in the sequence of counters to the new custom counter. Note that the custom counters must be in consecutive order, beginning with the number 0.
4. In the Name line, enter the name of the new counter.
5. In the Description line, enter the description of the counter that you want the help message to contain.

6. In the Query line, enter the text of the SQL query (on one line of the *vmon.cfg* file) that returns exactly one row from the database. This row must contain one column, a numerical value.
7. In the IsRate line, enter 0 if you want the database to report the counter as an absolute number. If you want the database to report the change in the counter per unit time, enter 1.

Note: Custom queries cannot return negative values.

Configuring the SQL Server Monitor

The SQL Server monitor measures the standard Windows resources on the SQL server machine.

Note: To monitor an SQL server through a firewall, use TCP, port 139.

To configure the SQL server monitor:

1. Click the SQL Server graph in the graph tree, and drag it into the right pane of the Run view.
2. Right-click the graph and choose **Add Measurement(s)**, or choose **Monitors** > **Add Online Measurement**.
3. In the Monitored Server Machines section of the SQL Server dialog box, click **Add** to enter the server name or IP address of the machine you want to monitor. Select the platform on which the machine runs, and click **OK**.

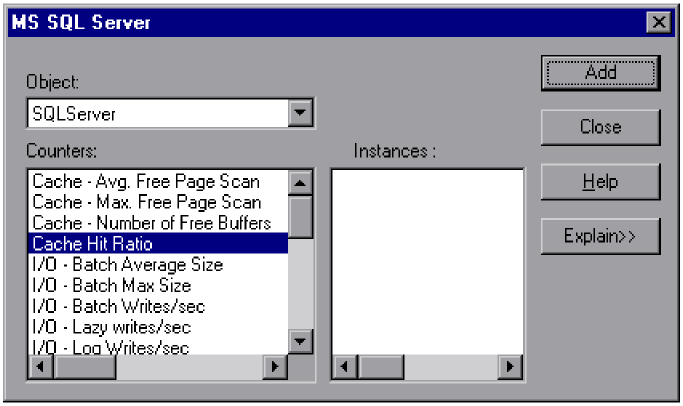
4 In the Resource Measurements section of the SQL Server dialog box, select the measurements you want to monitor. The following table describes the default counters that can be monitored on version 6.5 of the SQL Server:

Measurement	Description
% Total Processor Time (NT)	The average percentage of time that all the processors on the system are busy executing non-idle threads. On a multi-processor system, if all processors are always busy, this is 100%, if all processors are 50% busy this is 50% and if 1/4th of the processors are 100% busy this is 25%. It can be viewed as the fraction of the time spent doing useful work. Each processor is assigned an Idle thread in the Idle process which consumes those unproductive processor cycles not used by any other threads.
% Processor Time (Win 2000)	The percentage of time that the processor is executing a non-idle thread. This counter was designed as a primary indicator of processor activity. It is calculated by measuring the time that the processor spends executing the thread of the idle process in each sample interval, and subtracting that value from 100%. (Each processor has an idle thread which consumes cycles when no other threads are ready to run). It can be viewed as the percentage of the sample interval spent doing useful work. This counter displays the average percentage of busy time observed during the sample interval. It is calculated by monitoring the time the service was inactive, and then subtracting that value from 100%.
Cache Hit Ratio	The percentage of time that a requested data page was found in the data cache (instead of being read from disk)
I/O - Batch Writes/sec	The number of 2K pages written to disk per second, using Batch I/O. The checkpoint thread is the primary user of Batch I/O.
I/O - Lazy Writes/sec	The number of 2K pages flushed to disk per second by the Lazy Writer
I/O - Outstanding Reads	The number of physical reads pending

Measurement	**Description**
I/O - Outstanding Writes	The number of physical writes pending
I/O - Page Reads/sec	The number of physical page reads per second
I/O - Transactions/sec	The number of Transact-SQL command batches executed per second
User Connections	The number of open user connections

Note: To change the default counters for the SQL Server monitor, see "Changing a Monitor's Default Counters," on page 525.

5 To select additional measurements, click **Add.** A dialog box displaying the SQL Server object, its counters, and instances opens.



6 Select a counter and an instance. You can select multiple counters using the **Ctrl** key. The instance is relevant only if multiple instances of the highlighted counter are running. For a description of each counter, click **Explain>>** to expand the dialog box.

7 Click **Add** to place the selected counter on the resource list. Add all the desired resources to the list, and click **Close**.

8 Click **OK** in the SQL Server dialog box to activate the monitor.

Note: Certain measurements or counters are especially useful for determining server performance and isolating the cause of a bottleneck during an initial stress test on the SQL Server. For more information about these counters, see "Useful Counters for Stress Testing," on page 526.

Configuring the Sybase Monitor

The Sybase monitor enables monitoring of Sybase Adaptive Server Enterprise (Sybase ASE) servers (version 11 or later) on Windows and UNIX. The monitor connects to the Sybase ASE server (via the Adaptive Server Enterprise Monitor Server) and retrieves metrics from the server using standard, Sybase-provided libraries.

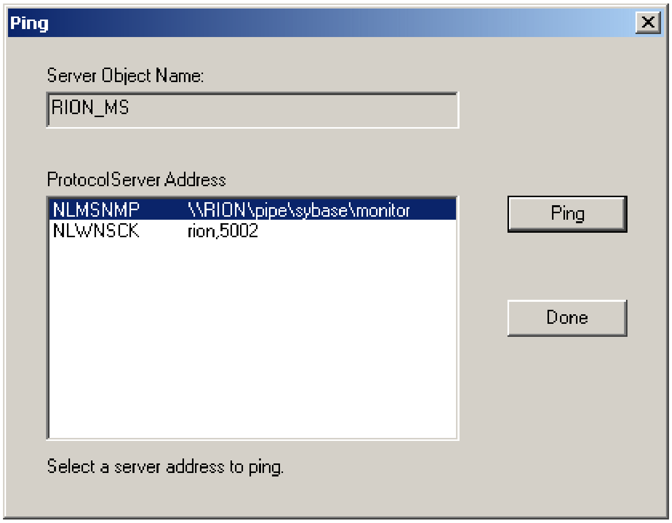
Note: When connecting to the monitored server, you connect to the Adaptive Server Enterprise Monitor Server, not the Sybase ASE server. The Adaptive Server Enterprise Monitor Server is an application that runs on the same machine as Sybase ASE server and retrieves performance information from it. The Adaptive Server Enterprise Monitor Server usually has the same name as the Sybase server, but with the suffix *_ms*.

In order to monitor the Sybase ASE server, you must first set up the Sybase monitor environment.

To set up the Sybase monitor environment:

1 Install the Sybase client files and libraries on the Console machine.

2 Verify a connection between the client and server on the Console machine. To do so, use the Sybase client's *dsedit* tool to ping the Adaptive Server Enterprise Monitor Server.



Server Object Name:
RION_MS

ProtocolServer Address

```
NLMSNMP     \\RION\pipe\sybase\monitor
NLWNSCK     rion,5002
```

Select a server address to ping.

Note: The port you use to monitor a Sybase server through a firewall depends on the configuration of the Sybase server. Configuration information for the connection between the client and server is located in the Sybase client *sql.ini* file.

To configure the Sybase ASE monitor:

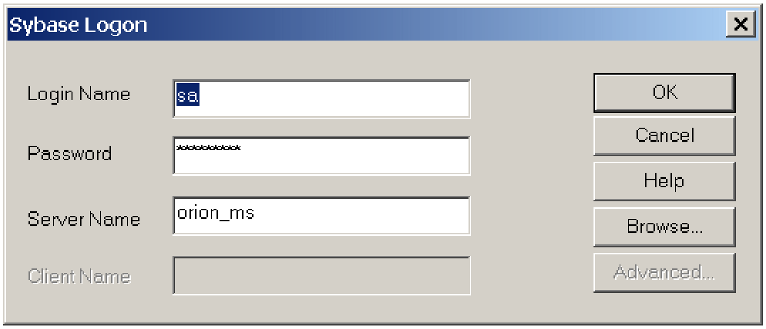
1 Click the Sybase graph in the graph tree, and drag it into the right pane of the Run view.

2 Right-click the graph and choose **Add Measurement(s)**, or choose **Monitors** > **Add Online Measurement.**

3 In the Monitored Server Machines section of the Sybase dialog box, click **Add** to enter the server name or IP address of the machine you want to monitor. Select any platform, and click **OK**.

4 In the Resource Measurements section of the Sybase dialog box, click **Add** to select the measurements that you want to monitor.

The Sybase Logon dialog box opens.



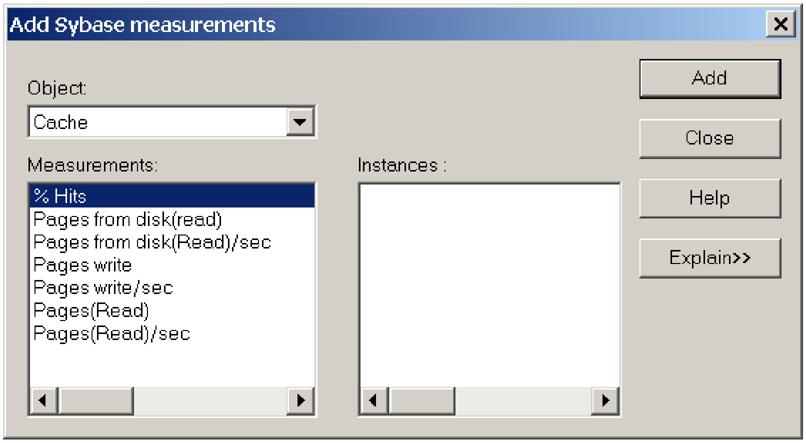
Login Name: sa

Password: **********

Server Name: orion_ms

5 Enter the login name and password of a user that has administrative privileges on the Sybase ASE server, as well as the Adaptive Server Enterprise Monitor Server name (usually the same name as the Sybase server but with the suffix *_ms*).

6 Click **OK**. The Add Sybase Measurements dialog box opens.



Object: Cache

Measurements:
% Hits
Pages from disk(read)
Pages from disk(Read)/sec
Pages write
Pages write/sec
Pages(Read)
Pages(Read)/sec

7 Select an object, measurement, and instance. You can select multiple measurements using the **CTRL** key. The instance is relevant only if multiple instances of the highlighted measurement are running. For a description of the measurements, click **Explain>>** to expand the dialog box.

The following measurements are available when monitoring a Sybase server:

Object	Measurement	Description
Network	Average packet size (Read)	Reports the number of network packets received
	Average packet size (Send)	Reports the number of network packets sent
	Network bytes (Read)	Reports the number of bytes received, over the sampling interval
	Network bytes (Read)/sec	Reports the number of bytes received, per second
	Network bytes (Send)	Reports the number of bytes sent, over the sampling interval
	Network bytes (Send)/sec	Reports the number of bytes sent, per second
	Network packets (Read)	Reports the number of network packets received, over the sampling interval
	Network packets (Read)/sec	Reports the number of network packets received, per second
	Network packets (Send)	Reports the number of network packets sent, over the sampling interval
	Network packets (Send)/sec	Reports the number of network packets sent, per second
Memory	Memory	Reports the amount of memory, in bytes, allocated for the page cache
Disk	Reads	Reports the number of reads made from a database device
	Writes	Reports the number of writes made to a database device
	Waits	Reports the number of times that access to a device had to wait

Object	Measurement	Description
Disk	Grants	Reports the number of times access to a device was granted
Engine	Server is busy (%)	Reports the percentage of time during which the Adaptive Server is in a "busy" state
	CPU time	Reports how much "busy" time was used by the engine
	Logical pages (Read)	Reports the number of data page reads, whether satisfied from cache or from a database device
	Pages from disk (Read)	Reports the number of data page reads that could not be satisfied from the data cache
	Pages stored	Reports the number of data pages written to a database device
Stored Procedures	Executed (sampling period)	Reports the number of times a stored procedure was executed, over the sampling interval
	Executed (session)	Reports the number of times a stored procedure was executed, during the session
	Average duration (sampling period)	Reports the time, in seconds, spent executing a stored procedure, over the sampling interval
	Average duration (session)	Reports the time, in seconds, spent executing a stored procedure, during the session
Locks	% Requests	Reports the percentage of successful requests for locks
	Locks count	Reports the number of locks. This is an accumulated value.

Object	Measurement	Description
Locks	Granted immediately	Reports the number of locks that were granted immediately, without having to wait for another lock to be released
	Granted after wait	Reports the number of locks that were granted after waiting for another lock to be released
	Not granted	Reports the number of locks that were requested but not granted
	Wait time (avg.)	Reports the average wait time for a lock
SqlSrvr	Locks/sec	Reports the number of locks. This is an accumulated value.
	% Processor time (server)	Reports the percentage of time that the Adaptive Server is in a "busy" state
	Transactions	Reports the number of committed Transact-SQL statement blocks (transactions)
	Deadlocks	Reports the number of deadlocks
Cache	% Hits	Reports the percentage of times that a data page read could be satisfied from cache without requiring a physical page read
	Pages (Read)	Reports the number of data page reads, whether satisfied from cache or from a database device
	Pages (Read)/sec	Reports the number of data page reads, whether satisfied from cache or from a database device, per second

Object	Measurement	Description
Cache	Pages from disk (Read)	Reports the number of data page reads that could not be satisfied from the data cache
	Pages from disk (Read)/sec	Reports the number of data page reads, per second, that could not be satisfied from the data cache
	Pages (Write)	Reports the number of data pages written to a database device
	Pages (Write)/sec	Reports the number of data pages written to a database device, per second
Process	% Processor time (process)	Reports the percentage of time that a process running a given application was in the "Running" state (out of the time that all processes were in the "Running" state)
	Locks/sec	Reports the number of locks, by process. This is an accumulated value.
	% Cache hit	Reports the percentage of times that a data page read could be satisfied from cache without requiring a physical page read, by process
	Pages (Write)	Reports the number of data pages written to a database device, by process
Transaction	Transactions	Reports the number of committed Transact-SQL statement blocks (transactions), during the session
	Rows (Deleted)	Reports the number of rows deleted from database tables during the session

Object	Measurement	Description
Transaction	Inserts	Reports the number of insertions into a database table during the session
	Updates	Reports the updates to database tables during the session
	Updates in place	Reports the sum of expensive, in-place and not-in-place updates (everything except updates deferred) during the session
	Transactions/sec	Reports the number of committed Transact-SQL statement blocks (transactions) per second
	Rows (Deleted)/sec	Reports the number of rows deleted from database tables, per second
	Inserts/sec	Reports the number of insertions into a database table, per second
	Updates/sec	Reports the updates to database tables, per second
	Updates in place/sec	Reports the sum of expensive, in-place and not-in-place updates (everything except updates deferred), per second

8 Click **Add** to place the selected measurement on the resource list. Add all the desired resources to the list, and click **Close**.

9 Click **OK** in the Sybase dialog box to activate the monitor.

23

Streaming Media Monitoring

During a session run, you can monitor the Windows Media Server and RealPlayer audio/video servers, as well as the RealPlayer and Media Player clients, in order to isolate server and client performance bottlenecks.

This chapter describes:

- Configuring the Windows Media Server Monitor
- Configuring the RealPlayer Server Monitor
- Viewing the RealPlayer Client Online Graph
- Viewing the Media Player Client Online Graph

Note: For instructions on recording a script containing streaming media functions, see the *ProTune Virtual User Generator User's Guide*.

About Streaming Media Monitoring

The streaming media monitors provide you with performance information for the Windows Media Server and RealPlayer audio/video servers, as well as the RealPlayer and Media Player clients. In order to obtain data for the Windows Media Server and RealPlayer Server, you need to activate the streaming media monitor before executing the session, and indicate which statistics and measurements you want to monitor. The RealPlayer Client and Media Player Client do not require pre-session activation or configuration.

Configuring the Windows Media Server Monitor

To monitor the Windows Media Server, you must first select the counters you want the Windows Media Server monitor to measure. You select these counters using the Windows Media Server dialog box.

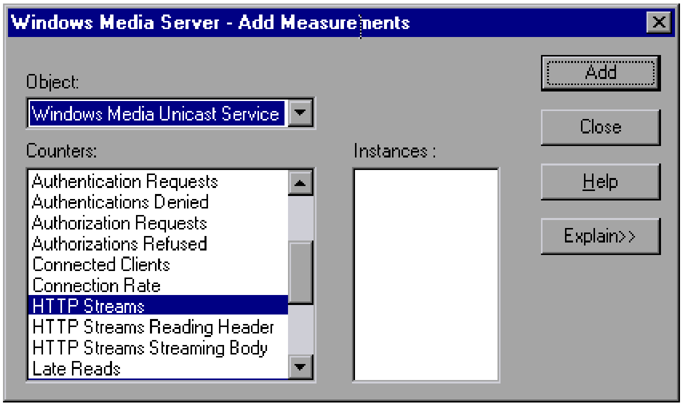
To configure the Windows Media Server monitor:

1. Click the Windows Media Server graph in the graph tree, and drag it into the right pane of the Run view.
2. Right-click the graph and choose **Add Measurement(s)**, or choose **Monitors** > **Add Online Measurement**.
3. In the Monitored Server Machines section of the Windows Media Server dialog box, click **Add** to enter the server name or IP address of the machine you want to monitor. Select the platform on which the machine runs, and click **OK**.
4. In the Resource Measurements section of the Windows Media Server dialog box, select the measurements you want to monitor. The following table describes the default counters that can be monitored:

Measurement	Description
Active Live Unicast Streams (Windows)	The number of live unicast streams that are being streamed
Active Streams	The number of streams that are being streamed
Active TCP Streams	The number of TCP streams that are being streamed
Active UDP Streams	The number of UDP streams that are being streamed
Aggregate Read Rate	The total, aggregate rate (bytes/sec) of file reads
Aggregate Send Rate	The total, aggregate rate (bytes/sec) of stream transmission
Connected Clients	The number of clients connected to the server
Connection Rate	The rate at which clients are connecting to the server
Consoles	The number of Consoles currently connected to the server
HTTP Streams	The number of HTTP streams being streamed

Measurement	Description
Late Reads	The number of late read completions per second
Pending Connections	The number of clients that are attempting to connect to the server, but are not yet connected. This number may be high if the server is running near maximum capacity and cannot process a large number of connection requests in a timely manner.
Stations	The number of station objects that currently exist on the server
Streams	The number of stream objects that currently exist on the server
Stream Errors	The cumulative number of errors occurring per second

5 To select additional measurements, click **Add.** The Windows Media Server - Add Measurements dialog box opens, displaying the Windows Media Unicast Service object, its counters, and instances.



Windows Media Server - Add Measurements

Object: Windows Media Unicast Service

Counters:
- Authentication Requests
- Authentications Denied
- Authorization Requests
- Authorizations Refused
- Connected Clients
- Connection Rate
- HTTP Streams
- HTTP Streams Reading Header
- HTTP Streams Streaming Body
- Late Reads

Instances :

6 Select a counter and an instance. You can select multiple counters using the **Ctrl** key. The instance is relevant only if multiple instances of the highlighted counter are running. For a description of each counter, click **Explain>>** to expand the dialog box.

7 Click **Add** to place the selected counter on the resource list. Add all the desired resources to the list, and click **Close**.

8 Click **OK** in the Windows Media Server dialog box to activate the monitor.

Configuring the RealPlayer Server Monitor

To monitor the RealPlayer Server, you must first select the counters you want the RealPlayer Server monitor to measure. You select these counters using the Real Server dialog box.

To configure the RealPlayer Server monitor:

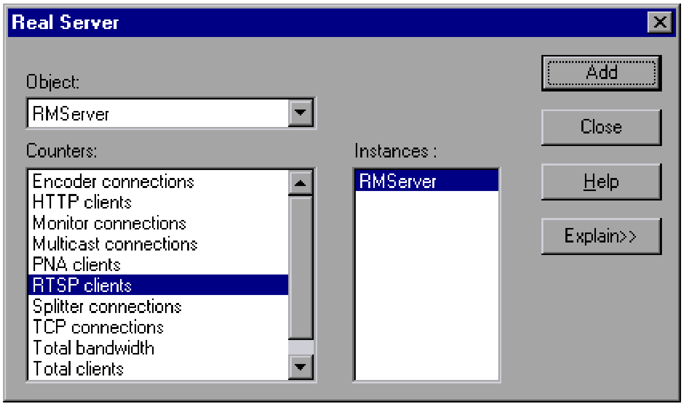
1 Click the Real Server graph in the graph tree, and drag it into the right pane of the Run view.

2 Right-click the graph and choose **Add Measurement(s)**, or choose **Monitors** > **Add Online Measurement**.

3 In the Monitored Server Machines section of the Real Server dialog box, click **Add** to enter the server name or IP address of the machine you want to monitor. Select the platform on which the machine runs, and click **OK**.

4 Click **Add** in the Resource Measurements section of the Real Server dialog box to select the measurements that you want to monitor.

Another Real Server dialog box opens, displaying the counters that can be monitored.



Object: RMServer

Counters:
Encoder connections
HTTP clients
Monitor connections
Multicast connections
PNA clients
RTSP clients
Splitter connections
TCP connections
Total bandwidth
Total clients

Instances : RMServer

5 Select a counter and an instance. You can select multiple counters using the **Ctrl** key. The instance is relevant only if multiple instances of the highlighted counter are running. For a description of each counter, click **Explain>>** to expand the dialog box.

The following table describes the default counters that can be monitored:

Measurement	Description
Encoder Connections	The number of active encoder connections
HTTP Clients	The number of active clients using HTTP
Monitor Connections	The number of active server monitor connections
Multicast Connections	The number of active multicast connections
PNA Clients	The number of active clients using PNA
RTSP Clients	The number of active clients using RTSP
Splitter Connections	The number of active splitter connections
TCP Connections	The number of active TCP connections
Total Bandwidth	The number of bits per second being consumed
Total Clients	The total number of active clients
UDP Clients	The number of active UDP connections

6 Click **Add** to place the selected counter on the resource list. Add all the desired resources to the list, and click **Close**.

7 Click **OK** in the Real Server dialog box to activate the monitor.

Viewing the RealPlayer Client Online Graph

You can view the RealPlayer Client online monitor graph by dragging it from the graph tree into the right pane of the Run view.

The following table describes the RealPlayer Client measurements that are monitored:

Measurement	Description
Current Bandwidth (Kbits/sec)	The number of kilobytes in the last second
Buffering Event Time (sec)	The average time spent on buffering
Network Performance	The ratio (percentage) between the current bandwidth and the actual bandwidth of the clip
Percentage of Recovered Packets	The percentage of error packets that were recovered
Percentage of Lost Packets	The percentage of packets that were lost
Percentage of Late Packets	The percentage of late packets
Time to First Frame Appearance (sec)	The time for first frame appearance (measured from the start of the replay)
Number of Buffering Events	The average number of all buffering events
Number of Buffering Seek Events	The average number of buffering events resulting from a seek operation
Buffering Seek Time	The average time spent on buffering events resulting from a seek operation
Number of Buffering Congestion Events	The average number of buffering events resulting from network congestion
Buffering Congestion Time	The average time spent on buffering events resulting from network congestion

Measurement	Description
Number of Buffering Live Pause Events	The average number of buffering events resulting from live pause
Buffering Live Pause Time	The average time spent on buffering events resulting from live pause

Viewing the Media Player Client Online Graph

You can view the Windows Media Player Client online monitor graph by dragging it from the graph tree into the right pane of the Run view.

The following table describes the Media Player Client measurements that are monitored:

Measurement	Description
Stream Quality (Packet-level)	The percentage ratio of packets received to total packets
Current bandwidth (Kbits/sec)	The number of kbits per second received
Stream Packet Rate	The number of packets received
Total number of recovered packets	The number of lost packets that were recovered. This value is only relevant during network playback.
Total number of lost packets	The number of lost packets that were not recovered. This value is only relevant during network playback.
Stream Quality (Sampling-level)	The percentage of stream samples received on time (no delays in reception)

24

ERP Server Resource Monitoring

During a session step run, you can monitor ERP server resources in order to isolate server performance bottlenecks.

This chapter describes:

- Setting Up the SAP Monitor
- Configuring the SAP Monitor

About ERP Server Resource Monitoring

The ERP server resource monitor provides you with performance information for the SAP R/3 system server. You can use the SAP monitor to view:

- the number of configured instances for each SAP system.
- data for all application instances (not just the one you logged on to)
- transactions used and the users that call them
- number of users working on the different instances
- performance history for recent periods of all instances
- response time distribution and resource consumption for any application server
- application server workload for today or for a recent period

In order to obtain this data, you need to activate the ERP server resource monitor before executing the session step, and indicate which statistics and measurements you want to monitor.

Setting Up the SAP Monitor

Before monitoring a SAP R/3 system server, you must set up the server monitor environment.

To set up the SAP monitor environment:

1. Install the SAP GUI client on the Console machine.
2. Click **F6** to check whether you can access the st03 transaction and query for *last minute load* information. If this functionality is not already enabled, enable it from the SAP R/3 client on the Console machine, using the username and password defined in the Console.

Configuring the SAP Monitor

To monitor a SAP R/3 system server, you must select the counters you want the SAP monitor to measure. You select these counters using the Add SAP Monitor Measurements dialog box.

Note: The SAP R/3 performance monitor supports SAP server versions 3.1 to 4.6, regardless of the SAP R/3 server's operating system and the platform on which it is installed.

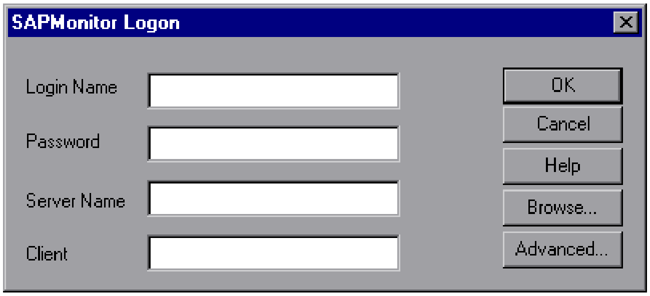
To configure the SAP monitor:

1. Click the SAP graph in the graph tree, and drag it into the right pane of the Run view.
2. Right-click the graph and choose **Add Measurement(s)**, or choose **Monitors** > **Add Online Measurement**.
3. In the Monitored Server Machines section of the SAP dialog box, click **Add** to enter the server name or IP address of the machine you want to monitor. Select the platform on which the machine runs, and click **OK**.

Note: You can also specify a system number and IP address in the Add Machine dialog box using the following format:
<system number:IP address>
For example: 199.35.106.162:00

---

4 Click **Add** in the Resource Measurements section of the SAP dialog box. The SAP Monitor Logon dialog box opens.



5 Enter your Login Name, Password, Server Name, and Client.

Note: If you want to connect to the SAP monitor through a router, you need to enter the router string into the Server Name field. A router string has the format:
<RouterString/ServerIP/S/sapdpxx>

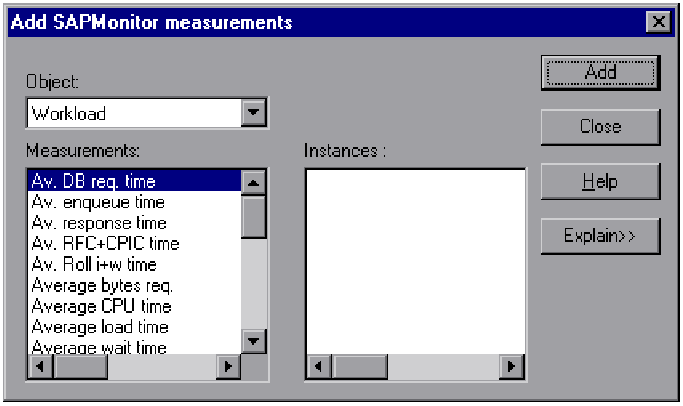
where RouterString is /H/<IP_ADDRESS>/H/<IP_ADDRESS>/H/
ServerIP is the application server IP address
and *xx* is the system number.

For example, if the router string = /H/199.35.107.9/H/204.79.199.244/H/, application server IP address = 172.20.11.6, and the system number = 00, you should enter the following string into the Server Name field:

/H/199.35.107.9/H/204.79.199.244/H/172.20.11.6/S/sapdp00

---

6 Click **OK**. The Add SAP Monitor Measurements dialog box opens.



Add SAPMonitor measurements

Object: Workload

Measurements:

Av. DB req. time
Av. enqueue time
Av. response time
Av. RFC+CPIC time
Av. Roll i+w time
Average bytes req.
Average CPU time
Average load time
Average wait time

Instances :

7 Select an object, a measurement, and an instance. You can select multiple measurements using the **Ctrl** key. The instance is relevant only if multiple instances of the highlighted measurement are running. For a description of each measurement, click **Explain>>** to expand the dialog box.

The following are the most commonly monitored counters:

Measurement	Description
Average CPU time	The average CPU time used in the work process.
Average response time	The average response time, measured from the time a dialog sends a request to the dispatcher work process, through the processing of the dialog, until the dialog is completed and the data is passed to the presentation layer. The response time between the SAP GUI and the dispatcher is not included in this value.
Average wait time	The average amount of time that an unprocessed dialog step waits in the dispatcher queue for a free work process. Under normal conditions, the dispatcher work process should pass a dialog step to the application process immediately after receiving the request from the dialog step. Under these conditions, the average wait time would be a few milliseconds. A heavy load on the application server or on the entire system causes queues at the dispatcher queue.
Average load time	The time needed to load and generate objects, such as ABAP source code and screen information, from the database.
Database calls	The number of parsed requests sent to the database.
Database requests	The number of logical ABAP requests for data in the database. These requests are passed through the R/3 database interface and parsed into individual database calls. The proportion of database calls to database requests is important. If access to information in a table is buffered in the SAP buffers, database calls to the database server are not required. Therefore, the ratio of calls/requests gives an overall indication of the efficiency of table buffering. A good ratio would be 1:10.

Measurement	Description
GUI time	The GUI time is measured in the work process and is the response time between the dispatcher and the GUI.
Roll ins	The number of rolled-in user contexts.
Roll outs	The number of rolled-out user contexts.
Roll in time	The processing time for roll ins.
Roll out time	The processing time for roll outs.
Roll wait time	The queue time in the roll area. When synchronous RFCs are called, the work process executes a roll out and may have to wait for the end of the RFC in the roll area, even if the dialog step is not yet completed. In the roll area, RFC server programs can also wait for other RFCs sent to them.
Average time per logical DB call	The average response time for all commands sent to the database system (in milliseconds). The time depends on the CPU capacity of the database server, the network, the buffering, and on the input/output capabilities of the database server. Access times for buffered tables are many magnitudes faster and are not considered in the measurement.

8 Click **Add** to place the selected measurement on the resource list. Add all the desired resources to the list, and click **Close**.

9 Click **OK** in the SAP dialog box to activate the monitor.

25

Java Performance Monitoring

During a session step run, you can monitor the resource usage of Enterprise Java Bean (EJB) objects, Java-based applications, and the TowerJ Java virtual machine, using the Java performance monitors:

This chapter describes:

- EJB Performance Monitoring
- JProbe Performance Monitoring
- Sitraka JMonitor Performance Monitoring
- TowerJ Performance Monitoring

About Java Performance Monitoring

The Java performance monitors provide you with performance information for Enterprise Java Bean (EJB) objects and Java-based applications, using the EJB, JProbe, Sitraka JMonitor, and the TowerJ Java virtual machine during session step execution. In order to obtain this data, you need to activate the Java performance monitors before executing the session step, and indicate which statistics and measurements you want to monitor.

EJB Performance Monitoring

Support Matrix:

Application Server	Version	Platform
WebLogic	4.x; 5.1; 6.0; 6.1	Windows; Solaris; AIX
WebSphere	3.x; 4.x	Windows; Solaris; AIX
Oracle 9i	1.0.2.2	Windows; Solaris; AIX

You can monitor Enterprise Java Bean (EJB) objects on a WebLogic, WebSphere, or Oracle 9iAS application server during a session step run using the EJB performance monitor. In order to monitor EJB objects, you must first install the EJB monitor, run the monitor detector, and activate the EJB monitor on the application server machine. You then configure the EJB monitor on the client machine by selecting the counters you want the monitor to measure.

Note: The server side installation contains new EJBDetector support files for generating EJB Vuser scripts. For more information on the EJBDetector, see the *ProTune Virtual User Generator User's Guide*.

Installing the EJB Monitor and Running the Monitor Detector

Before EJB objects can be monitored, you must install the EJB monitor support files, and verify that you have a valid JDK environment on the application server machine. You then prepare the EJB monitor for monitoring by running the monitor detector from the batch file, or from the command line.

To install the EJB monitor support files:

Create a home directory for the Mercury Interactive EJB support files—for example, MERC_MONITOR_HOME—and unzip the *<ProTune_Installation>\Add-ins\J2EE\EJB\jmonitor_<platform>.jar* file into that directory.

On UNIX platforms, use the jar utility to extract the installation jar:

Change to the MERC_MONITOR_HOME directory and type the following command:

```
jar -xvf <path to your jmonitor_<platform>.jar>
```

To run the monitor detector from the batch file:

1 Open the *env.cmd* (NT) or *env.sh* (UNIX) file and set the following variables:

JAVA_HOME Specify the root directory of the JDK installation.

APP_SERVER_DRIVE Specify the drive on which the application server is installed (for NT only).

DETECTOR_INS_DIR Specify the root directory of the Detector installation.

APP_SERVER_ROOT Follow these guidelines:
BEA WebLogic Servers 4.x and 5.x: Specify the application server root directory.
BEA WebLogic Servers 6.x: Specify the full path of the domain folder.
WebSphere Servers 3.x and 4.0: Specify the application server root directory.
Oracle OC4J: Specify the application server root directory.
Sun J2EE Server: Specify the full path to the deployable *.ear* file or directory containing a number of *.ear* files.

EJB_DIR_LIST (optional) Specify a list of directories/files, separated by ';' and containing deployable *.ear/.jar* files, and any additional classes directory or *.jar* files or used by your EJBs under test.

2 Run the *Mon_Detector.cmd* (NT) or *Mon_Detector.sh* (UNIX) batch file to collect information about the EJBs deployed. Running the monitor detector generates the following three files in the *<MERC_MONITOR_HOME>*\dat directory: *ejb_monitor.hooks*; *cjhook.ini*; and *regmon.properties*. These files contain information about the EJBs detected on the application server.

Note: You must run the monitor detector each time you add, change, or delete EJBs on the application server.

To run the monitor detector from a command line:

1 Add *<MERC_MONITOR_HOME>\classes*, *<MERC_MONITOR_HOME>\dat*, and the *<MERC_MONITOR_HOME>\classes\xerces.jar* file to the CLASSPATH environment variable.

2 Use the java MonDetect *<search root dir>* command line to collect information about the EJBs deployed.

<search root dir> Specify one or more directories or files in which to search for EJBs (separated by semicolons). Follow these guidelines:
BEA WebLogic Servers 4.x and 5.x: Specify the application server root directory.
BEA WebLogic Servers 6.x: Specify the full path of the domain folder.
WebSphere Servers 3.x and 4.0: Specify the application server root directory.
Oracle OC4J: Specify the application server root directory.
Sun J2EE Server: Specify the full path to the deployable *.ear* file or directory containing a number of *.ear* files.

Note that you can also specify a search list of directories and/or files to search. If unspecified, the CLASSPATH will be searched.

Running the monitor detector generates the following three files in the <*MERC_MONITOR_HOME*>\dat directory: *ejb_monitor.hooks*; *cjhook.ini*; and *regmon.properties*. These files contain information about the EJBs detected on the application server.

Note: You must run the monitor detector each time you add, change, or delete EJBs on the application server.

Configuring the EJB Monitor on the Application Server

After you have installed Mercury Interactive's EJB monitor support files on your WebLogic, WebSphere, or Oracle 9iAS machine, you must configure the application server to run with EJB monitor support.

Note: It is important to set the environment variables in the order in which they appear below.

WebLogic Server

The WebLogic 4.x-5.x server and the WebLogic 6.x server must be configured differently.

To configure the WebLogic 4.x-5.x server:

1 Copy the <*WebLogic Home*>*startWeblogic.cmd* file into <*WebLogic Home*>*startWeblogicMercury.cmd* so that the file is backed up.

2 Open the <*WebLogic Home*>*startWeblogicMercury.cmd* file.

3 In the 'runWebLogicJava' section of the file, after the WEBLOGIC_CLASSPATH environment settings, set the following environment variables:

For Windows platforms:

```
set MERC_MONITOR_HOME=<EJB Monitor Home Directory>

set CLASSPATH=%MERC_MONITOR_HOME%\dat
set JAVA_CLASSPATH=%MERC_MONITOR_HOME%\dat;%MERC_
MONITOR_HOME%\classes;%MERC_MONITOR_HOME%\classes\
xerces.jar;%JAVA_CLASSPATH%

set PATH=%PATH%;%MERC_MONITOR_HOME%\bin
```

For UNIX platforms:

```
MERC_MONITOR_HOME <EJB Monitor Home Directory>

CLASSPATH ${MERC_MONITOR_HOME}/dat

JAVA_CLASSPATH ${MERC_MONITOR_HOME}/dat:${MERC_MONITOR_
HOME}/classes:${MERC_MONITOR_HOME}/classes/xerces.jar:${JAVA_
CLASSPATH}

LD_LIBRARY_PATH ${LD_LIBRARY_PATH}:${MERC_MONITOR_HOME}/bin

export CLASSPATH

export LD_LIBRARY_PATH

export JAVA_CLASSPATH
```

Note: For IBM AIX platform replace LD_LIBRARY_PATH with LIBPATH. Replace <*EJB Monitor Home Directory*> with the EJB monitor installation root directory. Note that on UNIX platforms you may have to export the library path variables.

4 In the same section of the file, add a parameter to the command line:

```
-Xrunjdkhook.
```

For example on Windows platforms:

```
%JAVA_HOME%\bin\java -ms64m -mx64m -Xrunjdkhook -classpath
%JAVA_CLASSPATH%  -Dweblogic.class.path=%WEBLOGIC_CLASSPATH%
-Dweblogic.home=. -Djava.security.manager
-Djava.security.policy==.\weblogic.policy weblogic.Server
```

Note: For Solaris installation only.
If you are using JDK 1.2.x add a parameter to the command line:

```
-Dweblogic.classloader.preprocessor=com.mercuryinteractive.aim.
 MercuryWL5Preprocessor
```

for example, on Windows platforms:

```
%JAVA_HOME%\bin\java -ms64m -mx64m -classpath %JAVA_CLASSPATH%
-Dweblogic.classloader.preprocessor=com.mercuryinteractive.aim.
 MercuryWL5Preprocessor
-Dweblogic.class.path=%WEBLOGIC_CLASSPATH%
-Dweblogic.home=. -Djava.security.manager
-Djava.security.policy==.\weblogic.policy weblogic.Server
```

5 Run the <*WebLogic Home*>\startWeblogicMercury.cmd file.

To configure the WebLogic 6.x server:

1 Make a backup copy of the <*WebLogic Home*>\config\\<*domain name*>\startWeblogic.cmd file.

2 Open the <*WebLogic Home*>\config\\<*domain name*>\startWeblogicMercury.cmd file.

3 In the 'runWebLogic' section of the file, set the following environment variables:

For Windows platforms:

```
set MERC_MONITOR_HOME=<your MERC_MONITOR_HOME directory>

set CLASSPATH=%CLASSPATH%;%MERC_MONITOR_HOME%\dat;%
MERC_MONITOR_HOME%\classes;%MERC_MONITOR_HOME%\classes\
xerces.jar

set PATH=%PATH%;%MERC_MONITOR_HOME%\bin
```

For UNIX platforms:

```
MERC_MONITOR_HOME <EJB Monitor Home Directory>
```

```
CLASSPATH ${JAVA_CLASSPATH}:${MERC_MONITOR_HOME}/dat:$
{MERC_MONITOR_HOME}/classes:${MERC_MONITOR_HOME}/classes/
xerces.jar

LD_LIBRARY_PATH ${LD_LIBRARY_PATH}:${MERC_MONITOR_HOME}/
bin

export CLASSPATH

export LD_LIBRARY_PATH
```

Note: For IBM AIX platform replace LD_LIBRARY_PATH with LIBPATH. Replace <*EJB Monitor Home Directory*> with the EJB monitor installation root directory. Note that on UNIX platforms you may have to export the library path variables.

4 In the same section of the file add a parameter to the command line:

```
-Xrunjdkhook.
```

for example, on Windows platforms:

```
"%JAVA_HOME%\bin\java" -hotspot -ms64m -mx64m -Xrunjdkhook -classpath
%CLASSPATH% -Dweblogic.Domain=mydomain
-Dweblogic.Name=myserver "-Dbea.home=f:\bea"
“-Djava.security.policy==f:\bea\wlserver6.0/lib/weblogic.policy"
-Dweblogic.management.password=%WLS_PW% weblogic.Server
```

5 Run the <*WebLogic Home*>*config*\<*domain name*>\ *startWeblogicMercury.cmd* file.

WebSphere Server - Versions 3.0 and 3.5

By default, the WebSphere 3.x application server runs as an automatic service, upon machine startup. Since Mercury Interactive does not currently support ProTune EJB monitoring on a WebSphere server run as an automatic service, you must change the default WebSphere server startup to *manual*.

To change the default WebSphere 3.x server startup:

1. Select **Start** > **Settings** > **Control Panel**.
2. Double-click **Services**.
3. Select **IBM WS AdminServer**, and click the **Stop** button.
4. Double click **IBM WS AdminServer**, and select the **Manual** Startup Type.
5. Click **OK** to save your settings and close the dialog box.

You can now start the WebSphere Server from <*WebSphere Home*>\AppServer\bin\debug\adminserver.bat, instead of using the automatic service.

To add ProTune EJB monitor support to the WebSphere 3.x server:

1. Make a backup copy of the <*WebSphere Home*>*AppServer**bin**debug**adminserver.bat* file.
2. Open the <*WebSphere Home*>*AppServer**bin**debug**adminserver.bat* file.
3. Add the following environment variables at the end of the 'SET_CP' section:

   For Windows platforms:

   ```
   set CLASSPATH=<MERC_MONITOR_HOME>\dat;<MERC_MONITOR_
   HOME>\classes;<MERC_MONITOR_HOME>\classes\xerces.jar;
   %CLASSPATH%

   set PATH=%PATH%;<MERC_MONITOR_HOME>\bin
   ```

   For UNIX platforms:

   ```
   CLASSPATH ${MERC_MONITOR_HOME}/dat:${MERC_MONITOR_HOME}/
   classes:${MERC_MONITOR_HOME}/classes/xerces.jar:${CLASSPATH}

   LD_LIBRARY_PATH ${LD_LIBRARY_PATH}:${MERC_MONITOR_HOME}/bin

   export CLASSPATH
   ```

```
export LD_LIBRARY_PATH
```

Note: For IBM AIX platform replace LD_LIBRARY_PATH with LIBPATH. Replace <*EJB Monitor Home Directory*> with the EJB monitor installation root directory. Note that on UNIX platforms you may have to export the library path variables.

Note: For Solaris installation only.
If you are working with JRE1.2.x, you must download the patch file, PQ46831.jar, from IBM's Web site or FTP site:
http://www-3.ibm.com/software/webservers/appserv/efix-archive.html
ftp://ftp.software.ibm.com/software/websphere/appserv/support/fixes/pq46831/
Make sure to download the version that corresponds to your server version.
Add the patch file to the classpath:

```
setenv CLASSPATH PQ46831.jar:${CLASSPATH}
```

4 Run the *adminserver.bat* file.

5 Open the WebSphere Advanced Administrative Console, and select **View** > **Topology**.

6 Expand the WebSphere Administrative Domain tree by selecting <**server machine name**> > **Default Server**.

7 Select the **General** tab in the Application Server:Default Server window.

8 Type -Xrunjdkhook in the command line Arguments box, and click **Apply**.

If you are working with a WebSphere 3.0 Server with JDK1.1.7 IBM, double-click on **Environment**. Type _CLASSLOAD_HOOK in the Variable Name box, and jdkhook in the Value box. Click the **Add**, **OK**, and **Apply** buttons.

Note: For Solaris installation only.
If you are working with a WebSphere 3.5 Server with J2RE1.2.x, in the Command Line Arguments box, type the following and click **Apply**:

```
-Dcom.ibm.ejs.sm.server.ServiceInitializer=com.ibm.ejs.sm.server.WilyInitializer
-Dcom.ibm.websphere.introscope.implClass=com.mercuryinteractive.aim.
MercuryWASPreprocessor
```

---

9 Close the WebSphere Advanced Administrative Console.

10 Close and restart the *adminserver.bat* file.

WebSphere Server - Version 4.0

You can start the WebSphere 4.0 server using the startServerBasic.bat file or the startServer.bat file.

To configure the WebSphere 4.0 server:

1 Ensure that the WebSphere Administrative Server is running, and start the Administrator Console.

2 In the WebSphere Administrative Domain tree, expand the Nodes, hostname, and Application Servers subtrees, and select the Default Server (or the Application Server you wish to use with JMonitor).

3 For Windows 2000/NT or Solaris, click the **General** tab, and add the following variables to the **Environment** box:

Note: Replace <*EJB Monitor Home Directory*> with the EJB monitor installation root directory.

For Windows 2000/NT:

name=PATH

value=<*EJB Monitor Home Directory*>\bin

For Solaris:

name=LD_LIBRARY_PATH

value=<*EJB Monitor Home Directory*>/bin

Click **OK** to close the **Environment Editor** dialog box.

For AIX:

If the LIBPATH environment variable has been changed, you need to link the EJB monitor libraries to the /usr/lib directory.

Add the following command:

#ln -s <*EJB Monitor Home Directory*>/bin/libcjhook_mon.so /usr/lib/libcjhook_mon.so

#ln -s <*EJB Monitor Home Directory*>/bin/libconfig.so /usr/lib/libconfig.so

#ln -s <*EJB Monitor Home Directory*>/bin/libjdkhook.so /usr/lib/libjdkhook.so

#ln -s <*EJB Monitor Home Directory*>/bin/libmlib_ds.so /usr/lib/libcjhook_mon.so

#ln -s <*EJB Monitor Home Directory*>/bin/libmosifs.so /usr/lib/libmosifs.so

#ln -s <*EJB Monitor Home Directory*>/bin/libthrdutil.so /usr/lib/libthrdutil.so

---

Note: You will likely require root permissions in order to create the link. Alternatively, you can place the link in WebSphere's /bin directory (usually /usr/WebSphere/AppServer/bin).

---

4. Click the **JVM Settings** tab in the WebSphere Administrative Console, and add the following values to the classpath:

Note: Replace <*EJB Monitor Home Directory*> with the EJB monitor installation root directory.

For Windows 2000/NT:

<*EJB Monitor Home Directory*>\dat

<*EJB Monitor Home Directory*>\classes

<*EJB Monitor Home Directory*>\classes\xerces.jar

For Solaris or AIX:

<*EJB Monitor Home Directory*>/dat

<*EJB Monitor Home Directory*>/classes

<*EJB Monitor Home Directory*>/classes/xerces.jar

Note: For Solaris installation only.
If you are working with JRE1.2.x, you must download the patch file, PQ46831.jar, from IBM's Web site or FTP site:
http://www-3.ibm.com/software/webservers/appserv/efix-archive.html
ftp://ftp.software.ibm.com/software/websphere/appserv/support/fixes/pq46831/
Make sure to download the version that corresponds to your server version.
Add the following value to the classpath:
<*EJB Monitor Home Directory*>/classes/PQ46831.jar

---

5 Click the **Advanced JVM Settings** button. In the Command line arguments field, add the following value for Windows 2000/NT, Solaris, and AIX:

-Xrunjdkhook

Note: For Solaris installation only.
If you are working with JRE1.2.x, instead of -Xrunjdkhook
add the following value:
-Dcom.ibm.ejs.sm.server.ServiceInitializer=com.ibm.ejs.sm.server.
WilyInitializer
-Dcom.ibm.websphere.introscope.implClass=com.mercuryinteractive.
aim.MercuryWASPreprocessor

---

6 Click the **OK** and **Apply** buttons to save the changes for the Application server. You can now start and stop your WebSphere server using the ProTune EJB Monitor.

Oracle 9iAS Server

Once you have configured the support files and set up the JDK environment on the Oracle 9iAS application server, run the *oc4jMonitor.cmd* file on an NT machine, or the *oc4jMonitor.sh* file on a UNIX machine. The application server starts running with EJB monitor support.

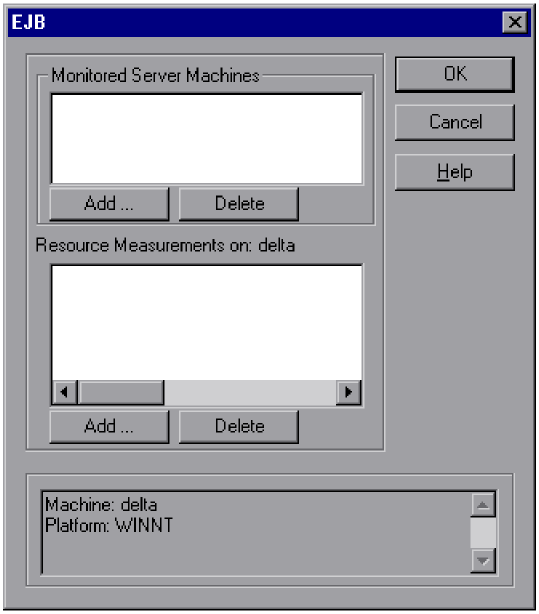
Configuring the EJB Monitor on the Client Machine

To monitor EJB performance, you must select the counters you want the EJB monitor to measure. You select these counters using the Console's EJB Monitor Configuration dialog box.

To configure the EJB monitor:

1 Click the EJB graph in the graph tree, and drag it into the right pane of the Run view.

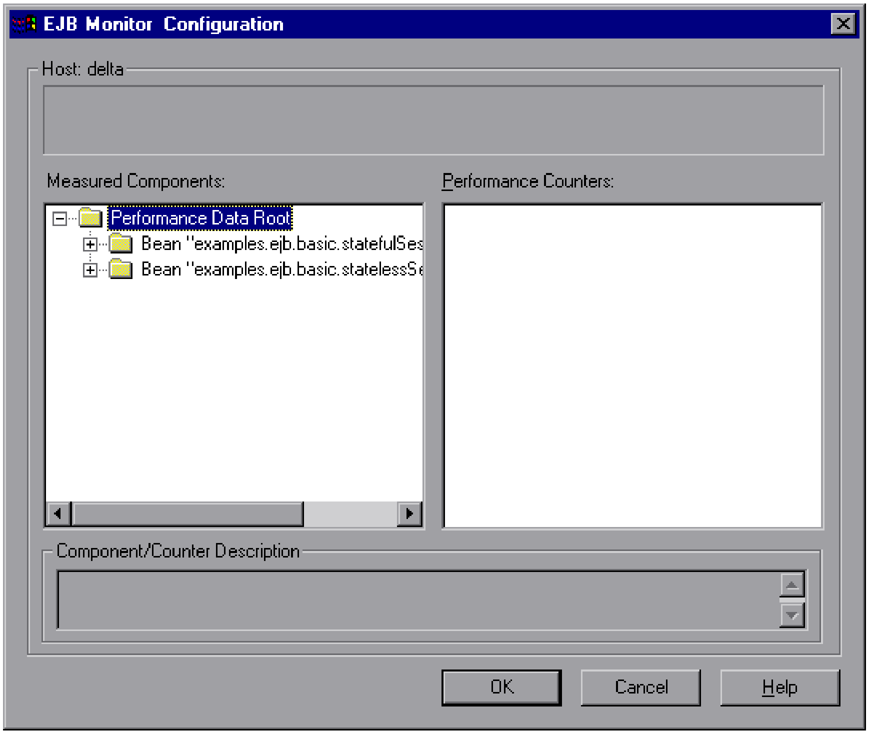
2 Right-click the graph and choose **Add Measurement(s)**, or choose **Monitors** > **Add Online Measurement**. The EJB dialog box opens.



Machine: delta
Platform: WINNT

3 Click **Add** in the Monitored Server Machines box to enter the server name or IP address of the machine you want to monitor. Select the platform on which the machine runs, and click **OK**.

4 Click **Add** in the Resource Measurements section of the EJB dialog box. The EJB Monitor Configuration dialog box opens, displaying the available EJBs.


EJB Monitor Configuration
Host: delta
Measured Components:
Performance Counters:
Performance Data Root
Bean "examples.ejb.basic.statefulSes
Bean "examples.ejb.basic.statelessSe
Component/Counter Description
OK
Cancel
Help

5 Expand the Measured Components tree and select the methods and counters you want to monitor. The following counters can be monitored for each method:

Measurement	Description
Average Response Time	The average response time of the EJB object being monitored.
Method Calls per Second	The number of EJB object method calls per second.

6 Click **OK** in the EJB Monitor Configuration dialog box, and in the EJB dialog box, to activate the EJB monitor.

JProbe Performance Monitoring

You can monitor Java-based applications during a session step run using the JProbe Java profiling tool. In order to monitor Java-based applications, you must first configure the JProbe tool to monitor the specified application. You then use the Console to select the counters you want the JProbe Java profiling tool to measure.

Note: You must run the JPlauncher before opening the Console.

Configuring the JProbe Tool to Monitor an Application

In order to monitor Java-based applications—for example, a WebLogic server—you must first configure the JProbe tool to monitor the specified application.

To configure the JProbe tool:

1. Launch **JProbe Profiler 3.0**.
2. Select **Program** > **Open JProbe Launch Pad**. Select the **Program** tab, and enter the following settings:

   Target Server: <*server name*>—for example, BEA WebLogic 5.1.0

   Server Home Directory: for example, G:\Weblogic

   Working Directory: for example, G:\Weblogic

   Classpath: for example, %CLASSPATH%

3. Select the **VM** (Virtual Machine) tab, and enter the following settings:

   Virtual Machine Type: for example, Java 2

   VM Path: for example, g:\JDK1.2\bin\java.exe

   VM Arguments: for example, -ms64m -mx64m -Dweblogic.home= -Djava.security.manager - Djava.security.policy=. \Weblogic.policy

   Snapshot Directory: for example, G:\TEMP

4 Select the **Attach** tab. In the JProbe Console section, select the option that corresponds to your setup (for example, choose **Local** if the JProbe tool is on the same machine as the Console). In the section relating to the port to be used, select **Use Default Port**.

Note: The Console usually uses port 4444 as the default port for the JProbe tool. To change the default port, edit the *<installation root>\dat\monitors\jprobe.cfg* file, or specify the server machine name in the Console's Add Machine dialog box as host:port.

---

5 Click the **Save As** button, and save the file in JProbe Launch Pad (JPL) format.

6 Close the JProbe Profiler.

7 In DOS, enter cd *<JProbe Profiler Dir>*/jplauncher. At the prompt, type the following:

```
jplauncher -jp_input=< .JPL file>  -jp_socket="<Console machine>:<Port>"
```

The JPlauncher establishes communication with the Console machine. The Console receives data from the JProbe tool through the above default port.

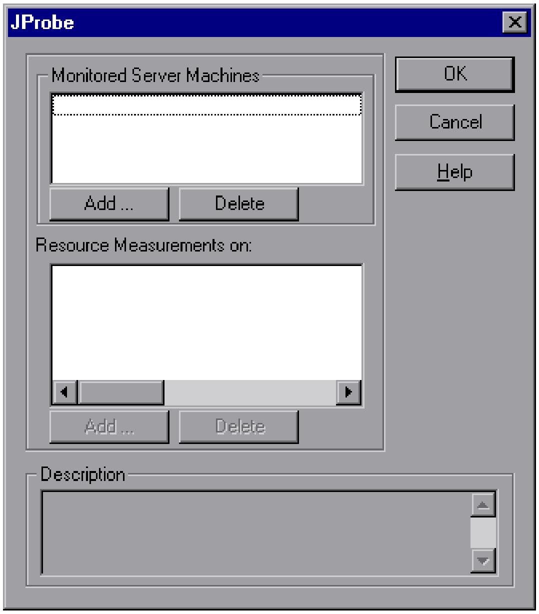
Configuring the JProbe Tool in the Console

To monitor a Java-based application, you must select the counters you want the JProbe tool to measure. You select these counters using the Console's JProbe dialog box.

To configure the JProbe tool:

1 Click the JProbe graph in the graph tree, and drag it into the right pane of the Run view.

2 Right-click the graph in the graph and choose **Add Measurement(s)**, or choose **Monitors** > **Add Online Measurement**. The JProbe dialog box opens.



3 Click **Add** in the Monitored Server Machines box to enter the server name or IP address of the machine you want to monitor. Select the platform on which the machine runs, and click **OK**.

4 Click **Add** in the Resource Measurements section of the JProbe dialog box. The following two measurements appear.

Measurement	Description
Allocated Memory (heap)	The amount of allocated memory in the heap (in bytes)
Available Memory (heap)	The amount of available memory in the heap (in bytes)

5 Click **OK** in the JProbe dialog box to activate the monitor.

Sitraka JMonitor Performance Monitoring

Support Matrix:

Application Server	Version	Platform
WebLogic	6.0; 6.1	Windows; Solaris; AIX
WebSphere	4.0	Windows; Solaris; AIX
Tomcat	3.2.3, 4.0.3	Windows; Solaris; AIX

Installation Options

The Sitraka JMonitor can be configured and run together with the EJB Monitor, or as a standalone monitor.

If you want to install the Sitraka JMonitor with the EJB Monitor, see "Configuring the EJB Monitor and the Sitraka JMonitor on the Application Server," on page 419.

Installing the Sitraka JMonitor on the Application Server

To install the Sitraka JMonitor, create a home directory for the Sitraka JMonitor support files—for example, MERC_MONITOR_HOME—and unzip the installation file *<ProTune_Installation>\Add-ins\J2EE\Sitraka\jmonitor_<platform>.jar* file into that directory.

On UNIX platforms, use the jar utility to extract the installation jar.

Change to the Sitraka JMonitor installation directory and type the following command:

```
jar -xvf <path to your Sitraka JMonitor installation jar>
```

Configuring the Sitraka JMonitor on the Application Server

After you have installed the Sitraka JMonitor support files on your WebLogic, WebSphere, or Tomcat server, you must configure the application server to run with Sitraka JMonitor support.

Note: It is important to set the environment variables in the order in which they appear below.

To configure the WebLogic 6.0/6.1 server:

1. Make a backup copy of the WebLogic startup script.

   For Windows 2000/NT, this will be:

   <WebLogic Home>\config\<domain name>\startWebLogic.cmd

   Name the new file startWebLogicJMonitor.cmd

   For Solaris or AIX, this will be in:

   $BEA_HOME/wlserver6.0/config/<domain name>/startWebLogic.sh

   or *$BEA_HOME/wlserver6.1/config/<domain name>/startWebLogic.sh*

   Name the new file startWebLogicJMonitor.sh

2. Open the *startWebLogicJMonitor.cmd* or *startWebLogicJMonitor.sh* file.

3. In the ‘runWebLogic’ section of the file (or just prior to the call to invoke the JVM), set the following environment variables:

Note: Replace any instances of *<install directory>* with the JMonitor installation directory. Note that on Unix platforms you may have to export the library path variables.

For Windows 2000/NT:

```
set JMONITOR_HOME=<install directory>
```

```
setCLASSPATH=%JMONITOR_HOME%\lib;%JMONITOR_HOME%\lib\
miniwebserver.jar;%JMONITOR_HOME%\lib\jlrutils.jar;%JMONITOR_HOME%\
lib\jmonitor.jar;%CLASSPATH%
```

```
set PATH=%PATH%;%JMONITOR_HOME%\bin\win32_ia32
```

For Solaris:

```
JMONITOR_HOME=<install directory>

CLASSPATH=$JMONITOR_HOME/lib:$JMONITOR_HOME/lib/miniwebserver.jar:$JMONITOR_HOME/lib/jlrutils.jar:$JMONITOR_HOME/lib/jmonitor.jar:$CLASSPATH

LD_LIBRARY_PATH=$JMONITOR_HOME/bin/solaris_sparc:$LD_LIBRARY_PATH

export LD_LIBRARY_PATH
```

For AIX:

```
JMONITOR_HOME=<install directory>

CLASSPATH=$JMONITOR_HOME/lib:$JMONITOR_HOME/lib/miniwebserver.jar:$JMONITOR_HOME/lib/jlrutils.jar:$JMONITOR_HOME/lib/jmonitor.jar:$CLASSPATH

LIBPATH=$JMONITOR_HOME/bin/aix_ppc:$LIBPATH

export LD_LIBRARY_PATH
```

4. In the same section of the file, modify the java command-line.

   For Windows 2000/NT, Solaris, and AIX, add the following parameter:

   ```
   -Xrunjmonitor:load_mws,proxy
   ```

   The command line will look similar to the following (paths shown are for Windows, and are for a specific installation of WebLogic):

   ```
   "%JAVA_HOME%\bin\java" -hotspot -ms64m -mx64m
   -Xrunjmonitor:load_mws,proxy -classpath %CLASSPATH%
   -Dweblogic.Domain=mydomain -Dweblogic.Name=myserver
   "-Dbea.home=f:\bea"
   "-Djava.security.policy==f:\bea\wlserver6.0/lib/weblogic.policy"
   -Dweblogic.management.password=%WLS_PW% weblogic.Server
   ```

5. Run the *startWebLogicJMonitor.cmd* or *startWebLogicJMonitor.sh* file to start the WebLogic instance with JMonitor enabled.

To configure the WebSphere 4.0 server:

1. Ensure that the WebSphere Administrative Server is running, and start the Administrator Console.
2. Expand the WebSphere Administrative Domain tree. Expand the Nodes, hostname, and Application Servers subtrees, and select the Default Server (or the Application Server you wish to use with JMonitor).
3. For Windows 2000/NT and Solaris, click the General tab, and add the following variables to the Environment Editor box:

Note: Replace any instances of <*install directory*> with the JMonitor installation directory.

For Windows 2000/NT:

name=PATH

value=<*install directory*>\bin\win32_ia32

For Solaris:

name=LD_LIBRARY_PATH

value=<*install directory*>/bin/solaris_sparc

Click **OK** to close the Environment Editor.

For AIX:

If the LIBPATH environment variable has been adjusted, you need to link the Sitraka JMonitor to the /usr/lib directory.

Add the following command:

#ln -s <*install directory*>/bin/aix_ppc/libjmonitor.so /usr/lib/libjmonitor.so

Note: you will likely require root permissions in order to create the link. Alternatively, the link can be placed in WebSphere's /bin directory (usually /usr/WebSphere/AppServer/bin).

4 Click the **JVM Settings** tab in the WebSphere Administrative Console, and add the following values to the classpath:

Note: Replace any instances of <*install directory*> with the JMonitor installation directory.

For Windows 2000/NT

<*install directory*>\lib

<*install directory*>\lib\jlrutils.jar

<*install directory*>\lib\miniwebserver.jar

<*install directory*>\lib\jmonitor.jar

For Solaris or AIX:

<*install directory*>\lib

<*install directory*>\lib\jlrutils.jar

<*install directory*>\lib\miniwebserver.jar

<*install directory*>\lib\jmonitor.jar

5 Click the **Advanced JVM Settings** button. In the Command line arguments field, add the following value for Windows 2000/NT, Solaris, and AIX:

```
-Xrunjdkhook:jmonitor:load_mws,proxy
```

6 Click the **OK** and **Apply** buttons to save the changes for the Application server. You can now start and stop your WebSphere server using the Sitraka JMonitor.

7 To start your application server without using the Sitraka JMonitor, remove the changes made in step 5.

To configure the Tomcat 3.2.3 server:

1 Make a backup copy of the Tomcat startup script.

For Windows 2000/NT, this will be:

<Tomcat Home>\bin\tomcat.bat

Name the new file tomcat_jmonitor.bat

For Solaris or AIX, this will be in:

<Tomcat home>/bin/tomcat.sh

Name the new file tomcat_jmonitor.sh

2 Open the *tomcat_jmonitor.bat* or *tomcat_jmonitor.sh* file.

3 Set the following environment variables:

For Windows 2000/NT, the following additions should be added prior to the line invoking the JVM (in the 'startServer' section of the batch file):

```
set JMONITOR_HOME=<install directory>

set CLASSPATH=%JMONITOR_HOME%\lib;%JMONITOR_HOME%\lib\
miniwebserver.jar;%JMONITOR_HOME%\lib\jlrutils.jar;%JMONITOR_HOME%\
lib\jmonitor.jar;%CLASSPATH%

set PATH=%PATH%;%JMONITOR_HOME%\bin\win32_ia32
```

For Solaris, the following lines should be added prior to invoking the JVM (just after the first export CLASSPATH found in the file):

```
JMONITOR_HOME=<install directory>

CLASSPATH=$JMONITOR_HOME/lib:$JMONITOR_HOME/lib/miniwebserver.
jar:$JMONITOR_HOME/lib/jlrutils.jar:$JMONITOR_HOME/lib/jmonitor.jar:
$CLASSPATH

export CLASSPATH

LD_LIBRARY_PATH=$JMONITOR_HOME/bin/solaris_sparc:$LD_LIBRARY_
PATH

export LD_LIBRARY_PATH
```

For AIX, the following lines should be added prior to invoking the JVM (just after the first export CLASSPATH found in the file):

```
JMONITOR_HOME=<install directory>

CLASSPATH=$JMONITOR_HOME/lib:$JMONITOR_HOME/lib/miniwebserver.j
ar:$JMONITOR_HOME/lib/jlrutils.jar:$JMONITOR_HOME/lib/jmonitor.jar:$CLA
SSPATH

export CLASSPATH

LIBPATH=$JMONITOR_HOME/bin/aix_ppc:$LD_LIBRARY_PATH

export LIBPATH
```

4. In the same section of the file, modify the java command-line.

   For Windows 2000/NT, Solaris, and AIX, add the following parameter:

   ```
   -Xrunjmonitor:load_mws,proxy
   ```

5. To start the Tomcat server with JMonitor, use the following command: tomcat_jmonitor start.

To configure the Tomcat 4.0.3 server:

1 Make a backup copy of the Tomcat startup script.

For Windows 2000/NT, this will be:

<Catalina Home>\bin\catalina.bat

Name the new file catalina_jmonitor.bat

For Solaris or AIX, this will be in:

<Tomcat home>/bin/catalina.sh

Name the new file catalina_jmonitor.sh

2 Open the *tomcat_jmonitor.bat* or *tomcat_jmonitor.sh file*.

3 Set the following environment variables:

For Windows 2000/NT, the following additions should be added prior to the line invoking the JVM (in the ‘doStart’ section of the batch file):

```
set JMONITOR_HOME=<install directory>

set CLASSPATH=%JMONITOR_HOME%\lib;%JMONITOR_HOME%\lib\
miniwebserver.jar;%JMONITOR_HOME%\lib\jlrutils.jar;%JMONITOR_
HOME%\lib\jmonitor.jar;%CLASSPATH%

set PATH=%PATH%;%JMONITOR_HOME%\bin\win32_ia32
```

For Solaris, the following lines should be added prior to invoking the JVM (at the beginning of the ‘Execute The Requested Command’ section):

```
JMONITOR_HOME=<install directory>

CLASSPATH=$JMONITOR_HOME/lib:$JMONITOR_HOME/lib/miniwebserver.
jar:$JMONITOR_HOME/lib/jlrutils.jar:$JMONITOR_HOME/lib/jmonitor.jar:
$CLASSPATH

export CLASSPATH

LD_LIBRARY_PATH=$JMONITOR_HOME/bin/solaris_sparc:$LD_LIBRARY_
PATH

export LD_LIBRARY_PATH
```

For AIX, the following lines should be added prior to invoking the JVM (at the beginning ofthe 'Execute The Requested Command' section):

```
JMONITOR_HOME=<install directory>

CLASSPATH=$JMONITOR_HOME/lib:$JMONITOR_HOME/lib/miniwebserver.jar:$JMONITOR_HOME/lib/jlrutils.jar:$JMONITOR_HOME/lib/jmonitor.jar:$CLASSPATH

export CLASSPATH

LIBPATH=$JMONITOR_HOME/bin/aix_ppc:$LD_LIBRARY_PATH

export LIBPATH
```

4 In the same section of the file, modify the java command-line.

For Windows 2000/NT, Solaris, and AIX, add the following parameter:

```
-Xrunjmonitor:load_mws,proxy
```

For Windows, the Java command line is found in the 'doneSetArgs' section. For Solaris and AIX, the Java call to start Tomcat is in the 'Execute The Requested Command' section.

5 To start the Tomcat server with JMonitor, use the following command: catalina_jmonitor start

To stop the Tomcat server with JMonitor, use the regular *catalina.bat* or *catalina.sh* file.

Configuring the EJB Monitor and the Sitraka JMonitor on the Application Server

If you want to monitor EJB objects using the Sitraka JMonitor and the EJB Monitor together, you must first install the EJB Monitor and run the monitor detector. For more information, see "Installing the EJB Monitor and Running the Monitor Detector," on page 392.

Installing the Sitraka JMonitor on the Application Server

To install the Sitraka JMonitor, create a home directory for the Sitraka JMonitor support files—for example, MERC_MONITOR_HOME—and unzip the installation file *<ProTune_Installation>\Add-ins\J2EE\Sitraka\jmonitor_<platform>.jar* file into that directory.

On UNIX platforms, use the jar utility to extract the installation jar.

Change to the Sitraka JMonitor installation directory and type the following command:

```
jar -xvf <path to your Sitraka JMonitor installation jar>
```

Support Matrix:

Application Server	Version	Platform
WebLogic	6.0; 6.1	Windows; Solaris; AIX
WebSphere	4.0	Windows; Solaris; AIX

After you have installed the EJB and Sitraka JMonitor support files on your WebLogic or WebSphere machine, you must configure the application server to run with EJB Monitor and Sitraka JMonitor support.

Note: It is important to set the environment variables in the order in which they appear below.

To configure the WebLogic 6.0/6.1 server:

1. Make a backup copy of the <*WebLogic Home*>*config*\<*domain name*>*startWeblogic.cmd* file.
2. Open the <*WebLogic Home*>*config*\<*domain name*>*startWeblogicMercury.cmd* file.
3. In the 'runWebLogic' section of the file, just before the Java command line used to start the server, set the following environment variables:

Note: For IBM AIX platform replace LD_LIBRARY_PATH with LIBPATH. Replace <*EJB Monitor Home Directory*> with the EJB monitor installation root directory. Replace <*Sitraka JMonitor Home Directory*> with the Sitraka JMonitor installation root directory. On UNIX platforms you may have to export the library path variables.

For Windows platforms:

```
set MERC_MONITOR_HOME=<EJB Monitor Home Directory>

set CLASSPATH=%MERC_MONITOR_HOME%\dat;%MERC_MONITOR_
HOME%\classes;%MERC_MONITOR_HOME%\classes\xerces.jar;
%CLASSPATH%

set PATH=%PATH%;%MERC_MONITOR_HOME%\bin

set JMONITOR_HOME=<Sitraka Jmonitor Home Directory>

set CLASSPATH=%JMONITOR_HOME%\lib;%JMONITOR_HOME%\lib\
miniwebserver.jar;%JMONITOR_HOME%\lib\jlrutils.jar;%JMONITOR_HOME%\
lib\jmonitor.jar;%CLASSPATH%

set PATH=%PATH%;%JMONITOR_HOME%\bin\win32_ia32
```

For UNIX platforms:

```
MERC_MONITOR_HOME <EJB Monitor Home Directory>

CLASSPATH${MERC_MONITOR_HOME}/
dat:${MERC_MONITOR_HOME}/classes:${MERC_MONITOR_HOME}/classes/
xerces.jar:${CLASSPATH}
```

```
LD_LIBRARY_PATH${LD_LIBRARY_PATH}:${MERC_MONITOR_HOME}/bin

JMONITOR_HOME <Sitraka Jmonitor Home Directory>

CLASSPATH${JMONITOR_HOME}/lib:${JMONITOR_HOME}/lib/
miniwebserver.jar:${JMONITOR_HOME}/lib/jlrutils.jar:${JMONITOR_HOME}/lib/
jmonitor.jar:${CLASSPATH}

setenv LD_LIBRARY_PATH${LD_LIBRARY_PATH}:${JMONITOR_HOME}/bin/
win32_ia32
```

4 In the same section of the file, add the following parameter:

```
-Xrunjdkhook:jmonitor:load_mws,proxy
```

For Windows platforms:

```
%JAVA_HOME%\bin\java -hotspot -ms64m -mx64m -
Xrunjdkhook:jmonitor:load_mws,proxy -classpath
%CLASSPATH%
-Dweblogic.Domain=mydomain
-Dweblogic.Name=myserver
"-Dbea.home=f:\bea"
"-Djava.security.policy==f:\bea\wlserver6.0/lib/weblogic.policy"
-Dweblogic.management.password=%WLS_PW% weblogic.Server
```

For UNIX platforms:

```
${JAVA_HOME}/bin/java -hotspot -ms64m -mx64m
-Xrunjdkhook:jmonitor:load_mws,proxy -classpath ${CLASSPATH}
-Dweblogic.Domain=mydomain -Dweblogic.Name=myserver "
-Dbea.home=usr/bea"
"-Djava.security.policy==usr/bea/wlserver6.0/lib/weblogic.policy"
-Dweblogic.management.password=${WLS_PW} weblogic.Server
```

5 Run the *<WebLogic Home>\config\<domain name>\startWeblogicMercury.cmd* file to start the server with JMonitor and ProTune EJB monitor support.

To configure the WebSphere 4.0 server:

1 Ensure that the WebSphere Administrative Server is running, and start the Administrator Console.

2 In the WebSphere Administrative Domain tree, expand the Nodes, hostname, and Application Servers subtrees, and select the Default Server (or the Application Server you wish to use with JMonitor).

3 For Windows 2000/NT and Solaris, click the General tab, and add the following variables to the Environment Editor box:

Note: Replace <*EJB Monitor Home Directory*> with the EJB monitor installation root directory, and <*Sitraka Jmonitor Home Directory*> with the JMonitor installation root directory.

For Windows 2000/NT:

name=PATH

value=<*EJB Monitor Home Directory*>\bin;<*Sitraka Jmonitor Home Directory*>\bin\win32_ia32

For Solaris:

name=LD_LIBRARY_PATH

value=<*EJB Monitor Home Directory*>/bin:<*Sitraka Jmonitor Home Directory*>/bin/solaris_sparc

Click **OK** to close the Environment Editor.

For AIX:

If the LIBPATH environment variable has been changed, you need to link the Sitraka JMonitor and the EJB monitor libraries in the /usr/lib directory.

Add the following command:

#ln -s <*Sitraka Jmonitor Home Directory*>/bin/aix_ppc/libjmonitor.so /usr/lib/libjmonitor.so

#ln -s <*EJB Monitor Home Directory*>/bin/libcjhook_mon.so /usr/lib/libcjhook_mon.so

#ln -s <*EJB Monitor Home Directory*>/bin/libconfig.so /usr/lib/libconfig.so

#ln -s <*EJB Monitor Home Directory*>/bin/libjdkhook.so /usr/lib/libjdkhook.so

#ln -s <*EJB Monitor Home Directory*>/bin/libmlib_ds.so/usr/lib/libcjhook_mon.so

#ln -s <*EJB Monitor Home Directory*>/bin/libmosifs.so /usr/lib/libmosifs.so

#ln -s <*EJB Monitor Home Directory*>/bin/libthrdutil.so/usr/lib/libthrdutil.so

---

Note: You will likely require root permissions in order to create the link. Alternatively, you can place the link in WebSphere's /bin directory (usually /usr/WebSphere/AppServer/bin).

---

4 Click the **JVM Settings** tab in the WebSphere Administrative Console, and add the following values to the classpath:

Note: Replace <*EJB Monitor Home Directory*> with the EJB monitor installation root directory, and <*Sitraka Jmonitor Home Directory*> with the JMonitor installation root directory.

For Windows 2000/NT:

<*EJB Monitor Home Directory*>\dat

<*EJB Monitor Home Directory*>\classes

<*EJB Monitor Home Directory*>\classes\xerces.jar

<*Sitraka Jmonitor Home Directory*>\lib

<*Sitraka Jmonitor Home Directory*>\lib\jlrutils.jar

<*Sitraka Jmonitor Home Directory*>\lib\miniwebserver.jar

<*Sitraka Jmonitor Home Directory*>\lib\jmonitor.jar

For Solaris or AIX:

<*EJB Monitor Home Directory*>/dat

<*EJB Monitor Home Directory*>/classes

<*EJB Monitor Home Directory*>/classes/xerces.jar

<*Sitraka Jmonitor Home Directory*>/lib

<*Sitraka Jmonitor Home Directory*>/lib/jlrutils.jar

<*Sitraka Jmonitor Home Directory*>/lib/miniwebserver.jar

<*Sitraka Jmonitor Home Directory*>/lib/jmonitor.jar

5 Click the **Advanced JVM Settings** button. In the Command line arguments field, add the following value for Windows 2000/NT, Solaris, and AIX:

```
-Xrunjdkhook:jmonitor:load_mws,proxy
```

6 Click the **OK** and **Apply** buttons to save the changes for the Application server. You can now start and stop your WebSphere server using the Sitraka JMonitor.

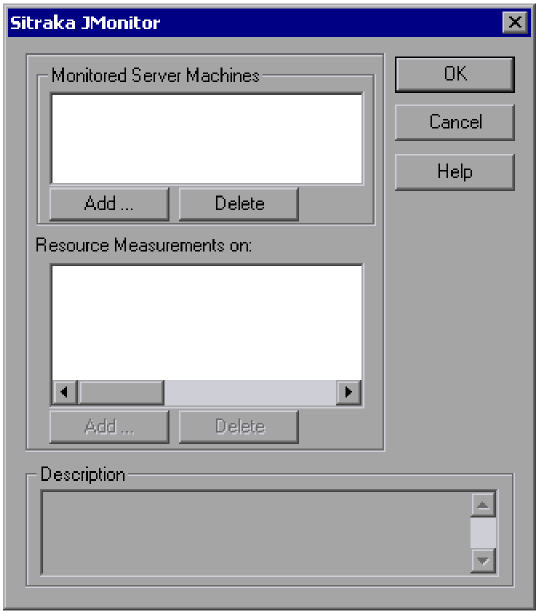
Configuring the Sitraka JMonitor in the Console

To monitor a Java-based application, you must select the counters you want the Sitraka JMonitor to measure. You select these counters using the Console's Sitraka JMonitor dialog box.

To configure the Sitraka JMonitor monitor:

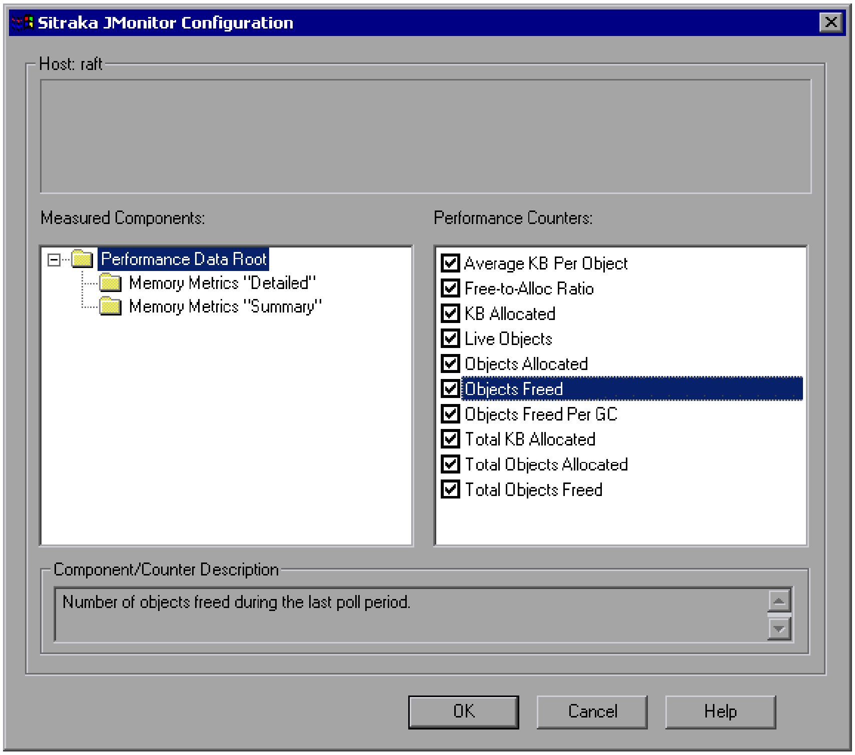
1 Click the Sitraka JMonitor graph in the graph tree, and drag it into the right pane of the Run view.

2 Right-click the graph in the graph and choose **Add Measurement(s)**, or choose **Monitors** > **Add Online Measurement**. The Sitraka JMonitor dialog box opens.



3 Click **Add** in the Monitored Server Machines box to enter the server name or IP address of the machine you want to monitor. Select the platform on which the machine runs, and click **OK**.

4 Click **Add** in the Resource Measurements section of the Sitraka JMonitor dialog box. The Sitraka JMonitor dialog box opens, displaying the available counters.



Number of objects freed during the last poll period.

5 Expand the Measured Components tree and select the methods and counters you want to monitor.

The following counters are available for the Sitraka JMonitor:

SummaryMemoryMetrics

Measurement	Description
% Free Heap Space	The percentage of free heap space since the last report.
% GC-To-Elapsed Time	The percentage of garbage collection to elapsed time.
% GC-To-Poll Time	The percentage of garbage collection to poll time.
% Used Heap	The percentage of used heap space since the last report.
Average GC Time (ms)	The average time, in milliseconds, spent performing garbage collections since the metric was enabled. (Disabling metric resets value to zero).
GC Time (ms)	The time, in milliseconds, spent performing garbage collections during the last poll period.
Heap Size (KB)	Total heap size, in kilobytes.
KB Freed	The number of kilobytes freed in the last poll period.
KB Freed Per GC	Average number of kilobytes freed per garbage collection since the metric was enabled (Disabling the metric resets value to zero).
Number of GCs	The number of garbage collections during the last poll period.
Total GC Time (ms)	The total time, in milliseconds, spent performing garbage collections since the metric was enabled. (Disabling the metric resets the value to zero).
Total GCs	The total number of garbage collections since the metric was enabled. (Disabling the metric resets value to zero).

Measurement	Description
Total KB Freed	The total number of kilobytes freed since the metric was enabled. (Disabling the metric resets value to zero).
Used Heap (KB)	Used heap size, in kilobytes.

DetailedMemoryMetrics

Measurement	Description
Average KB Per Object	The average number of kilobytes per object since the metric was enabled. (Disabling the metric resets value to zero).
Free-to-Alloc Ratio	The objects freed to objects allocated ratio since the metric was enabled (Disabling the metric resets value to zero).
KB Allocated	The number of kilobytes allocated since the metric was enabled. (Disabling the metric resets value to zero).
Live Objects	The change in number of live objects during the last poll period.
Objects Allocated	The number of objects allocated in the last poll period.
Objects Freed	The number of objects freed during the last poll period.
Objects Freed Per GC	The average number of objects freed per garbage collection since the metric was enabled. (Disabling the metric resets value to zero).
Total KB Allocated	The kilobytes allocated since metric was enabled. (Disabling the metric resets value to zero).

Measurement	Description
Total Objects Allocated	The number of objects allocated since the metric was enabled. (Disabling the metric resets value to zero).
Total Objects Freed	The number of objects freed since the metric was enabled. (Disabling the metric resets value to zero).

6 Click **OK** in the Sitraka JMonitor Configuration dialog box, and in the Sitraka JMonitor dialog box, to activate the Sitraka JMonitor monitor.

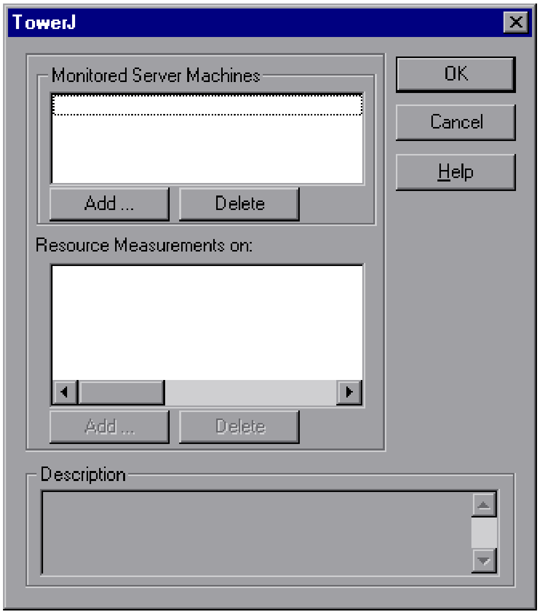
TowerJ Performance Monitoring

To monitor the TowerJ Java virtual machine, you must select the counters you want the TowerJ monitor to measure. You select these counters using the TowerJ dialog box.

To configure the TowerJ monitor:

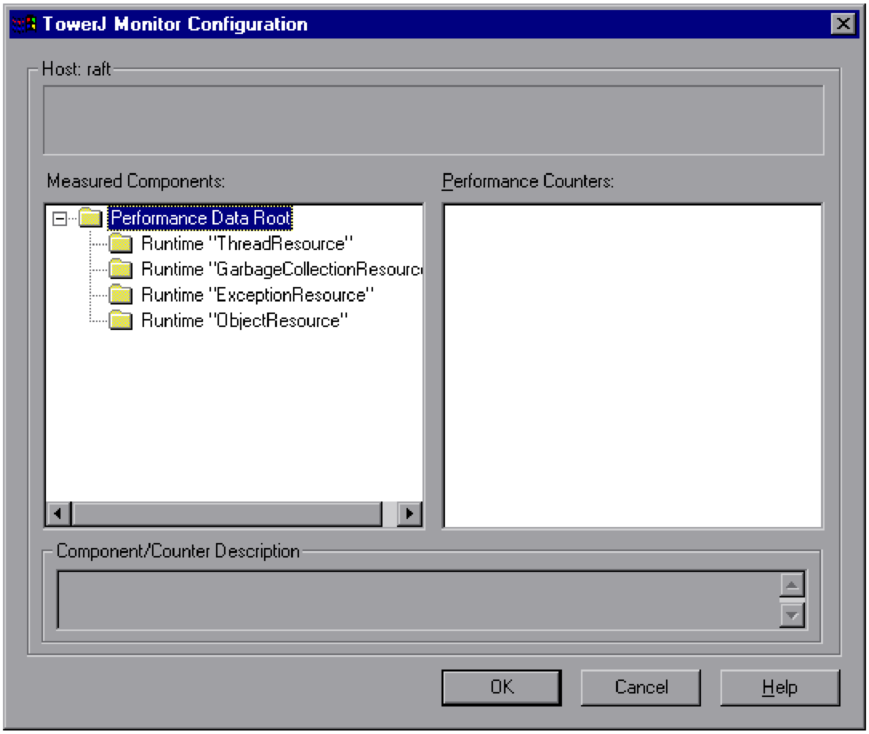
1 Click the TowerJ graph in the graph tree, and drag it into the right pane of the Run view.

2 Right-click the graph and choose **Add Measurement(s)**, or choose **Monitors** > **Add Online Measurement**. The TowerJ dialog box opens.



3 Click **Add** in the Monitored Server Machines box to enter the server name or IP address of the machine you want to monitor. Select the platform on which the machine runs, and click **OK**.

4 Click **Add** in the Resource Measurements section of the TowerJ dialog box. The TowerJ dialog box opens, displaying the available counters.



5 Expand the Measured Components tree and select the methods and counters you want to monitor. The following counters are available for the TowerJ Java virtual machine:

ThreadResource Measurement	Description
ThreadStartCountTotal	The number of threads that were started.
ThreadStartCountDelta	The number of threads that were started since the last report.
ThreadStopCountTotal	The number of threads that were stopped.
ThreadStopCountDelta	The number of threads that were stopped since the last report

GarbageCollection Resource Measurement	Description
GarbageCollectionCount Total	The number of times the garbage collector has run.
GarbageCollectionCount Delta	The number of times the garbage collector has run since the last report.
PreGCHeapSizeTotal	The total pre-GC heap space.
PreGCHeapSizeDelta	The total pre-GC heap space since the last report
PostGCHeapSizeTotal	The total post-GC heap space.
PostGCHeapSizeDelta	The total post-GC heap space since the last report.
NumPoolsTotal	The number of pools.
NumPoolsDelta	The number of pools since the last report.
NumSoBlocksTotal	The number of small object blocks.
NumSoBlocksDelta	The number of small object blocks since the last report.
NumLoBlocksTotal	The number of large object blocks.
NumLoBlocksDelta	The number of large object blocks.
NumFullSoBlocksTotal	The number of full small object blocks.
NumFullSoBlocksDelta	The number of full small object blocks since the last report.
TotalMemoryTotal	Total memory (heap size).
TotalMemoryDelta	Total memory (heap size) since the last report.
NumMallocsTotal	The number of current mallocs.
NumMallocsDelta	The number of current mallocs since the last report.
NumBytesTotal	The number of current bytes allocated.
NumBytesDelta	The number of current bytes allocated since the last report.

GarbageCollection Resource Measurement	Description
TotalMallocsTotal	The total number of mallocs.
TotalMallocsDelta	The total number of mallocs since the last report.
TotalBytesTotal	The total number of bytes allocated.
TotalBytesDelta	The total number of bytes allocated since the last report.

ExceptionResource Measurement	Description
ExceptionCountTotal	The number of exceptions thrown.
ExceptionCountDelta	The number of exceptions thrown since the last report.

ObjectResource Measurement	Description
NumberOfObjectsTotal	The number of objects.
NumberOfObjectsDelta	The number of objects since the last report.

6 Click **OK** in the TowerJ Monitor Configuration dialog box, and in the TowerJ dialog box, to activate the TowerJ monitor.

26

Troubleshooting Online Monitors

ProTune monitors allow you to view the performance of the session step during execution.

The following sections describe several tips and known issues relating to the online monitors.

- Troubleshooting Server Resource Monitors
- Troubleshooting the Network Delay Monitor
- Network Considerations

Troubleshooting Server Resource Monitors

In order to monitor resources on a server machine, you must be able to connect to that machine. If monitoring is unsuccessful and ProTune cannot locate the specified server, make sure that the specified server is available. Perform a "ping" operation by typing ping <*server_name*> from the Console machine command line.

Once you verify that the machine is accessible, check this table for additional tips on troubleshooting the monitor.

Problem	Solution
Cannot monitor a Windows machine on a different domain, or "access denied."	To gain administrative privileges to the remote machine, perform the following from the command prompt: %net use \\<*MachineName*>/ user:[<*Domain*>\<*RemoteMachineUsername*>] At the password prompt, enter the password for the remote machine.

Problem	Solution
Cannot monitor an NT/Win 2000 machine (An error message is issued: “computer_name not found” or “Cannot connect to the host”)	The NT/Win 2000 machine you want to monitor only enables monitoring for users with administrator privileges. In order to allow monitoring for non-admin users, you must grant read permission to certain files and registry entries (Microsoft tech-note number Q158438.) The required steps are: **a.** Using Explorer or File Manager, give the user READ access to: %windir%\system32\PERFCxxx.DAT %windir%\system32\PERFHxxx.DAT where *xxx* is the basic language ID for the system—for example, 009 for English. These files may be missing or corrupt. If you suspect this; expand these files off of the installation cd. **b.** Using REGEDT32, give the user READ access to: HKEY_LOCAL_MACHINE\Software\Microsoft\Windows NT\CurrentVersion\Perflib and all sub keys of that key. **c.** Using REGEDT32, give the user at least READ access to: HKEY_LOCAL_MACHINE\System\CurrentControlSet\Control\SecurePipeServers\winreg
Some Win 2000 counters cannot be monitored from an NT machine.	Run the Console on a Win 2000 machine.
Some Windows default counters are generating errors	Remove the problematic counters and add the appropriate ones using the “Add Measurement” dialog box.
You cannot get performance counters for the SQL server (version 6.5) on the monitored machine.	There is a bug in SQL server version 6.5. As a workaround, give read permission to the following registry key at the monitored machine (use regedt32): HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\MSSQLServer\MSSQLServer (Microsoft tech-note number Q170394)

Problem	**Solution**
The selected measurements are not displayed in the graph.	Ensure that the display file and online.exe are registered. To register the monitor dll's, without performing a full installation, run the *set_mon.bat* batch file located in ProTune/bin.
When monitoring a Windows machine, no measurements appear in the graph.	Check the built-in Windows Performance Monitor. If it is not functional, there may be a problem with the communication setup.
When monitoring a UNIX machine, no measurements appear in the graph.	Ensure that an *rstatd* is running on the UNIX machine (Refer to Chapter 17, "System Resource Monitoring.").
Cannot monitor one of the following Web servers: MS IIS, MS ASP, or ColdFusion	Refer to problem above, "Cannot monitor a Windows machine."
Cannot monitor the WebLogic (JMX) server	Open the *ProTune*root folder\dat\monitors\WebLogicMon.ini file, and search for: [WebLogicMonitor] JVM=javaw.exe Change javaw.exe to java.exe. A window containing trace information opens.

Troubleshooting the Network Delay Monitor

If monitoring is unsuccessful and ProTune cannot locate the source or destination machines, make sure that the specified machines are available to your machine. Perform a "ping" operation. At the command line prompt, type:

```
ping server_name
```

To check the entire network path, use the trace route utility to verify that the path is valid.

For Windows, type tracert <*server_name*>*.*

For UNIX, type traceroute <*server_name*>.

If the monitoring problem persists once you verify that the machines are accessible and that the network path is valid, perform the following procedures:

1) If you are using the TCP protocol, run *<ProTune root folder\bin\webtrace.exe* from the source machine to determine whether the problem is related to the Console, or the WebTrace technology on which the Network Delay monitor is based. If you are using the UDP or ICMP protocols, the problem must be related to the Console and not WebTrace, since these protocols are not WebTrace technology-based.

2) If you receive results by running *webtrace.exe*, the problem is related to the Console. Verify that the source machine is not a UNIX machine, and contact Mercury Interactive's Customer Support with the following information:

- the Console log file, *drv_log.txt*, located in the temp directory of the Console machine.
- the *traceroute_server* log file, located on the source machine.
- the debug information located in the *TRS_debug.txt* and *WT_debug.txt* files in the path directory. These files are generated by adding the following line to the [monitors_server] section of the *<ProTune root folder\dat\mdrv.dat* file, and rerunning the Network monitor:

```
ExtCmdLine=-traceroute_debug path
```

3) If you do not receive results by running *webtrace.exe*, the problem is related to the WebTrace technology, on which the Network Delay monitor is based. Perform the following procedures on the source machine:

- Verify that the *packet.sys* file (the Webtrace driver) exists in the WINNT\system32\drivers directory.
- Check whether a driver (such as “Cloud” or “Sniffer”) is installed on top of the network card driver. If so, remove it and run WebTrace again.
- Verify that there are administrator permissions on the machine.
- Using ipconfig /all, check that only one IP address is assigned to the network card. WebTrace does not know how to handle multiple IP addresses assigned to the same card (IP spoofing).
- Check the number of network cards installed. Run webtrace –devlist to receive a list of the available network cards.
- If there is more than one card on the list, run webtrace -dev <dev_name> <destination>, where <dev_name> is one of the network card names shown in the list. If you discover that WebTrace is binding to the wrong card, you can use webtrace set_device <dev_name> in order to set a registry key that instructs WebTrace to use a specified card instead of the default one.
- Verify that the network card is of the Ethernet type.
- Contact Mercury Interactive’s Customer Support with the output of webtrace.exe –debug (for example, webtrace.exe –debug www.merc-int.com) and ipconfig /all on the machine.

Network Considerations

If you notice extraordinary delays on the network, refer to one of the following sections to increase the performance:

- Network Bandwidth Utilization
- Ethernet-bus Based Networks

➤ Working on a WAN or Heavily Loaded LAN

Network Bandwidth Utilization

In most load-testing session steps, the network card has little impact on session step performance. Network cards are manufactured to handle the bandwidth of the physical network layer. Packets are transferred over an Ethernet at a rate that complies with IEEE 803.x standards. If the network becomes a bottleneck, the issue is not the brand of the network card, but rather the bandwidth limitations on the physical layer (--i.e. Ethernet, FDDI, ATM, Ethernet Token-ring, etc.).

That is, instead of load testing over a T10 line, upgrade your line to DS3 (45Mbps), or T100 (100Mbps).

Below are a few tips that will help qualify the need to upgrade the network:

1) Run the performance monitor on the Vuser load generators. As the number of Vusers increases, check the network byte transfer rate for saturation. If a saturation point has been reached, do not run any more Vusers without upgrading the network—otherwise performance of Vusers will degrade. Degradation is exponential in networking environments.

2) Run the performance monitor on the server machine. Run many Vusers on several load generator machines. Check the kernel usage and network transfer rate for saturation. If saturation is reached with less than the desired Vuser load, upgrade the network.

3) Every network has a different Maximum Transmission Unit or MTU, which is set by the network administrator. The MTU is the largest physical packet size (in bytes) that a network can transmit. If a message is larger than the MTU, it is divided into smaller packets before being sent.

If clients and servers are passing large data sets back and forth, instruct the network administrator to increase the MTU in order to yield better bandwidth utilization. Ideally, you want the MTU to be the same as the smallest MTU of all the networks between your machine and a message's final destination.

If you send a message that is larger than one of the MTUs, it will be broken up into fragments, slowing transmission speeds. If the MTU is too high, it may cause unintended degradation. Trial and error is the only sure way of finding the optimal MTU, but there are some guidelines that can help. For example, most Ethernet networks have an MTU of 1500.

If the desired MTU reduces performance, upgrade the network or reduce the MTU to improve performance.

Ethernet-bus Based Networks

The following guidelines apply to Ethernet-bus based networks:

Networks with only 2 active machines communicating yield a maximum of 90% bandwidth utilization.

Networks with 3 active machines communicating yield a maximum of approximately 85% bandwidth utilization.

As the number of active machines on the network increases, the total bandwidth utilization decreases.

Working on a WAN or Heavily Loaded LAN

When you work with ProTune on a WAN or heavy loaded LAN, you may notice some unusual ProTune behavior, which indicates network problems. The Output window may contain messages about retries, lost packets, or message mismatch. This is because some of the messages from the Console may not be reaching the ProTune agent. To solve this problem, you should reduce the network traffic or improve the network bandwidth.

The following steps may help reduce network traffic:

➤ Click the **Run-Time Settings** button. In the Log tab, select **Disable logging**.

➤ Initialize all users before running them. Run them only after initialization is completed.

Part VI

Tuning Your System

27

Tuning Your System from the Console

Once you've run session steps on your topology and analyzed the test results, you use the Console to administer and tune your hosts and services from a remote location. You continue this process until your system reaches optimal performance.

This chapter includes the following topics:

- Tuning Flow
- Configuring Host Connection Parameters
- Installing and Starting a Tuning Agent
- Viewing the Host Information
- Changing Tuning Parameter Values
- Updating the Host or Service with Changes

About Tuning Your System from the Console

ProTune's tuning features allow you to tune hosts and services remotely, from the Console machine. After ensuring that you have the necessary permissions and access rights on the host, you install a tuning agent—a small application—on the host. The tuning agent allows the Console to view the host's services, and to configure their settings.

ProTune's **Tune** tab contains the following:

- **Server Configuration**: A tree structure listing the hosts and services you can tune.

- **Information tab**: Describes the properties of the hosts and services you select in the Server Configuration tree.
- **Tuning tab**: Allows you to change the configurable properties of the hosts and services.
- Additional buttons that help you tune your system.
- Links to help topics that explain and guide you through the performance tuning process.

Tuning Flow

You tune each host in your system by following this procedure:

1. Configuring Host Connection Parameters
2. Installing and Starting a Tuning Agent
3. Viewing the Host Information
4. Changing Tuning Parameter Values
5. Updating the Host or Service with Changes

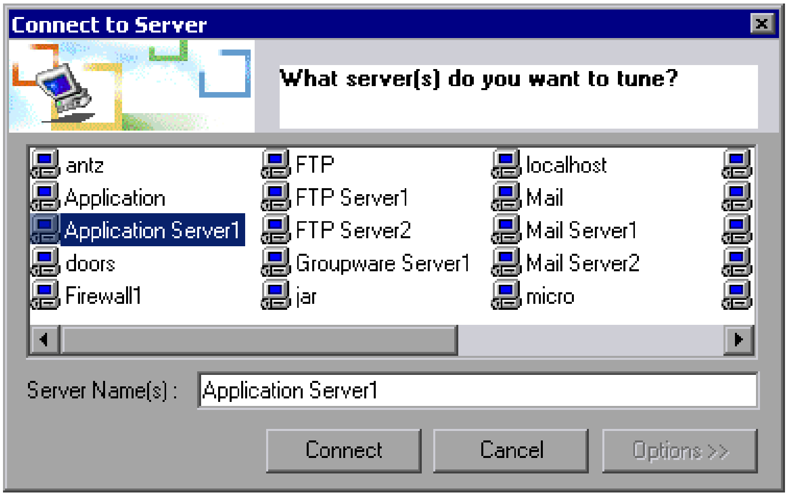
Configuring Host Connection Parameters

This involves choosing or specifying the host that you want to tune, and configuring its connection parameters.

To choose a host:

1. Click the **Connect to Host** button, or right-click the **Server Configurations** link on the left side of the window and click **Add New**.

The Connect to Server dialog box is displayed.



Connect to Server

What server(s) do you want to tune?

antz
Application
Application Server1
doors
Firewall1
FTP
FTP Server1
FTP Server2
Groupware Server1
jar
localhost
Mail
Mail Server1
Mail Server2
micro

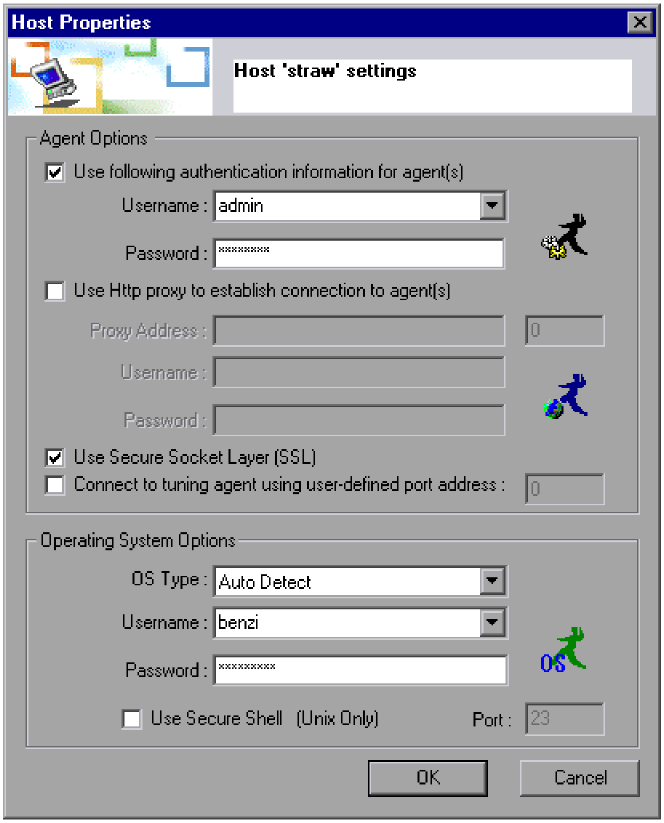
Server Name(s) : Application Server1

2 Select the host that you want to tune, and click **Connect**. An icon representing the host appears in the Server Configuration tree.

Note: Since the new host does not yet have a tuning agent running on it, the host's icon is red, and in the window's right pane ProTune displays a message stating that no system information is available about the host. ProTune also displays an additional message stating that it is unable to establish a connection to the remote service. After you install a tuning agent, the icon will change to green and the host information will be displayed. For details of how to install a tuning agent, see "Installing and Starting a Tuning Agent," on page 450.

To configure the connection parameters:

1 Click **View Host Properties**, or right-click the host's icon and choose Properties. The Host Properties dialog box is displayed.



Host Properties

Host 'straw' settings

Agent Options

☑ Use following authentication information for agent(s)

Username : admin

Password : *********

☐ Use Http proxy to establish connection to agent(s)

Proxy Address : 0

Username :

Password :

☑ Use Secure Socket Layer (SSL)

☐ Connect to tuning agent using user-defined port address : 0

Operating System Options

OS Type : Auto Detect

Username : benzi

Password : **********

☐ Use Secure Shell (Unix Only) Port : 23

2 Enter the tuner username and password. This determines the actions that the tuner allows you to perform on the host. You can select a username and

password defined by the system administrator, or choose one of the following predefined username/password pairs:

Username	Password	Authorization	Comments
guest	(no password)	viewing access only	default
mercury	expert	viewing and update access	
admin	changeit	full administration access	The administrator of the tuner can define and grant tuning privileges to a user.

If you do not check the **Use following authentication information...** box, ProTune uses the *guest* username, allowing you only to query the host.

3 If the tuning agent on the host is accessible only through a proxy over a firewall, define the proxy settings in the **Use Http proxy...** dialog box.

4 If the agent is running (or will run) on a non-standard port (OTP-SSL 4863 or OTP 4862), you can specify a user-defined port. Check the **Connect to tuning agent** box and enter the port address in the relevant field.

5 Check the **Use Secure Socket Layer (SSL)** box to use SSL for all connections (recommended).

6 In the Operating System Options section, enter the host's username and password in the appropriate fields.

7 To use secure shell (for UNIX only), check the **Use Secure Shell** box and enter the port address in the Port field.

8 Click OK to close the Host Properties dialog box and save your settings.

Installing and Starting a Tuning Agent

To enable the Console to tune a host machine, you install a tuning agent on the host, and start the agent. For details about tuning agents, see "Configuring Tuning Agents," on page 503.

You can install the tuning agent remotely from the Console machine or locally on the host.

Host Requirements for a Tuning Agent

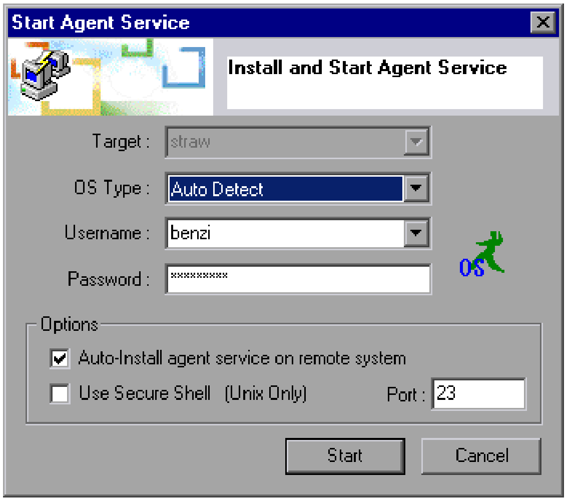
Before you install and start a tuning agent, note the following:

- The tuning agent requires a Java-enabled environment. Ensure that JRE/JDK 1.2 (or later) is installed on the host before you attempt to start the tuning agent. Either JRE or JDK can be used.
- The tuning agent does not require any registry updates. It uses the PE_HOME, PE_USE_SSL, and PE_USE_PORT optional environment variables. Alternatively, you can pass these as arguments to the pe_agent and pe_registry commands. For information on these commands, see "Configuring Tuning Agents," on page 503.
- You do not need to set the CLASSPATH environment variable on the host; the pe_agent and pe_registry commands set CLASSPATH automatically, based on the PE_HOME variable.

To install a tuning agent remotely from the Console machine:

1. Click the host's icon in the Server Configurations tree, and then click the **Start Tuning Agent** button on the toolbar.

The Start Agent Service dialog box is displayed:



Start Agent Service

Install and Start Agent Service

Target : straw

OS Type : Auto Detect

Username : benzi

Password : **********

Options

☑ Auto-Install agent service on remote system

☐ Use Secure Shell (Unix Only) Port : 23

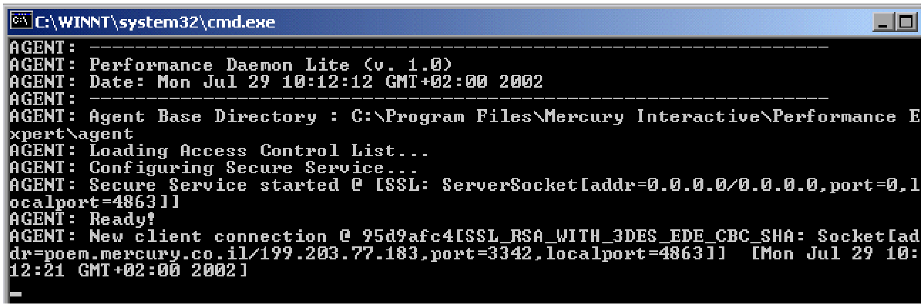
2 Specify the host's OS Type. To let ProTune automatically detect the operating system, choose Auto Detect.

3 Check the **Auto-Install...** box.

4 If you are installing the tuning agent on a UNIX machine, check the **Use Secure Shell** box and specify the port address, if applicable.

5 Click **Start**. If you've configured your settings correctly, ProTune installs the tuning agent on the host. On the host, the tuning agent opens a command window on which you can view the installation activity.

Following is an example of the command window showing the tuning agent's activity on a Windows host when starting:



C:\WINNT\system32\cmd.exe

```
AGENT: -----------------------------------------------------------------
AGENT: Performance Daemon Lite (v. 1.0)
AGENT: Date: Mon Jul 29 10:12:12 GMT+02:00 2002
AGENT: -----------------------------------------------------------------
AGENT: Agent Base Directory : C:\Program Files\Mercury Interactive\Performance E
xpert\agent
AGENT: Loading Access Control List...
AGENT: Configuring Secure Service...
AGENT: Secure Service started @ [SSL: ServerSocket[addr=0.0.0.0/0.0.0.0,port=0,l
ocalport=4863]]
AGENT: Ready!
AGENT: New client connection @ 95d9afc4[SSL_RSA_WITH_3DES_EDE_CBC_SHA: Socket[ad
dr=poem.mercury.co.il/199.203.77.183,port=3342,localport=4863]]  [Mon Jul 29 10:
12:21 GMT+02:00 2002]
_
```

On the Console machine, the host's icon in the Server Configurations tree changes to green, indicating that the connection to the host is alive.

Tip: If the Console displays an error message telling you that it is unable to establish a connection, right-click the host's icon and choose Refresh.

If the tuning agent does not start, or doesn't show the expected information, you may need to configure its settings. See "Configuring Tuning Agents," on page 503.

To install a tuning agent locally on a Windows host:

1. Extract the perfagent.tar file (located in the \ProTune\console\bin directory) to a directory (for example, to *C:\Program Files\Mercury Interactive\Performance Expert*).
2. Set the PE_HOME environment variable (in this example, PE_HOME = C:\Program Files\Mercury Interactive\Performance Expert).
3. Launch the tuning agent batch file. In this example, you would run the following file:

   ```
   C:\Program Files\Mercury Interactive\Performance Expert\agent\bin\pe_agent.bat
   ```

   You can skip step 2 by passing the path to the installation directory to pe_agent.bat in the command line, as in the following example:

   ```
   % pe_agent.bat 0 true C:\Program Files\Mercury Interactive\Performance Expert
   ```

To install a tuning agent locally on a UNIX host:

1. Extract the pe_agent.tar file to a directory (for example, to */usr/local/perfexpert*)
2. Set PE_HOME environment variable (in this example, setenv PE_HOME /usr/local/perfexpert -- for CSH, ...)
3. Launch the tuning agent batch file. In this example, you would run the following file:

   ```
   /usr/local/perfexpert/agent/bin/pe_agent
   ```

You can skip step 2 by passing the path to the installation directory to pe_agent.bat in the command line, as in the following example:

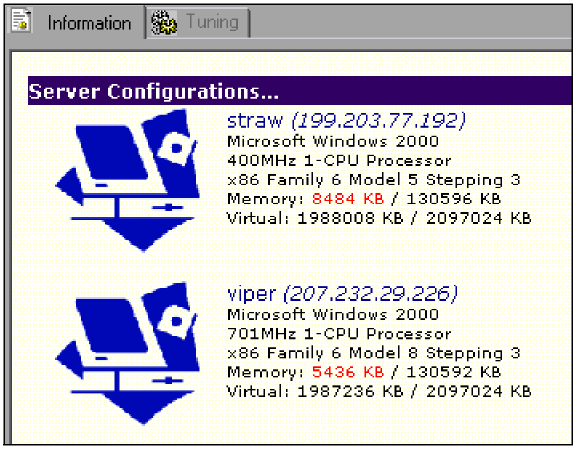
```
% pe_agent.bat 0 true /usr/local/perfexpert
```

Viewing the Host Information

The Tune window contains two tabs:

- The **Information** tab shows you the host property information.
- The **Tuning** tab allows you to change the values of those properties that are configurable.

After you've connected to the host and the tuning agent is running, click the Server Configurations element. The **Information** tab shows summary information for each of the hosts, as in the following example:



Server Configurations...

straw *(199.203.77.192)*
Microsoft Windows 2000
400MHz 1-CPU Processor
x86 Family 6 Model 5 Stepping 3
Memory: 8484 KB / 130596 KB
Virtual: 1988008 KB / 2097024 KB

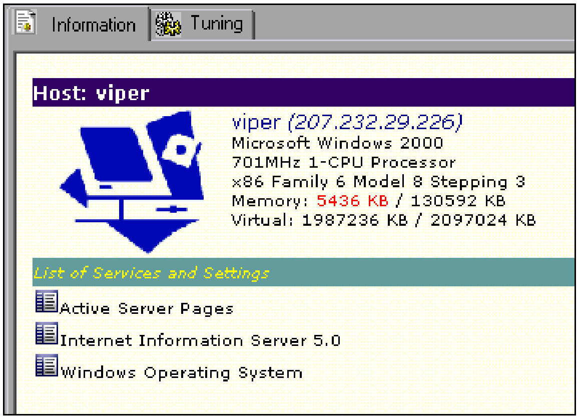
viper *(207.232.29.226)*
Microsoft Windows 2000
701MHz 1-CPU Processor
x86 Family 6 Model 8 Stepping 3
Memory: 5436 KB / 130592 KB
Virtual: 1987236 KB / 2097024 KB

To view detailed information about a host, click the host's icon in the Server Configurations tree. The **Information** tab then displays the following information about the host:

- Hostname and IP address
- CPU type, including number of processors

- Memory—available RAM over total RAM. The virtual memory information shows available memory over total virtual memory (pagefile).
- Services and settings

Following is an example of what the **Information** tab shows when you click a host icon:



Host: viper

viper (207.232.29.226)
Microsoft Windows 2000
701MHz 1-CPU Processor
x86 Family 6 Model 8 Stepping 3
Memory: 5436 KB / 130592 KB
Virtual: 1987236 KB / 2097024 KB

List of Services and Settings

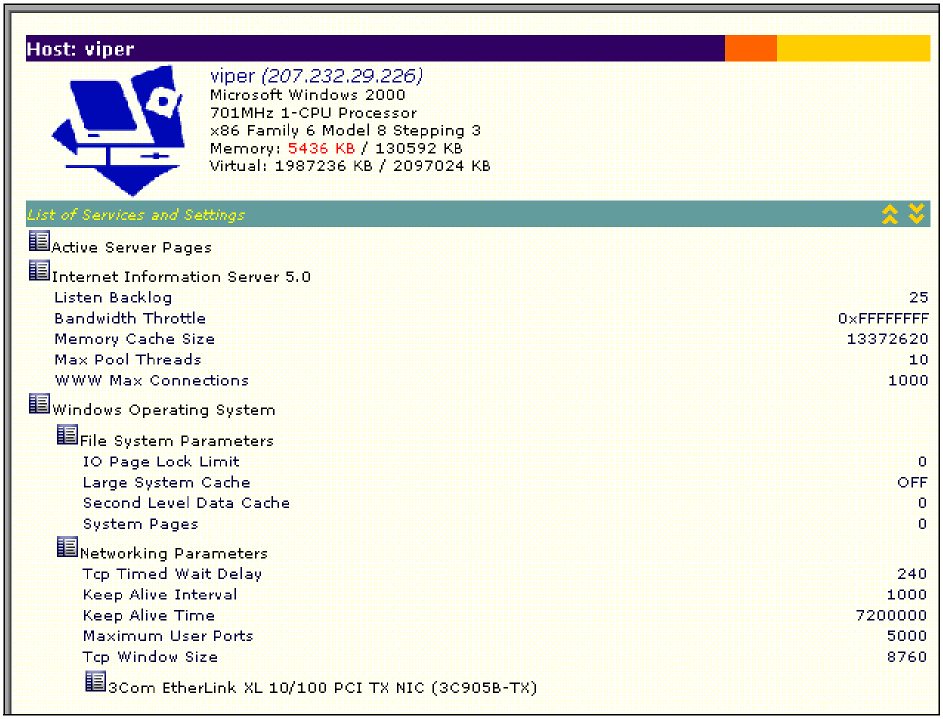
Active Server Pages
Internet Information Server 5.0
Windows Operating System

Each host in the Server Configurations tree contains a list of the running services. Sub-elements of each service are divided into categories and sub-categories, as defined by the agent on the remote host machine.

To view information about a host or host-related service:

1. Click the host or service in the tree; ProTune displays the information in the **Information** tab.
2. Click a host to see the information about all of its services and settings.
3. Click a service to see only information about the selected host, its categories and its sub-categories.

4 To view a node's sub-elements, click the node's icon to expand it.



```
Host: viper
viper (207.232.29.226)
Microsoft Windows 2000
701MHz 1-CPU Processor
x86 Family 6 Model 8 Stepping 3
Memory: 5436 KB / 130592 KB
Virtual: 1987236 KB / 2097024 KB

List of Services and Settings
Active Server Pages
Internet Information Server 5.0
  Listen Backlog                    25
  Bandwidth Throttle        0xFFFFFFFF
  Memory Cache Size           13372620
  Max Pool Threads                  10
  WWW Max Connections             1000
Windows Operating System
  File System Parameters
    IO Page Lock Limit               0
    Large System Cache             OFF
    Second Level Data Cache          0
    System Pages                     0
  Networking Parameters
    Tcp Timed Wait Delay           240
    Keep Alive Interval           1000
    Keep Alive Time            7200000
    Maximum User Ports            5000
    Tcp Window Size               8760
    3Com EtherLink XL 10/100 PCI TX NIC (3C905B-TX)
```

5 To collapse a node, click its icon again.

6 To expand all the nodes, click the **Expand All** button.

7 To collapse all the nodes, click the **Collapse All** button.

Using Expert mode: Some tuning parameters are displayed only if you are in Expert mode.

To enable Expert mode:

➤ Click the icon of the host or service for which you want to enable it, and then choose Expert from the box in the **Tune** tab's toolbar. Alternatively, you can right-click the host or service and choose Expert Mode. Expert mode is enabled for the selected node and its sub-nodes, and the extra parameters are displayed.

To disable Expert mode:

- Choose Normal from the box in the toolbar, or right-click the host or service and choose Expert Mode. This disables Expert mode for the selected host or service.

Changing Tuning Parameter Values

You use the **Tuning** tab to change values of tuning and configuration parameters for the selected host or service. For each parameter, ProTune displays the following:

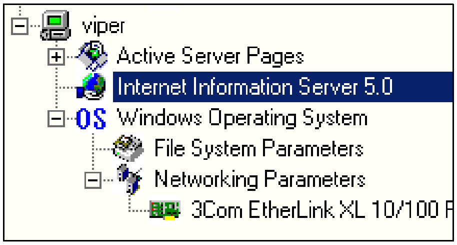
- Current value
- Recommended value—the value recommended by ProTune
- New value—the value that you entered (if you changed the old value)

The different parameters are color-coded as follows:

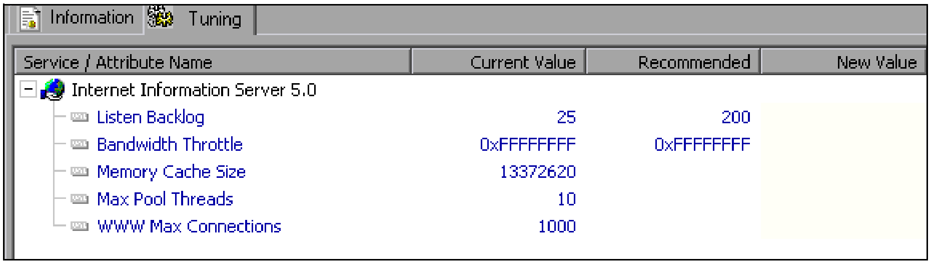
- blue—important
- black—for advanced users
- red—critical tuning parameter
- gray—read-only

To change the value of a host parameter:

1 Select the host or service that you want to tune, by clicking the relevant icon in the Server Configuration tree.



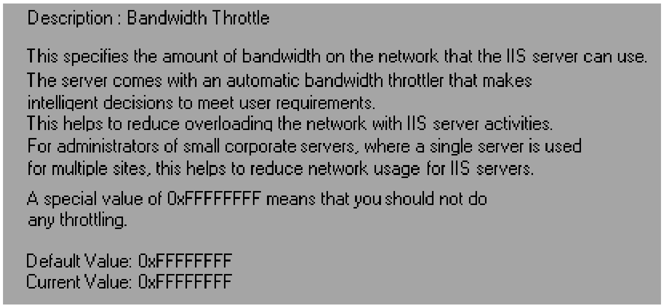
In the **Tuning** tab, ProTune displays the parameters relevant to the selected host or service.



Service / Attribute Name	Current Value	Recommended	New Value
Internet Information Server 5.0			
Listen Backlog	25	200	
Bandwidth Throttle	0xFFFFFFFF	0xFFFFFFFF	
Memory Cache Size	13372620		
Max Pool Threads	10		
WWW Max Connections	1000		

The values in the Recommended column are based on information in ProTune’s knowledge base.

When you click a property, the lower section of the **Tuning** tab displays a description of the property and its values, and may include tuning recommendations.



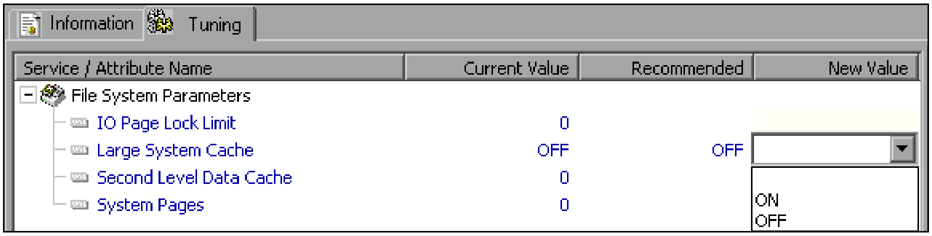
Description : Bandwidth Throttle

This specifies the amount of bandwidth on the network that the IIS server can use.
The server comes with an automatic bandwidth throttler that makes
intelligent decisions to meet user requirements.
This helps to reduce overloading the network with IIS server activities.
For administrators of small corporate servers, where a single server is used
for multiple sites, this helps to reduce network usage for IIS servers.

A special value of 0xFFFFFFFF means that you should not do
any throttling.

Default Value: 0xFFFFFFFF
Current Value: 0xFFFFFFFF

2 Click the parameter that you want to configure, and click the parameter’s New Value column. If the parameter is configurable, a text box opens, or a list box appears.



Service / Attribute Name	Current Value	Recommended	New Value
File System Parameters			
IO Page Lock Limit	0		
Large System Cache	OFF	OFF	
Second Level Data Cache	0		
System Pages	0		

3 Enter the new value. In the case of a list box, choose it from the list.

Updating the Host or Service with Changes

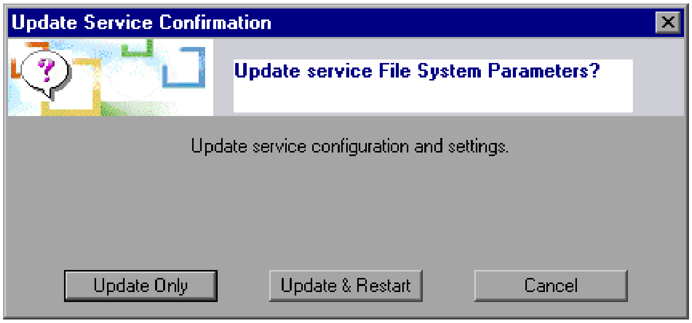
Changes that you make to parameter values do not take effect until you update the relevant host or service.

When you update a host, all of the services in the host's tree are updated (if any of the parameters have been assigned new values).

When you update a service or service category, only the selected service or category is updated.

To update a host or service:

1 Click **Commit Host Changes**, or right-click the host or service icon in the Server Configurations tree and click Update. The Update Service Confirmation dialog box is displayed.



Update Service Confirmation

Update service File System Parameters?

Update service configuration and settings.

2 Click **Update Only** to update the selected host or service, or click **Update & Restart** to update the host or service and then restart it. Note that some services need to be restarted for the changes to take effect.

ProTune displays a message informing you that the changes have taken effect.

3 To view the changed values, right-click the host icon and click **Refresh**.

28

Exporting and Importing Configuration Settings

This chapter describes how to export and import configuration settings.

It includes the following topics:

➤ Exporting a Host or Service’s Configuration Settings

➤ Importing Configuration Settings for a Host or Service

➤ Saving and Loading Profiles

➤ Creating a New Profile

About Exporting and Importing Configuration Settings

Once you’ve viewed (and possibly changed) configuration settings on your hosts and services, you can export the settings to save them for use in the future. You can also import previously saved settings to hosts and services.

Exporting settings allows you to track changes on the hosts and services, and to import the settings into other devices.

Importing previously saved settings for a host or service allows you to replicate desired settings across multiple hosts with similar configurations, avoiding the need to manually update the settings on each host or service individually.

You can export and import settings for a service, a host, or a group of hosts. When you export the settings of a group of hosts, you create a profile. You can subsequently import the profile.

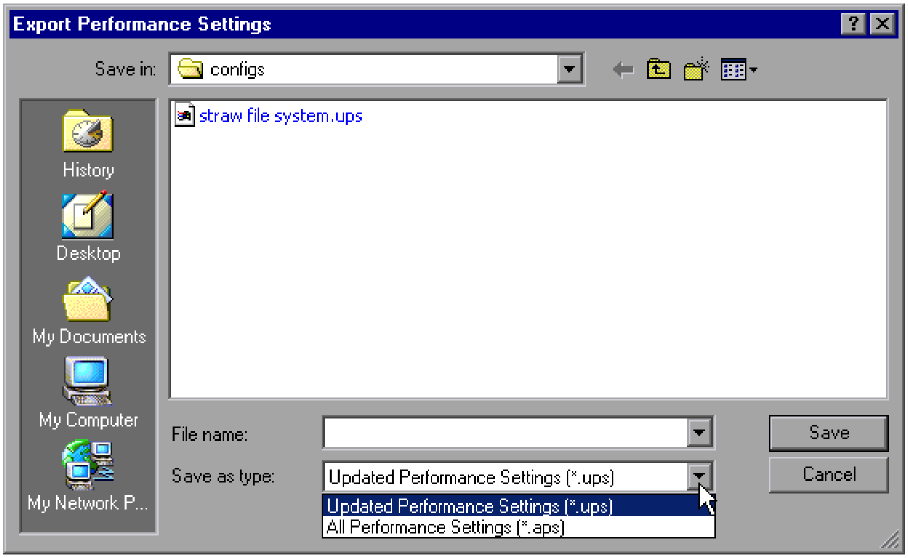
Exporting a Host or Service's Configuration Settings

The Export function allows you to store the configuration settings of the entire host, or only a subset of these settings (for example, the settings for a specific service). You can export the following sets of values:

- Updated settings. This means exporting only the settings to which you've assigned new values but have not committed. See "Updating the Host or Service with Changes," on page 458 for details on how to commit changes.
- All the settings, including those that have not been changed.

To export configuration settings:

1. Click the host or service whose values you want to export and then click **Save/Export configuration settings**. Alternatively, you can right-click the host or service and click Export Settings. The Export Performance Settings dialog box is displayed.



Export Performance Settings

Save in: configs

straw file system.ups

File name:

Save as type: Updated Performance Settings (*.ups)

History
Desktop
My Documents
My Computer
My Network P...
Updated Performance Settings (*.ups)
All Performance Settings (*.aps)
Save
Cancel

2. From the Save as Type box, choose whether to save only the updated (and uncommitted) settings or all the settings for the selected host or service. If you save only the updated settings, ProTune saves them in a file with a *.ups* extension; if you choose to save all the settings, they are saved in a file with an *.aps* extension.

3 Enter a meaningful name for your settings file and click **Save**.

ProTune saves your settings in a file with the name you specified.

Importing Configuration Settings for a Host or Service

The Import function allows you to import previously stored settings and apply them to hosts and services. You can import and apply settings to:

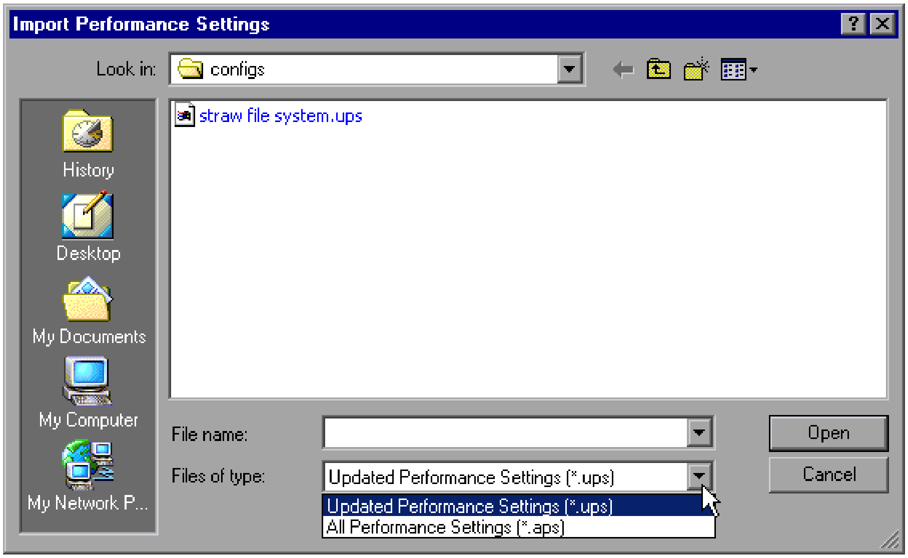
- Hosts
- Services
- Sub-categories of services

Only the entries that are relevant to the host, service or sub-category into which you are importing will be imported from the settings file. For example, you may have a file that contains settings for both Internet Information Server (IIS) and Windows Operating System. If you click the host's Windows Operating System tree element and then import the settings from the file, ProTune loads only the network settings for Windows Operating System, not the IIS settings. Another example: Importing Apache settings into a host that includes IIS does not overwrite the host's IIS settings.

To import configuration settings:

1 Click the host or service whose values you want to import and then click **Load/Import configuration settings**. Alternatively, you can right-click the

host or service and click Import Settings. The Import Performance Settings dialog box is displayed.



Import Performance Settings

Look in: configs

straw file system.ups

File name:

Files of type: Updated Performance Settings (*.ups)

History
Desktop
My Documents
My Computer
My Network P...
Updated Performance Settings (*.ups)
All Performance Settings (*.aps)
Open
Cancel

2 From the Files of type box, choose whether to view the files containing only updated performance settings (files with *.ups* extensions), or files containing all the performance settings for the selected host or service (files with *.aps* extensions).

3 Select the settings file from which you want to import, and click **Open**.

ProTune imports the settings into the specified host, service or category.

4 To view the new settings, right-click the host icon (not the service) and click **Refresh**.

Saving and Loading Profiles

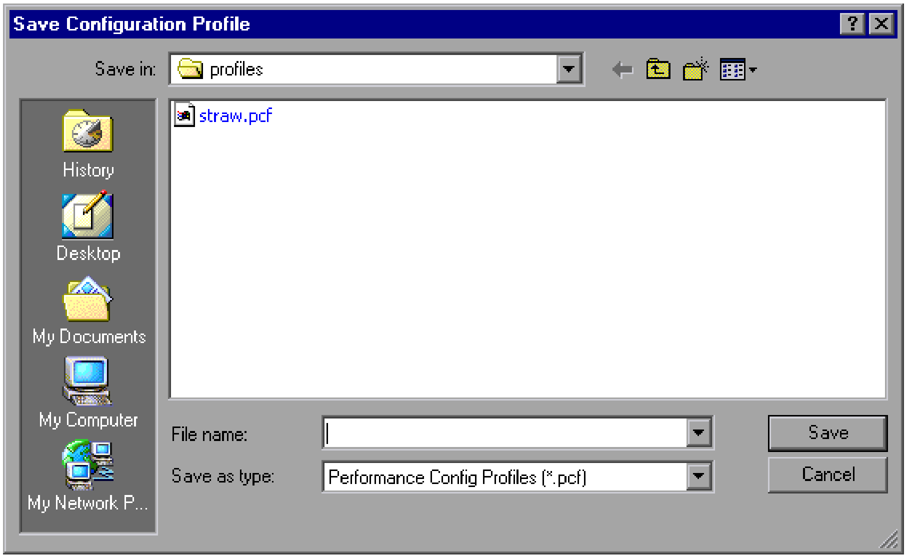
A profile contains all the configuration settings for a group of hosts.

When dealing with large clusters of hosts that may have different operational functions, creating profiles lets you save and load configuration settings for all the hosts in the group. For example, you may find it useful to group a cluster of Web servers into a profile, or group a set of servers relating to an e-commerce or intranet application.

When you save a profile, ProTune saves all the configuration settings for all the hosts that appear in the Tune window.

To save a profile:

1 Click **Save Tuning Profile**, or right-click the Server Configuration tree element and click Save Profile. The Save Configuration Profile dialog box is displayed.



Save Configuration Profile

Save in: profiles

straw.pcf

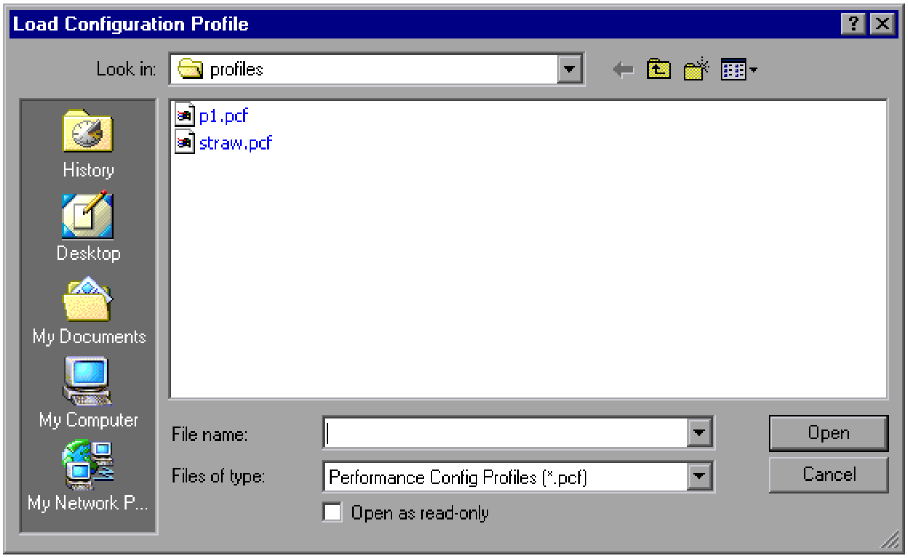
File name:

Save as type: Performance Config Profiles (*.pcf)

2 Enter a name for your profile and click **Save**. ProTune saves the profile with a *.pcf* extension.

To load a profile:

1 Click **Open Tuning Profile**, or right-click the Server Configuration tree element and click Load Profile. The Load Configuration Profile dialog box is displayed, showing the saved profiles as *.pcf* files.



Look in: profiles

Files of type: Performance Config Profiles (*.pcf)

2 Choose the profile you want to load, and click **Open**. ProTune loads the profile and connects to each of the profile's hosts to retrieve and load the profile's settings into the current host configuration. ProTune displays the new configuration in the **Tune** tab.

Creating a New Profile

Note: When ProTune creates a new profile, it erases the current profile from memory. This means that all the server settings that you defined will be cleared from the window. If you need to save your settings, make sure you save the existing profile before creating a new one.

To create a new profile:

1. Click **New Tuning Profile**. ProTune removes the server icons and their accompanying information from the **Tune** tab.
2. Add the required servers and services, and connect to them.
3. Save the new profile.

29

Configuring Tuning Agents

A tuning agent is an application that runs on the host that you want to tune, and allows you to tune the host remotely. This chapter describes the tuning agent's features and explains how to configure its settings.

The chapter includes the following topics:

➤ Changing the Tuning Agent's Port

➤ Using the Performance Expert Registry

➤ Automatically Starting the Tuning Agent when Booting

About Configuring Tuning Agents

The tuning agent interrogates the host and gathers performance-related information and tuning parameters, and passes the information to the Console. It allows the user to remotely configure and administer the target system.

The tuning agent is a passive service and does not consume any CPU resources when not processing requests. The agent requires only between 10 and 15MB of memory on the host.

The Console uses the tuning agent to change the target system's configurable parameters.

You install the tuning agent over the network from the ProTune Console workstation, or locally on the target system (see "Installing and Starting a Tuning Agent," on page 450).

Changing the Tuning Agent's Port

By default, all communication between the Console and the tuning agent is handled by a proprietary messaging protocol encoded and secured over Secure Sockets Layer (SSL). You can change the tuning agent's default port from OTP-SSL 4863 to a user defined port.

When you install the tuning agent, the installation process installs the pe_agent.bat (Windows) and pe_agent (UNIX) batch files on the host.

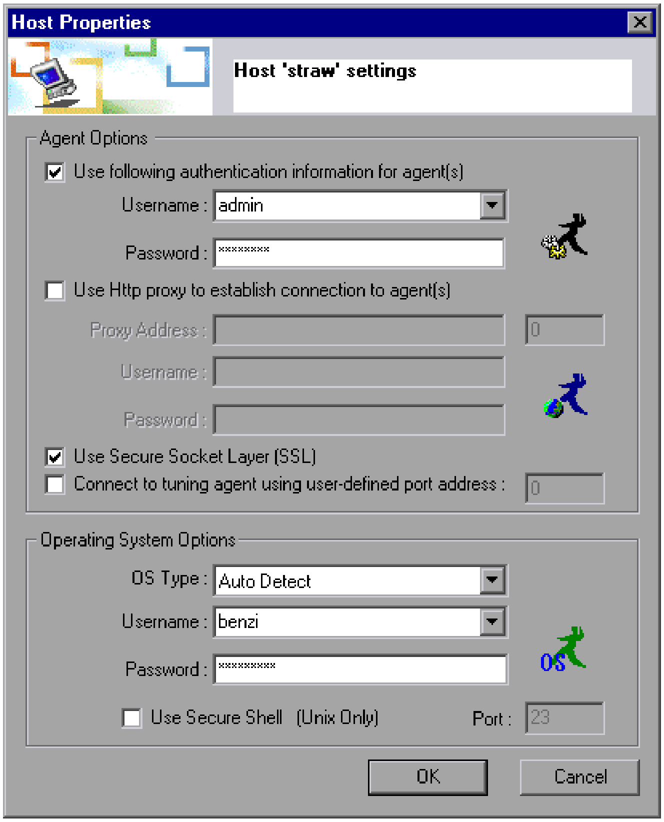
Tip: On a Windows host, the pe_agent.bat file is located in the *Program Files*\Mercury Interactive\Performance Expert\agent\bin directory.

To modify the agent's listening port, use one of the following methods:

Via the Host Properties dialog box:

1 In the Server Configurations tree, click the host and then click the **View Host Properties** button. Alternatively, right-click the host and choose Properties.

The Host Properties dialog box is displayed:



Host Properties

Host 'straw' settings

Agent Options

☑ Use following authentication information for agent(s)

Username : admin

Password : *********

☐ Use Http proxy to establish connection to agent(s)

Proxy Address : 0

Username :

Password :

☑ Use Secure Socket Layer (SSL)

☐ Connect to tuning agent using user-defined port address : 0

Operating System Options

OS Type : Auto Detect

Username : benzi

Password : **********

☐ Use Secure Shell (Unix Only) Port : 23

2 Check the **Connect to tuning agent...** box and specify the port in the adjacent field.

Specifying the port in the command line:

➤ Specify the listening port when you invoke the batch file from the command line. Use the following syntax:

```
pe_agent <Port#> <SSL_Flag> <Path_to_PE_Installation>
```

The following examples illustrate how to use the batch file:

Command	Action
pe_agent 1234	Launches the tuning agent at port 1234 with SSL enabled.
pe_agent 1235 false	Launches the tuning agent at port 1235 with SSL disabled.
pe_agent 1236 true C:\protune	Launches the tuning agent at port 1236 with SSL enabled, using configuration and tuners from C:\protune.

➤ Set the tuning agent's environment variables and then run the batch file. The following example illustrates this method:

```
set PE_USE_PORT=4444
set PE_USE_SSL=true
set PE_HOME=C:\protune
pe_agent
```

Tip: On a UNIX host, use a similar procedure, depending on your UNIX shell.

Using the Performance Expert Registry

The Performance Expert Registry allows you to configure the tuners on a host.

Each environment (for example, IIS, Apache, and Oracle) has a dedicated tuner that is capable of administering the environment. In some cases the tuner needs information on where to find the application that needs tuning, and may also need logon credentials.

The Performance Expert Registry provides a command-line interface for configuring the individual tuners.

To invoke the Performance Expert Registry:

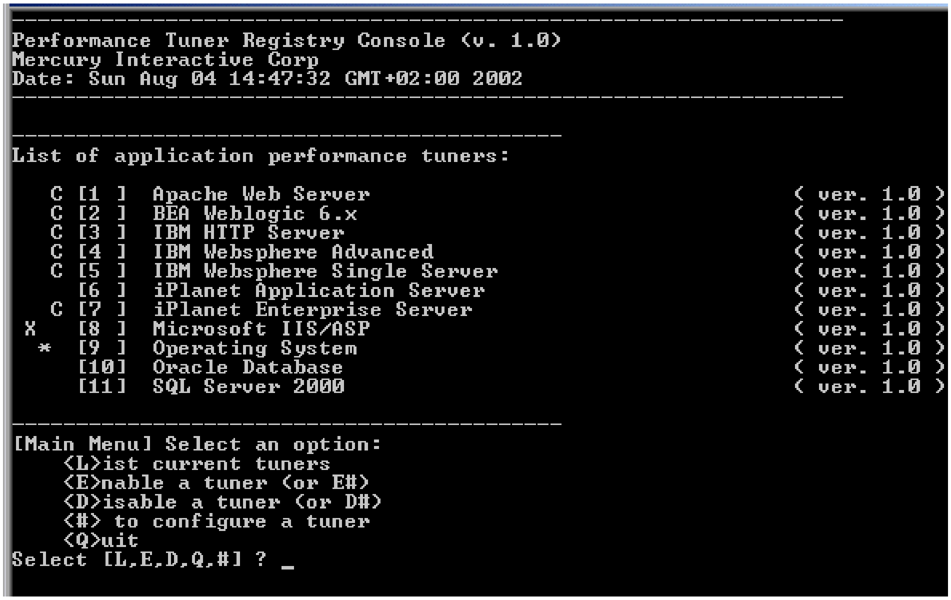
1 On the host that is being tuned, set the PE_HOME environment variable so it points at the Performance Expert's home directory. **Note:** When you install the tuning agent remotely from the Console machine, ProTune sets the environment variable to this value.

2 From the command line, enter one of the following commands:

➤ pe_registry.bat (Windows)

➤ pe_registry (UNIX)

ProTune invokes the Performance Expert Registry and displays the Performance Tuner Registry Console:



```
-----------------------------------------------------------------
Performance Tuner Registry Console (v. 1.0)
Mercury Interactive Corp
Date: Sun Aug 04 14:47:32 GMT+02:00 2002
-----------------------------------------------------------------

-------------------------------------------
List of application performance tuners:

   C [1 ]  Apache Web Server                           ( ver. 1.0 )
   C [2 ]  BEA Weblogic 6.x                            ( ver. 1.0 )
   C [3 ]  IBM HTTP Server                             ( ver. 1.0 )
   C [4 ]  IBM Websphere Advanced                      ( ver. 1.0 )
   C [5 ]  IBM Websphere Single Server                 ( ver. 1.0 )
     [6 ]  iPlanet Application Server                  ( ver. 1.0 )
   C [7 ]  iPlanet Enterprise Server                   ( ver. 1.0 )
 X   [8 ]  Microsoft IIS/ASP                           ( ver. 1.0 )
  *  [9 ]  Operating System                            ( ver. 1.0 )
     [10]  Oracle Database                             ( ver. 1.0 )
     [11]  SQL Server 2000                             ( ver. 1.0 )

-------------------------------------------
[Main Menu] Select an option:
    <L>ist current tuners
    <E>nable a tuner (or E#)
    <D>isable a tuner (or D#)
    <#> to configure a tuner
    <Q>uit
Select [L,E,D,Q,#] ? _
```

The window displays a list of all the services for which tuners are available. The services are marked as follows:

Sign	Indicates	Comments
*	The tuner is active on the host (the service is installed on the machine).	If the service is not installed on the host, the tuner is not marked as active.
C	You may need to configure the tuner before it can become active. For example, you may need to specify where the service is installed.	
X	The tuner is disabled.	The user cannot view information about the service for which the tuner is intended, and cannot tune or administer the service.

The Performance Tuner Registry Console allows you to perform the following actions:

- ➤ List the current tuners
- ➤ Enable a tuner
- ➤ Disable a tuner
- ➤ Configure a tuner
- ➤ Quit the Performance Tuner Registry Console

To list the current tuners:

➤ Type “L” and press <Enter>.

To enable a tuner:

1 Type “E” and press <Enter>.

The Performance Tuner Registry Console asks you to enter the ID of the tuner that you want to enable.

2 Type the tuner ID (the number in brackets that appears before the tuner's name) and press <Enter>.

Note: If the tuner requires configuration (indicated by the letter "C" before the tuner's name), you cannot enable it.

The next time you list the current tuners, the enabled tuner appears with an asterisk.

To disable a tuner:

1 Type "D" and press <Enter>.

The Performance Tuner Registry Console asks you to enter the ID of the tuner that you want to disable.

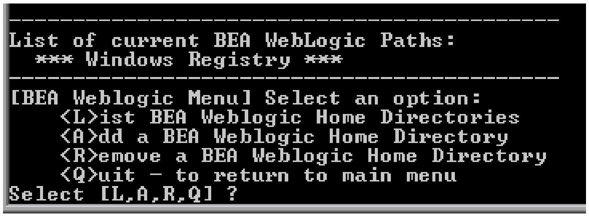
2 Type the tuner ID (the number in brackets that appears before the tuner's name) and press <Enter>.

The next time you list the current tuners, the enabled tuner appears without an asterisk.

To configure a tuner:

1 Type the tuner's number to select it, and press <Enter>.

The configuration menu for the selected tuner is displayed. The following example shows the configuration menu for the WebLogic Application Server tuner:



```
-------------------------------------------
List of current BEA WebLogic Paths:
  *** Windows Registry ***
-------------------------------------------
[BEA Weblogic Menu] Select an option:
    <L>ist BEA Weblogic Home Directories
    <A>dd a BEA Weblogic Home Directory
    <R>emove a BEA Weblogic Home Directory
    <Q>uit - to return to main menu
Select [L,A,R,Q] ?
```

2 Follow the onscreen instructions for configuring the selected tuner.

To quit the Performance Tuner Registry Console:

➤ Type "Q" and press <Enter>.

Automatically Starting the Tuning Agent when Booting

You can configure the host to start the tuning agent automatically when the host is started.

On a Windows system: Using regedit.exe, create a key in the Windows registry under:
[HKLM\Software\Microsoft\Windows\CurrentVersion\Run]
PE_AGENT = "%PE_HOME%\agent\bin\pe_agent.bat" (REG_SZ)

On a UNIX system: Update your startup file under /etc, and add a command to launch pe_agent.

When you launch the tuning agent (pe_agent.bat or pe_agent), you can include the following optional arguments:

Argument	Specifies	Values
[PE_USE_PORT]	listening port	Default: 0
[PE_USE_SSL]	whether SSL is enabled	True False
[PE_HOME]	location of PE_HOME (if the PE_HOME environment variable is not defined)	

Following is the syntax for launching the tuning agent:

```
pe_agent[.bat] [PE_USE_PORT] [PE_USE_SSL] [PE_HOME]
```

Following is an example of how to run the pe_agent.bat command:

```
pe_agent.bat 0 true C:\Program Files\Mercury Interactive\PerfExpert
```

30

Tune Tab Functions

This chapter describes various functions that are available to you via the **Tune** tab.

The chapter describes the following functions:

- Start a Service
- Stop a Service
- Reboot all Host Machines
- Restart a Service or Host
- Reconnect the Console to a Server
- Stop the Tuning Agent
- Print Host Configurations
- Reload Host Configuration
- Remove Host from Server Configurations Tree

About Tune Tab Functions

Some tuning functions can be performed by all categories of users—whether they have only read-only permissions, or update permissions, or full administrative privileges. Some functions are available only to users who have administrator privileges (that is, users who connect with the *admin* username). If you attempt to perform a function for which you are not authorized, ProTune displays an error message.

For details of the various types of users and their usernames and passwords, see the table on page 449.

Start a Service

(For users with administrator access only).

To start a service:

1 Right-click the service, choose Admin, and then choose Start Service.

ProTune displays a dialog box requesting confirmation.

2 Click **Yes** to start the service.

Stop a Service

(For users with administrator access only).

To stop a service:

1 Right-click the service, choose Admin, and then choose Stop Service.

ProTune displays a dialog box requesting confirmation.

2 Click **Yes** to stop the service.

Reboot all Host Machines

(For users with administrator access only).

To reboot all the host machines:

1 Click the Server Configurations node and then click the **Restart Service/Host** button. Alternatively, right-click the Server Configurations node, choose Admin, and then choose Reboot All.

ProTune displays a dialog box requesting confirmation.

2 Click **Reboot All** to reboot all the host machines.

Restart a Service or Host

(For users with administrator access only).

Restarting involves stopping the service or host and then starting it.

To restart a service or host:

1 Click the service or host in the Service Configurations tree and then click the **Restart Service/Host** button. Alternatively, right-click the service or host, choose Admin, and then choose Restart Service or Reboot Host to restart the service or host, respectively.

ProTune displays a dialog box requesting confirmation.

2 Click **Yes** to restart the service or host.

Reconnect the Console to a Server

Once the tuning agent has been installed on a server, you can access the server by clicking the **Connect to Host button** on the toolbar. In the Connect to Server dialog box, choose the server you want to tune and click **Connect**. If the tuning agent is running on the server, the Console connects to the server via the tuning agent, and shows you the server information.

If the tuning agent has been installed on the server but is not currently running, start it by clicking the **Start Tuning Agent** button. Clear the Auto-Install... box, verify that the other fields have the correct values, and click **Start**.

On the Console machine, the server icon in the Server Configurations tree changes to green, indicating that the connection to the server is alive.

Stop the Tuning Agent

To stop the tuning agent that is running on the server, click the server's icon and click the **Stop Tuning Agent** button. When ProTune asks you to confirm the action, click **Yes**.

Print Host Configurations

You may find it useful to keep a hard-copy record of a host's tuning settings. ProTune allows you to print the configuration settings as they appear in the **Information** tab.

To print a host's configuration settings:

1. Expand the icons in the **Information** tab so that the data that you want to print is displayed.
2. Right-click the server's icon in the Servers Configurations tree, and choose Print.

Reload Host Configuration

To reload the current settings from a host machine, click the host's icon in the Servers Configurations tree and click the **Reload Host Configuration** button. Alternatively, right-click the host's icon and choose Refresh.

Remove Host from Server Configurations Tree

To remove a host from the Server Configurations tree, right-click the host's icon and choose Remove.

31

Tuning UNIX Hosts

This chapter describes the permissions, access rights and actions that you need to perform before you can tune a host machine running a UNIX operating system.

It includes the following sections:

- Solaris Requirements
- IBM AIX Requirements
- HP-UX Requirements
- Linux Requirements

Solaris Requirements

Access Rights and Permissions

The user running the tuning agent must have access rights and permissions to execute the following commands:

Command	Gives this information	Default location
psrinfo	CPU speed and number of CPUs	/usr/sbin
prtconf	Total RAM	/usr/sbin
vmstat	Available RAM	/usr/sbin
swap	Total and available virtual memory	/usr/sbin

Command	Gives this information	Default location
ndd	Network tuning parameters	/usr/sbin
/etc/system file	File system tuning parameters	

PATH Environment

Update the PATH environment so it includes the /usr/sbin directory.

Verification

After updating the path, execute each of the commands listed in the table above to verify that you have the appropriate access rights for executing them. **Note:** Some commands may require root privileges. Also verify the existence of the system file in the /etc directory.

IBM AIX Requirements

Access Rights and Permissions

The user running the tuning agent must have access rights and permissions to execute the following commands:

Command	Gets this information	Default location
uname	Name and version of the operating system	/usr/bin
bootinfo	Total RAM	/usr/bin
vmstat	Total and available virtual memory	/usr/bin
lsdev	Number of processors	/usr/bin

PATH Environment

Update the PATH environment so it includes the /usr/bin directory.

Verification

After updating the path, execute each of the commands listed in the table above to verify that you have the appropriate access rights for executing them. **Note:** Some commands may require root privileges.

HP-UX Requirements

Access Rights and Permissions

The user running the tuning agent must have access rights and permissions to execute the following commands:

Command	Gets this information	Default location
uname	Name and version of the operating system	/usr/bin
model	CPU speed	/usr/sbin
swapinfo	Total and available virtual memory	/usr/sbin
dmesg	Total and available RAM	/etc
ioscan	Number of processors	/usr/sbin

PATH Environment

Update the PATH environment so it includes the following directories:

- /usr/sbin
- /usr/bin
- /etc

Verification

After updating the path, execute each of the commands listed in the table above to verify that you have the appropriate access rights for executing them. **Note:** Some commands may require root privileges.

Linux Requirements

Access Rights and Permissions

The user running the tuning agent must have access rights and permissions to execute the following commands:

Command	Gets this information	Default location
uname	Name of the operating system	/usr/bin
/proc/cpuinfo file	Operating system version, CPU speed and total number of CPUs	
/proc/meminfo file	All memory statistics	

PATH Environment

Update the PATH environment so it includes the /usr/bin directory.

Verification

After updating the path, execute each of the commands listed in the table above to verify that you have the appropriate access rights for executing them. **Note:** Some commands may require root privileges. Also check for the existence of the meminfo and cpuinfo files in the /proc directory.

Part VII

Appendixes

- Troubleshooting the Console
- Performing Path Translation
- Working in Expert Mode
- Working with Server Monitor Counters
- Working with Digital Certificates

A

Troubleshooting the Console

ProTune enables you to test entire applications. If one of the components of the application is not configured properly, ProTune sessions will not run.

This appendix discusses the most common ProTune problems:

- ProTune Communications
- Failure to Communicate with a Load Generator
- Failure to Connect to the AUT Database
- Failure to Access Files
- Failed Vusers or Transactions
- Increasing the Number of Vusers on a Windows Machine
- Troubleshooting Firewalls

About Troubleshooting

ProTune relies heavily upon communication between machines on a network. If communication is not established properly, the Console will be unable to send commands to remote load generators and the session will fail. By understanding the reason for the failure and determining when the failure occurred, you can solve most of the communication-related problems.

In order to ensure that the problem lies with your session and not your script, you should verify that your script runs properly on all remote load generators as a stand-alone:

- Test your GUI scripts on Windows platforms using WinRunner.
- Test your scripts on UNIX platforms by running them from the command line.
- Test all other types of scripts on Windows platforms by running them from VuGen, or by running a single user from the Console.

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

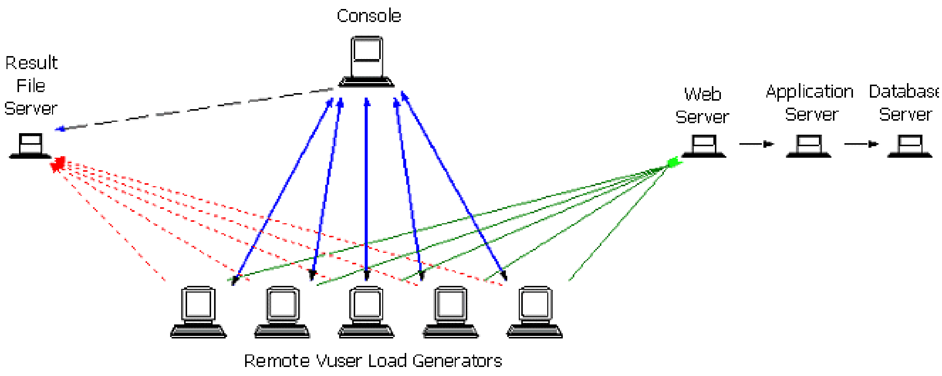
For more information on running scripts in stand-alone mode, refer to the appropriate guide for creating scripts.

ProTune Communications

Most communication problems can be solved if you understand your ProTune configuration. This knowledge helps you to determine the source of the problem and perform the necessary actions to correct it.

The following diagram illustrates a sample network running ProTune. There are five servers: The ProTune Console, the Web server, the application server, the database server, and the file server which stores the session results (note that result files can also be saved on a non-dedicated server). There are five remote load generators, each one running multiple Vusers.

The arrows indicate the type of communication necessary between the elements of the network. The Vusers communicate with the Console in both directions (send/receive), but with the file server in one direction (send). The Console must have access to the file server. All Vusers participating in the session must be able to communicate with the Web server in both directions (send/receive). In order for a client machine to connect to the server machine, it must be able to resolve the server machine name.



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

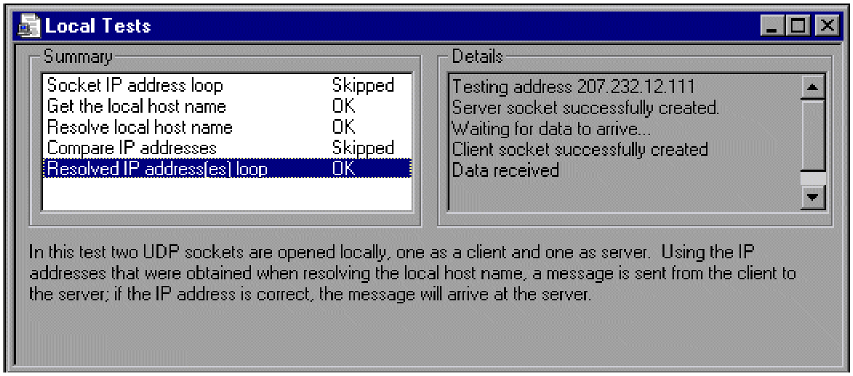
Failure to Communicate with a Load Generator

The most common communication error is the failure of the Console machine to connect with a remote load generator. Check the following items:

- TCP/IP setup
- TCP/IP connectivity
- Load generator connections
- UNIX shell

Checking TCP/IP Setup

The first step in checking your configuration is to verify your machine's TCP/IP setup. ProTune includes a utility called Hostinfo (hostinfo.exe), located under ProTune's bin directory. This utility provides information about the current machine—local name and local address. It also insures that TCP/IP is properly installed on the current machine.



Local Tests

Socket IP address loop	Skipped
Get the local host name	OK
Resolve local host name	OK
Compare IP addresses	Skipped
Resolved IP address(es) loop	OK

Testing address 207.232.12.111
Server socket successfully created.
Waiting for data to arrive...
Client socket successfully created
Data received

In this test two UDP sockets are opened locally, one as a client and one as server. Using the IP addresses that were obtained when resolving the local host name, a message is sent from the client to the server; if the IP address is correct, the message will arrive at the server.

When you invoke Hostinfo, it automatically verifies the TCP stack by:

- retrieving and resolving the local machine name
- retrieving and resolving the IP address

To resolve the IP address, Hostinfo tries to communicate using two UDP sockets on the same machine. It verifies that the IP address obtained while resolving the machine name is the same as the actual IP address of this machine.

To display the results of a test in the Details box, highlight the test name.

Note that the Edit menu in Hostinfo allows you to copy all machine information to the clipboard for sending to support personnel.

Checking TCP/IP Connectivity

Make sure that TCP/IP connectivity is functional on the Console and Vuser machines. Use a ping utility or type ping <server_name> from the DOS command line to verify communication with a remote machine. Make sure that the remote load generator and Console machines can ping each other by IP addresses and machine names.

If the ping does not respond, or fails with a time-out, then the machine name is not recognized. To solve this problem, edit the hosts file, located in the *WINNT\system32\drivers\etc* directory, and add a line with both the IP address, and the name. For example:

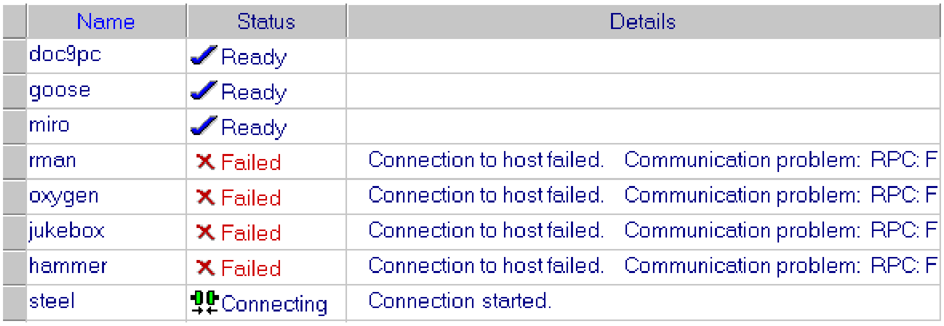
```
#      102.54.94.97     rhino.acme.com     # source server
#      38.25.63.10      x.acme.com         # x client host
```

Load Generator Connections

To verify the load generator connectivity, connect to each one of the remote load generators from the Console's Load Generators dialog box. In the load generator's Platform field, select a Windows or UNIX platform. Select the load generator(s) and click the Connect button. The status changes to *Connecting*.

If the Connection fails, the status changes to *Failed* and details are written to the Details box. Double-click the details box for more information about a failure.

If a connection succeeds, the status changes to *Ready*, and the actual platform name appears in the Platform box (such as WINNT, UNIX, etc.)



Name	Status	Details
doc9pc	✓ Ready	
goose	✓ Ready	
miro	✓ Ready	
rman	✗ Failed	Connection to host failed. Communication problem: RPC: F
oxygen	✗ Failed	Connection to host failed. Communication problem: RPC: F
jukebox	✗ Failed	Connection to host failed. Communication problem: RPC: F
hammer	✗ Failed	Connection to host failed. Communication problem: RPC: F
steel	Connecting	Connection started.

If your session uses several domains (for example, Vusers on a different domain than the Console), the Console may have trouble communicating with the load generators. This occurs because the Console uses the short load generator name—not including the domain—by default. To solve this, you must tell the Console to determine the full load generator names, including the domains.

Modify the *miccomm.ini* file in the Console machine's Windows directory as follows:

```
[tcpnet]
LocalHostNameType= 1
```

The possible values for LocalHostNameType are:

0 - Attempt to use the full machine name.
1 - Use the short machine name. This is the default.

Note: In certain environments such as WINS, load generators are unable to resolve machine names.

Connecting to a Console with Multiple IP Addresses

If the load generator machine does not recognize the Console machine by its short name or full name, and the Console machine has more than one IP address, you can define an alias name for the Console machine in the load generator's *hosts* file, located in the WINNT\system32\drivers\etc directory. The alias name should point to the IP address you want the load generator to recognize. For example: 255.0.0.1 delta.

UNIX Shell

For UNIX Vusers, make sure that the Windows Console can execute a remote shell command. Type the following at the DOS command prompt: rsh -l <UNIX user login name> <load generator name> <command>. If you get a message indicating that permission is denied, make sure the *.rhosts* file in your UNIX home directory contains Console machine permission for the user login name. In some cases, a "+" character must be added at the end of the *.rhosts* file. For example, if you log on to the Console as *bill* and connect to the UNIX load generator as *mike*, you must ensure that *mike* allows *bill* to log on using his name. This can be done by adding the line "+ bill" at the beginning of mike's *.rhosts* file.

For more information on setting user login names, see "Configuring Load Generator Settings" on page 61.

To use UNIX without RSH:

1 On the UNIX Load Generator machine, run the agent daemon by running the following command from *<ProTune directory>/bin*:

```
m_daemon_setup -install
```

This runs a daemon called m_agent_daemon, and if successful you will receive a message: m_agent_daemon installed successfully.

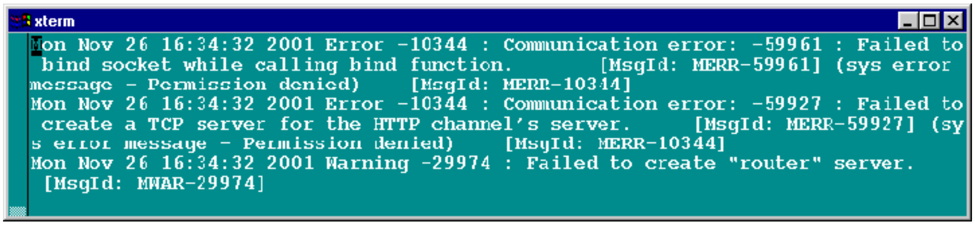
The agent will now keep running, even if the user is logged off. It will only stop running using the command explained in step 3, or by rebooting the machine.

➤ If you receive the message ERROR: File m_agent_daemon doesn't exist, this means that you are not in the same directory as the file (meaning not in *<ProTune_root>/bin* directory, or the file really doesn't exist, which indicates a problem with the installation).

➤ If a daemon of this name is already being run by the same user you will receive the following warning:
WARNING: Could not install m_agent_daemon, reason - user <user_name> is already running m_agent_daemon on this machine.

➤ If an error occurred, you will receive the following error message:
ERROR: Could not install m_agent_daemon. Check log file m_agent_daemon[xxx].log in your temp directory.

- ➤ If you look at the log file m_agent_daemon[xxx].log in the temp directory, you will see the following errors, even if the installation succeeded:



```
Mon Nov 26 16:34:32 2001 Error -10344 : Communication error: -59961 : Failed to
 bind socket while calling bind function.        [MsgId: MERR-59961] (sys error
message - Permission denied)     [MsgId: MERR-10344]
Mon Nov 26 16:34:32 2001 Error -10344 : Communication error: -59927 : Failed to
 create a TCP server for the HTTP channel's server.      [MsgId: MERR-59927] (sy
s error message - Permission denied)     [MsgId: MERR-10344]
Mon Nov 26 16:34:32 2001 Warning -29974 : Failed to create "router" server.
 [MsgId: MWAR-29974]
```

These messages appear because the ProTune agent always tries to open port number 443 (because any agent can be a MI Listener, and the MI Listener always listens to this port), and in UNIX machines, this port cannot be opened by any user except for the root user. However, this will not interfere with using this agent for the Load Generator machine.

2 In the Console, in the Generators > Load Generator Information > Unix Environment tab, check the **Don't use RSH** option. Then connect as usual.

3 To stop the agent daemon, run the following command the *<ProTune_root>/bin* directory: m_daemon_setup -remove

This stops the m_agent_daemon, and if successful you will receive a message: m_agent_daemon removed successfully.

- ➤ If no daemon of this name is being run by this user, you will receive the following warning:
  WARNING: Could not remove m_agent_daemon, reason - user <user_name> is not running m_agent_daemon on this machine.

- ➤ If an error occurred, you will receive the following error message:
  ERROR: Could not remove m_agent_daemon. Check log file m_agent_daemon[xxx].log in your temp directory.

Failure to Connect to the AUT Database

If you are running a database application, you must ensure that all remote clients can connect with the database server. If network or configuration errors occur when the client accesses the server, you must correct them before running a session. To ensure that your client application can connect with the database server, perform the following tests.

- ➤ Ping
- ➤ SQL utilities

Ping: Ensure that the client can communicate with the database server using TCP/IP. Use a ping utility or type ping <server_name> from the DOS command line.

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

Failure to Access Files

A ProTune session will fail if the result path or script is inaccessible to one or more of the participating machines. Check the following items:

- ➤ Path Translation
- ➤ Script
- ➤ Result Path

Path Translation: A script's location (path) is always based on the Console machine's mapping of that location. If a Vuser load generator maps to the script's path using a different name, path translation is required. Path translation translates the Console's mapping of a given location to the Vuser load generator's mapping. For example, if one machine maps the script directory as *g:\test*, while another maps it as *h:\test*, the paths should be translated.

Path translation is also effective across platforms—between Windows and UNIX. You use path translation to translate the Windows Console paths into paths recognized by UNIX.

Note: Path translation is only required if you chose to save all scripts and results to a shared network drive. In the default setup, ProTune saves files locally and collates them to the Console machine; no path translation is required.

Suppose that your script is in the /usr/jon/lr_test1 directory and runs on the UNIX machine, *sunny*. To translate it from the Windows Console machine, *pc1*, where your UNIX directory is mapped as *r*, enter the following line in the path translation table:

```
pc1              r:\              /usr/jon       sunny
```

To translate the f:\qa Console directory to all load generator machines running */m/qa/lr_test2/lr_test2.usr* on a UNIX platform, type:

```
win              f:\qa            /m/qa          UNIX
```

If the paths are not translated properly, the session will fail. For more information about path translation, see Appendix B, “Performing Path Translation.”

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

Result Path: Make sure that the result path is accessible to all load generators participating in the session through path translation and permissions. Check the permissions of the result directory files and modify them if necessary.

Failed Vusers or Transactions

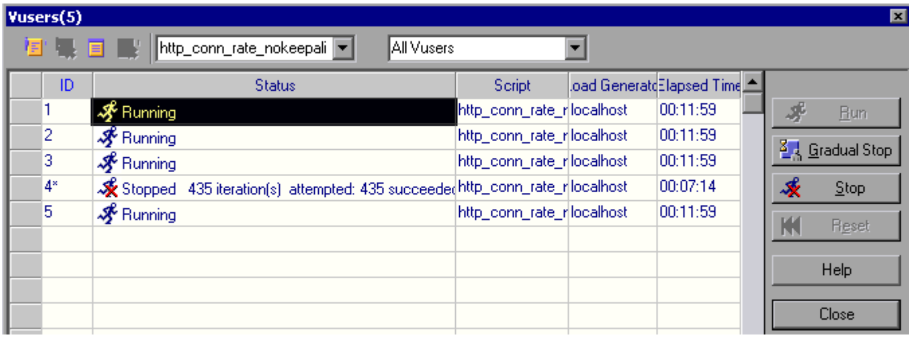
ProTune Vusers or transactions may fail for a variety of reasons relating to the network, database, or actual script. You can find information about session runs from the following sources:

- Run View
- Output Window
- Output File (excluding GUI Vusers)
- Analysis Reports and Graphs

Run View

The Run view is part of the ProTune Console. The Session Groups window in the top left-hand corner of the view indicates the status of the Vuser groups during and after the session run. During the session run, the columns will show a PENDING, INITIALIZING, READY, RUNNING, or RENDEZVOUS status. You can also view the status of each individual Vuser in the Vusers dialog box. If a Vuser fails and does not complete the script execution, ProTune displays an error status. If a Vuser completes the script execution, ProTune indicates the transaction status of a completed script run using the DONE.FAILED or DONE.PASSED status.

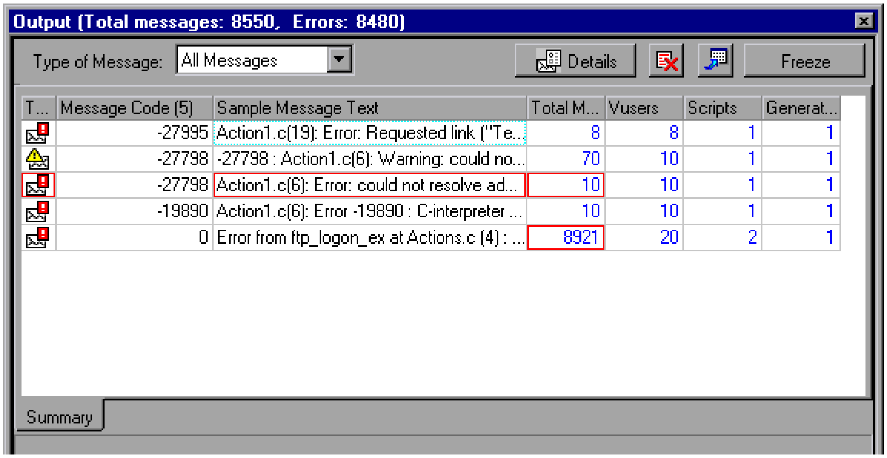
For more information about the Vuser states, see Chapter 10, "Running a Session."



ID	Status	Script	.oad Generatc	Elapsed Time
1	Running	http_conn_rate_r	localhost	00:11:59
2	Running	http_conn_rate_r	localhost	00:11:59
3	Running	http_conn_rate_r	localhost	00:11:59
4*	Stopped 435 iteration(s) attempted: 435 succeedec	http_conn_rate_r	localhost	00:07:14
5	Running	http_conn_rate_r	localhost	00:11:59

Output Window

View the Output window from the Console. The output window contains useful information for debugging a session. The output window lists five types of messages: errors, warnings, notifications, debug, and batch. An error message usually results in a failed script. A warning message indicates that the Vuser encountered a problem, but test execution continued. A notification provides useful information such as recorded think time values and other run-time information. A debug message is sent if you enable the debugging feature in **Tools** > **Options** > **Debug Information** (Expert Mode). Batch messages are sent instead of message boxes appearing in the Console, if you are using automation.



Output (Total messages: 8550, Errors: 8480)

Type of Message: All Messages

T...	Message Code (5)	Sample Message Text	Total M...	Vusers	Scripts	Generat...
❗	-27995	Action1.c(19): Error: Requested link ("Te...	8	8	1	1
⚠	-27798	-27798 : Action1.c(6): Warning: could no...	70	10	1	1
❗	-27798	Action1.c(6): Error: could not resolve ad...	10	10	1	1
❗	-19890	Action1.c(6): Error -19890 : C-interpreter ...	10	10	1	1
❗	0	Error from ftp_logon_ex at Actions.c (4) : ...	8921	20	2	1

For more information about the Output window, see Chapter 11, "Viewing Vusers During Execution."

Output File

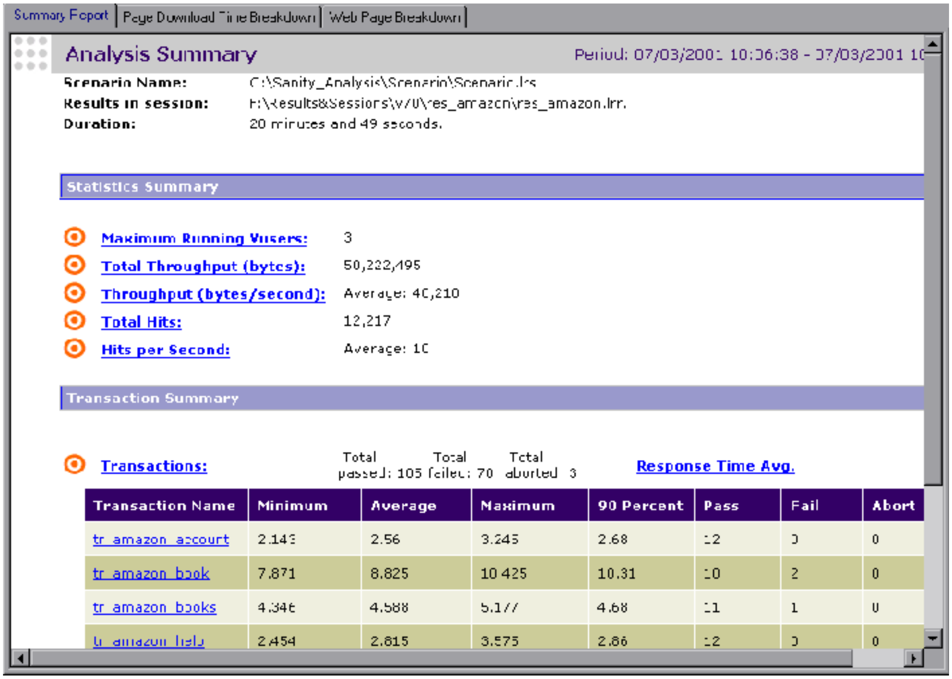
You can view information about script execution in an output file located in the Vuser result directory. The output file, *output.txt*, contains:

- a list of the primary functions called during the session
- error messages from the database server
- transactions and rendezvous information

The extent of the information sent to the output file depends on the output file settings. In the VuGen's run-time settings, you specify a Brief or Extended log. For the Extended log, you can specify a full trace, returned data, or current parameter value. An extended log is helpful for debugging a script, but if you are not debugging, Extended log is not recommended as it introduces extra overhead. For more information about configuring run-time settings, refer to the *ProTune Creating Virtual User Scripts* guide.

Analysis Reports and Graphs

You can generate graphs and reports to view information about the session run. For example, the Session Summary report displays a table containing the session's run-time data and provides links to the following graphs: Running Vusers, Throughput (Web), Hits Per Second (Web), HTTP Responses per Second, Transaction Summary, and Average Transaction Response Time.



Analysis Summary   Period: 07/03/2001 10:36:38 - 07/03/2001 10

Scenario Name:	C:\Sanity_Analysis\Scenario\Scenario.lrs
Results in session:	F:\Results&Sessions\v70\res_amazon\res_amazon.lrr.
Duration:	20 minutes and 49 seconds.

Statistics Summary

Maximum Running Vusers:	3
Total Throughput (bytes):	50,222,495
Throughput (bytes/second):	Average: 40,210
Total Hits:	12,217
Hits per Second:	Average: 10

Transaction Summary

Transactions: Total passed: 105 Total failed: 70 Total aborted 3   Response Time Avg.

Transaction Name	Minimum	Average	Maximum	90 Percent	Pass	Fail	Abort
tr_amazon_account	2.143	2.56	3.245	2.68	12	0	0
tr_amazon_book	7.871	8.825	10.425	10.31	10	2	0
tr_amazon_books	4.346	4.588	5.177	4.68	11	1	0
tr_amazon_help	2.454	2.815	3.575	2.86	12	0	0

For more information on the available graphs and reports, see the *ProTune Analysis User's Guide*.

Increasing the Number of Vusers on a Windows Machine

Under the normal settings of a Windows machine, you are limited to several hundred Vusers. This limitation is related to the operating system and not to the CPU or memory.

To work around the limitation of the Windows operating system, modify the Windows Kernel as follows:

1. Save a copy of the registry file in case you have trouble with these modifications.
2. Run Regedit.
3. Go to following key in KEY_LOCAL_MACHINE:

   ```
   System\CurrentControlSet\Control\Session Manager\SubSystems
   ```

4. Select the Windows key. The default Windows key for NT 4.0 looks like this:

   ```
   %SystemRoot%\system32\csrss.exe ObjectDirectory=\Windows
   SharedSection=1024,3072
   Windows=On SubSystemType=Windows ServerDll=basesrv,1
   ServerDll=winsrv:UserServerDllInitialization,3
   ServerDll=winsrv:ConServerDllInitialization,2
   ProfileControl=Off MaxRequestThreads=16
   ```

   The SharedSection=1024,3072 key has the format xxxx,yyyy where:

   xxxx defines the maximum size of the system-wide heap (in kilobytes)

   yyyy defines the size of the per desktop heap.

5. Increase the SharedSection parameter by changing the yyyy settings from 3072 to 8192 (which is 8 MB).

   This setup successfully allowed 1250 Oracle Vusers to run on a Windows machine using 2 Pentium PRO 200 MHz with 1 GB RAM.

   Each Vuser in this setup used approximately 2MB memory. Other Vusers may require more memory.

ProTune was able to load over 2500 Vusers when the Windows terminal server was run as the Operating System and the above registry setting was changed.

The above registry changes enable you to run more threads, allowing you to run more Vusers on the machine. This implies that you are not bound by the Windows operating system; you are only bound by hardware and internal scalability limitations.

Troubleshooting Firewalls

There are three log files which provide additional information about activity over the firewall.

The **ProTune agent log file** contains information about communication activity between the ProTune agent and the MI Listener.

- ➤ To open the file on Windows machines, right-click the ProTune agent icon in the system tray of the ProTune agent machine, and select **View Log**. Alternatively, open the latest *<temp_directory>\ProTune_agent_startup<unique identifier>.log* file (if the ProTune agent is a process), or *<temp_directory>\ProTune_agent_service<unique identifier>.log* file (if the ProTune agent is a service), in a text editor.
- ➤ To open the file on UNIX machines, open the *<temp_directory>/m_agent_daemon<unique identifier>.log* file in a text editor.
- ➤ To increase the logging level, select Agent Settings from Start->Programs->ProTune->Advanced Settings (or open file <ProTune_root>\launch_service\dat\br_lnch_server.cfg in a text editor), and in the Log section, set AgentExtended to 1.

The **MI Listener log file** contains information about MI Listener communication with the ProTune agent and the Console.

To open the file, right-click the MI Listener Agent icon in the system tray of the MI Listener machine, and select **View Log**. Alternatively, open the latest *<temp_directory>**ProTune_agent_startup<unique identifier>.log* file (if the ProTune agent is a process), or *<temp_directory>**ProTune_agent_service<unique identifier>.log* file (if the ProTune agent is a service), in a text editor.

To increase the logging level, select **Start** > **Programs** > **ProTune** > **Advanced Settings** > **Agent Settings**, or open the *<ProTune_root>**launch_service**dat**br_lnch_server.cfg* file in a text editor. In the Log section, set AgentExtended to 1.

The **Console log file** contains information about communication activity between the Console and the MI Listener.

To open the file on Windows machines, open the *<temp_directory>**drv_log.txt* file in a text editor.

Verifying Connection Between ProTune Agent and MI Listener

If there is a proper connection between the ProTune agent and the MI Listener:

➤ On Windows platforms, the agent icon's light in the system tray will turn from red to green.

➤ On UNIX platforms, a file called *<Local_machine_key>_connected_to_MI_Listener* will be created in the temporary directory of the ProTune agent machine. Local_machine_key is the value set in the Agent Configuration, as described in Chapter 12, "Working with Firewalls." The file will be removed when the ProTune agent disconnects from the MI Listener.

➤ On both UNIX and Windows platforms, the following message will appear in the ProTune agent log file: Notify Connected to MI Listener.

Note: The ProTune agent tries to connect to the MI Listener machine every Timeout seconds (as defined in the Agent Configuration). After a successful connection, if no Console has connected through this MI Listener to the agent after another Timeout, the ProTune agent will disconnect from the Console.
On a Windows machine, the agent icon's light in the system tray will turn from green to red. On UNIX machines, the file <Local_machine_key>_connected_to_MI_Listener will be removed from the temporary directory in the ProTune agent machine.
In both Windows and UNIX, the message Disconnected from MI Listener will appear in the ProTune agent log file.

UNIX Connection Errors

After installing the *m_agent_daemon* as described in Chapter 12, "Working with Firewalls," you should receive a message: m_agent_daemon installed successfully.

Agent Daemon Errors

ERROR: File m_agent_daemon doesn't exist.

This error means that you are not in the same directory as the file (meaning not in <ProTune_root>/bin directory, or the file really doesn't exist, which indicates a problem with the installation).

WARNING: Could not install m_agent_daemon, reason - user <user_name> is already running m_agent_daemon on this machine.

This warning message occurs when a daemon of this name is already being run by the same user.

ERROR: Could not install m_agent_daemon. Check log file m_agent_daemon[xxx].log in your temp directory.

This error indicates that some error has occurred when loading the daemon. You should check the log file and consult the following troubleshooting tips.

ProTune Agent Log File Errors

Error - 10344 : Communication Error: -59961 : Failed to bind a socket while calling bind function.

Error -10344 : Communication Error: -59927 : Failed to create a TCP server for the HTTP channel's server.

Warning -29974 : Failed to create "router" server.

These messages appear because the ProTune agent always tries to open port number 443 (because any agent can be a MI Listener, and the MI Listener always listens to this port), and in UNIX machines, this port cannot be opened by any user except for the root user. However, this will not interfere with using this agent for the Load Generator machine.

Error -10343 : Communication error : -59981 : Failed to connect to remote host - <MI_Listener_name>.

The MI Listener is not being run at the time of the connection attempt on the machine set in MI Listener Name in the Agent Configuration.

Error -10343 : Communication error: -59928 : Unresolved server name .

The name passed in MI Listener Name in the Agent Configuration is not a name, full name or IP address of a valid machine, or no value was set.

Error -10343 : Communication error: -59928 : Unresolved server name .

The name passed in Proxy Name in the Agent Configuration is not a name, full name or IP address of a valid machine.

Error -10343 : Communication error: -59945 : Client failed to connect to a PROXY Server with the following settings: (-server_port=<proxy_server_port>)(-server_fd_primary=2)(-server_type=8)(-allowed_msg_size=0)(-allowed_msgs_num=0)(-proxy_configuration_on)(-tcp_tunnel_configuration_on).

The Proxy Name field is empty.

Error -10343 : Communication error: -59982 : Failed to connect to remote host - <MI_Listener_Name>. The remote address is not a valid address.

Error -10343 : Communication error: -59945 : Client failed to connect to a PROXY Server with the following settings:
(-server_name=<proxy_server_name>)(-server_port=<proxy_server_port>)(-server_fd_primary=2)(-server_type=8)(-allowed_msg_size=0)(-allowed_msgs_num=0)(-proxy_configuration_on)(-tcp_tunnel_configuration_on).

The Proxy Port set in Agent Configuration, has been set to the wrong port number.

Error -10343 : Communication error: -59913 : NTLM authentication to proxy server error - connection to proxy refused.

The proxy server is configured in for NTLM authentication and the Proxy User Name, Proxy Password and/or Proxy Domain are not set correctly in the Agent Configuration.

Error -10343 : Communication error: - 59880 : Basic authentication to proxy server error - connection to proxy refused.

The proxy server is configured in for Basic authentication and the Proxy User Name and/or Proxy Password are not set correctly in the Agent Configuration.

Error -10343 : Communication error: -59907 : SSL connect error : verify host failed : wrong DNS test .

This error occurs when you have set the Check Server Certificates setting to True, and have not issued a new certificate to the MI Listener machine (see Appendix E, "Working with Digital Certificates" for more details).

Error -10343 : Communication error: -59907 : SSL connect error : certificate verify failed.

Error -10343 : Communication error: -59907 : SSL connect error : sslv3 alert handshake failure.

Error -10343 : Communication error: -59907 : SSL connect error : sslv3 alert bad certificate.

Error -10343 : Communication error: -59907 : SSL connect error : sslv3 alert certificate expired.

These errors occur when you set the Check Server Certificates setting to True. See Appendix E, “Working with Digital Certificates” to learn how to issue a valid certificate.

Error -10343 : Communication error: -59910 : SSL initialization error : Certificate not found .

Error -10343 : Communication error: -59910 : SSL initialization error : No such file or directory.

Error -10343 : Communication error: -59910 : SSL initialization error : system lib.

These errors occur when the Client Certificate owner setting in the Agent Configuration is set to True, but no certificate was installed in the ProTune agent machine (see Appendix E, “Working with Digital Certificates” for more details).

MI Listener Log File Errors

Error - 10344 : Communication Error: -59961 : Failed to bind a socket while calling bind function.

Error -10344 : Communication Error: -59927 : Failed to create a TCP server for the HTTP channel’s server.

Warning -29974 : Failed to create "router" server.

This error means that another process on the MI Listener machine is occupying port 443 (for instance the IIS service).

Error -10343 : Communication error: -59904 : SSL accept error : sslv3 alert certificate expired.

These errors occur when you have set the Check Server Certificates setting to True, and the MI Listener's certificate is expired.

Error -10343 : Communication error: -59904 : SSL accept error : sslv3 alert bad certificate.

These errors occur when you have set the Check Server Certificates setting to True, and either:

- The MI Listener's certificate does not have a signature that is included in the ProTune agent's CA List.
- The MI Listener's certificate has a future verification date.

See Appendix E, "Working with Digital Certificates" to learn how to issue a valid certificate and how to add a Certification Authority to a CA list, or how to create a certificate with a new validation date.

Error -10343 : Communication error: -59904 : SSL accept error : peer did not return a certificate.

These errors indicate that the Check Client Certificates setting in the MI Listener Configuration is set to True, but the Client Certificate owner setting in the Agent Configuration is set to False.

Error -10343 : Communication error: -59904 : SSL accept error : no certificate returned.

These errors indicate that the Check Client Certificates setting in the MI Listener Configuration is set to True, and the Client Certificate owner setting in the Agent Configuration is set to True, but either:

- The ProTune agent's certificate does not have a signature that is included in the MI Listener's CA List.
- The ProTune agent's certificate has a future verification date.

See Appendix E, "Working with Digital Certificates" to learn how to issue a valid certificate and how to add a Certification Authority to a CA list, or how to create a certificate with a new validation date.

Error -10343 : Communication error: -59904 : SSL accept error : no certificate returned.

These errors indicate that the Check Client Certificates setting in the MI Listener Configuration is set to True, and the Client Certificate Owner setting in the Agent Configuration is set to True, but the ProTune agent's certificate has expired.

General Connection Errors

These errors can occur when using all configurations.

If no errors appear both in the ProTune agent log, and the MI Listener log, but the agent does not connect to the MI Listener, make sure that the FireWallServiceActive attribute in the Firewall section in the *<ProTune_Installation>\dat\br_lnch_server.cfg* file on the ProTune agent machine, is set to 1.

Verifying Connection Between the Console and Agent through the MI Listener

When there is a successful connection between the ProTune agent and the MI Listener, and the Console machine fails to connect, you should check the following:

- The **Name** field in the Load Generators dialog in the Console should match the name set in the **Local Machine Key** in the Agent Configuration.
- The **MI Listener** field in the **Load Generators** > **Details** > **Firewall** tab of the above host matches the name set in the **MI Listener Name** in the Agent Configuration.
- In the Tools menu of the Console, in the **Options** > **Timeout** tab, the **Load Generator Connect timeout** might need to be increased, because the Firewalls may slow down the communication.
- Make sure that the Console machine recognizes the ProTune agent machine (e.g., by using the ping utility). If this fails, there is a configuration problem in the system not related to ProTune, and it must be solved before the connection can be made.
- Make sure that the Console has successfully connected to the MI Listener by checking port 50500 on the MI Listener machine (you can use the netstat utility, on the MI Listener machine).

B

Performing Path Translation

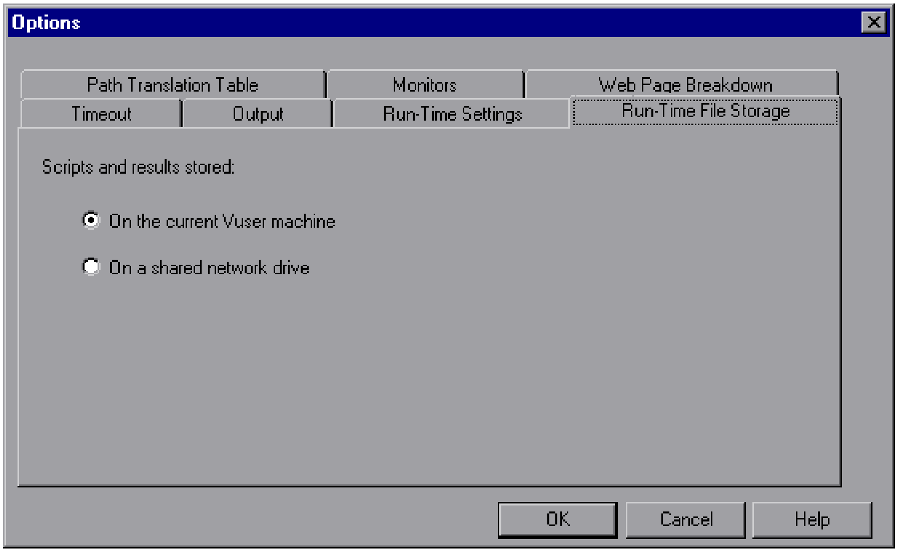
When you run a session, ProTune gathers run-time data from the participating Vusers. By default, ProTune stores the data in temporary files on each Vuser machine. After the session, the data is collated in the general results directory.

Alternatively, you can instruct ProTune to write the run-time data directly to a shared network drive. (See Chapter 6, "Configuring Session Steps.") This method is not recommended, since it increases network traffic and necessitates path translation.

Understanding Path Translation

Path Translation is a mechanism used by ProTune to convert a remote path name for the Console. A typical session might have the ProTune Console running on a Windows-based machine and include multiple Vusers running on both Windows-based and UNIX load generators. One remote load generator may map the network drive as *F*, while another load generator maps the same drive as *H*. In a complex session such as this, you need to ensure that all participating machines recognize the same network drive.

You instruct ProTune to store scripts and run-time data results on a shared network drive from the Run-time File Storage tab of the Options dialog box.



Scripts and results stored:

● On the current Vuser machine

○ On a shared network drive

Result and script files stored on a shared network drive require you to perform path translation.

The Script view contains a list of all the scripts associated with a session—and their locations. A script's location (path) is always based on the Console machine's mapping of that location. If a Vuser load generator maps to the script's path using a different name, path translation is required.

For example, assume that the Console is running on a Windows-based machine named *pc2*, and that a script is located on a network drive. The Console machine maps the network drive as *m*:\lr_tests. If the remote Vuser machine (load generator) hosting the Vusers also maps the path as *m*:\lr_tests, no translation is necessary. However, if the remote machine maps the path as another drive or path, for example *r*:\lr_tests, you must translate the path to enable the load generator to recognize the script location.

Similarly, when saving run-time result files to a shared drive that is mapped differently by the Console and remote load generator, you must perform path translation.

Path translation is also effective across platforms—between Windows and UNIX. You use path translation to translate Windows-based paths (as seen by the Console) into paths recognized by the UNIX Vuser load generator.

Adding Entries to the Path Translation Table

To translate a path from one Windows-based computer to another, or between Windows-based and UNIX machines, you create an entry in the Path Translation table. This table contains a list of paths translated into formats that can be recognized by different machines.

Each line of the Path Translation table has the following format:

```
<console_host><console_path><remote_path>[<remote_host>]
```

console_host	The name or type of the machine that is running the Console. For example, if the Console is running on a Windows-based computer, you could type win in the host field. Alternatively, you could enter the name of the machine running the Console (for example, LOADPC1).

The value of *console_host* can be:

hostname	the name of the machine running the Console
win	the Console is running on a Windows-based computer
unix	the Console is running on a UNIX machine
all	the Console is running on a Windows-based or a UNIX machine

console_path

The path of a specific directory—as recognized by the Console. For example, if the directory *scripts* is located on the network drive *r*—as mapped by the Console—type the path r:\scripts in the *console_path* field.

remote_path

The path of a specific directory—as recognized by the remote machine. For example, if the directory *scripts* is located on the network drive *n—as mapped by the remote load generator*—type the path n:\scripts in the *remote_path* field.

If a Vuser on the remote UNIX load generator recognizes the above path as /m/tests, you would type this path in the *remote_path* field.

remote_host

The name or type of the remote load generator. For example, if all the remote machines are UNIX workstations, you could type unix in the *remote_host* field. The options for the *remote_host* field are the same as the options for the *console_host* field, listed above. The *remote_host* parameter is optional.

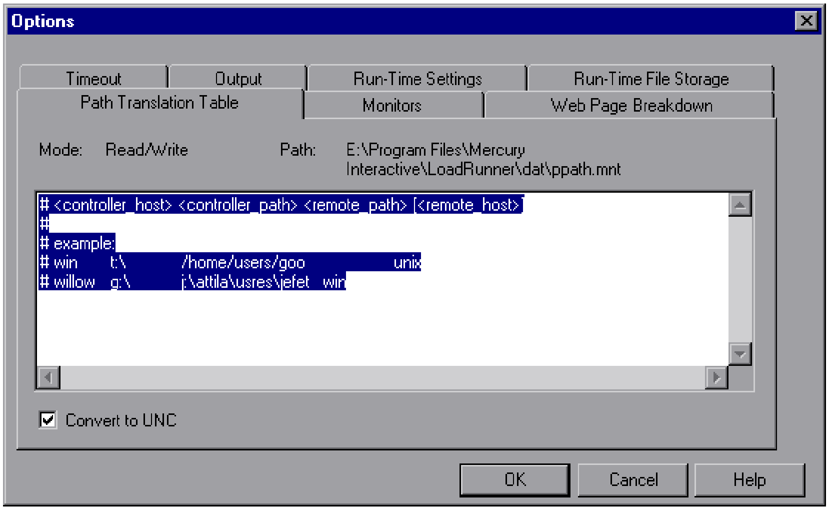
Editing the Path Translation Table

You maintain the Path Translation table using the ProTune Console. ProTune saves the Path Translation table as an ASCII file, *ppath.mnt.* This file, stored in ProTune_directory/dat, has a one–line entry for each network path to translate.

To edit the Path Translation table:

1 Start the ProTune Console.

2 Choose **Tools** > **Options** and select the **Path Translation Table** tab. The Path Translation Table view opens.



Mode: Read/Write

Path: E:\Program Files\Mercury Interactive\LoadRunner\dat\ppath.mnt

```
# <controller_host> <controller_path> <remote_path> [<remote_host>]
#
# example:
# win      t:\          /home/users/goo             unix
# willow   g:\          j:\attila\usres\jefet   win
```

☑ Convert to UNC

3 Before you enter path translation information, consider using the Universal Naming Convention method. If your machines are Windows machines, you can tell the Console to convert all paths to UNC, and all machines will be able to recognize the path without requiring path translation. An example of UNC format is \\machine_a\results.

Select the Convert to UNC check box to tell ProTune to ignore the path translation table and to convert all paths to the Universal Naming Convention.

4 If your machines are not Windows machines and you require path translation, type the path information into the table. You can insert comments by typing the “#” symbol at the start of a line in the table.

5 Click **OK** to close the table and save the information.

Path Translation Examples

The following section illustrates sample Path Translation Table entries.

Note that when you translate a Windows-based path to a UNIX path, you must enter the appropriate slashes—forward slashes for UNIX and back slashes for Windows-based paths.

The examples below show the use of the Path Translation table for a Windows-based Console called Merlin.

In the first example, Vusers are running on a Windows 2000 machine, Oasis. Merlin maps the network drive as f:, while Oasis maps it as g:\loadtest.

merlin	f:\	g:\loadtest\	Oasis

In the second example, Vusers are running on a UNIX machine, Ultra. Ultra maps the networks drive as /u/tests/load.

merlin	f:\	/u/tests/load/	Ultra

In the third example, the mapping of the network drive by the remote load generator Jaguar, is identical to the Console's mapping, so no translation is required. This line can be excluded from the Path Translation table.

merlin	n:\	n:\	Jaguar

In the fourth example, all Windows-based Vuser load generators map the network drive as m:\loadtest.

merlin	l:\mnt\	m:\loadtest\	win

C

Working in Expert Mode

Advanced users can fine-tune the ProTune configuration settings while working in *Expert Mode*. In Expert mode, additional options are displayed in the Options dialog box and in the Load Generator Information dialog box. This appendix describes the additional settings that are available in the Expert mode:

- Entering Expert Mode
- Options - Agent Settings
- Options - General Settings
- Options - Debug Information Settings
- Options - Output Settings
- Options - Monitor Settings
- Load Generator Information - UNIX Environment Settings
- Load Generator Information - Connection Log Settings

Entering Expert Mode

The ProTune Console Expert mode is intended for support personnel to provide access to system information. When you work in the Expert mode, the Console dialog boxes contain additional options for fine tuning the Console operation.

To activate the Expert mode, choose **Tools** > **Expert Mode.** An active Expert mode is indicated by a check mark.

To exit the Expert mode, repeat the above process.

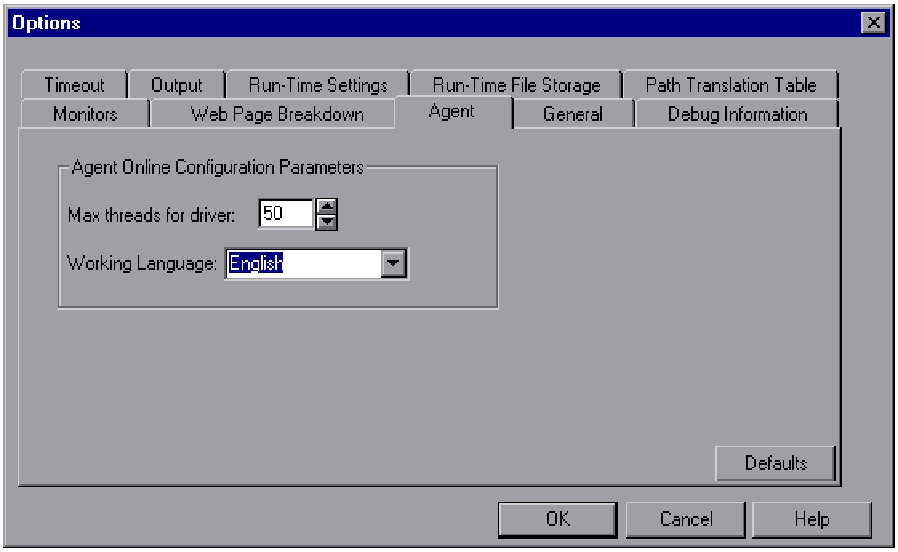
Options - Agent Settings

The Agent settings allow you to customize the behavior of the ProTune agent on a remote load generator machine. Using the Options dialog box, you set the online configuration parameters for the agent.

To set the Agent settings:

1 Enter Expert mode (see above).

2 Choose **Tools** > **Options**. The Options dialog box appears. Select the **Agent** tab.



Agent Online Configuration Parameters

Max threads for driver: 50

Working Language: English

3 Select the maximum number of threads to be executed for the current Vuser's driver.

4 Select the agent's working language (English or Japanese).

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

Options - General Settings

The General tab in the Options dialog box allows you to specify global settings for data table storage and multiple IP address allocation, and instruct ProTune not to collate log files.

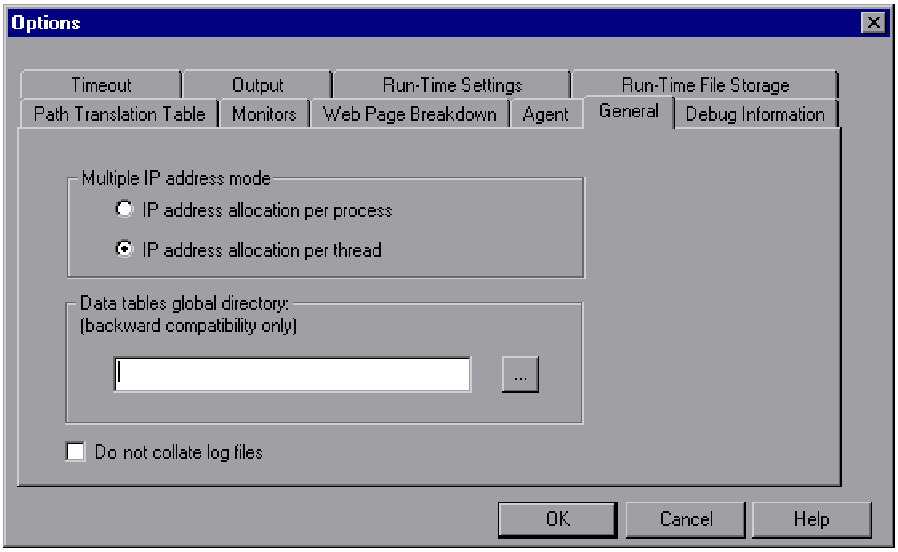
Multiple IP address mode: The mode used to allocate IP addresses when the multiple IP address option is enabled (**Session** > **Enable IP Spoofer**). The Console can allocate an IP address per process or per thread. Web Vusers require IP address allocation per process. WinSock Vuser IP addresses can be allocated per thread or per process. Allocation per thread results in a more varied range of IP addresses in a session.

Data tables global directory: The network location for data tables used as a source for parameter values. This setting is only required for scripts created with earlier versions of ProTune.

Do not collate log files: Instructs ProTune to collate only result files, and not log files.

To set the General Expert mode settings:

1 Choose **Tools** > **Options**. The Options dialog box appears. Select the **General** tab.



Multiple IP address mode

○ IP address allocation per process

● IP address allocation per thread

Data tables global directory:
(backward compatibility only)

□ Do not collate log files

2 Select the Multiple IP address mode.

3 Enter the global directory for data tables.

4 If you want ProTune to collate only result files and not log files, check **Do not collate log files**.

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

Options - Debug Information Settings

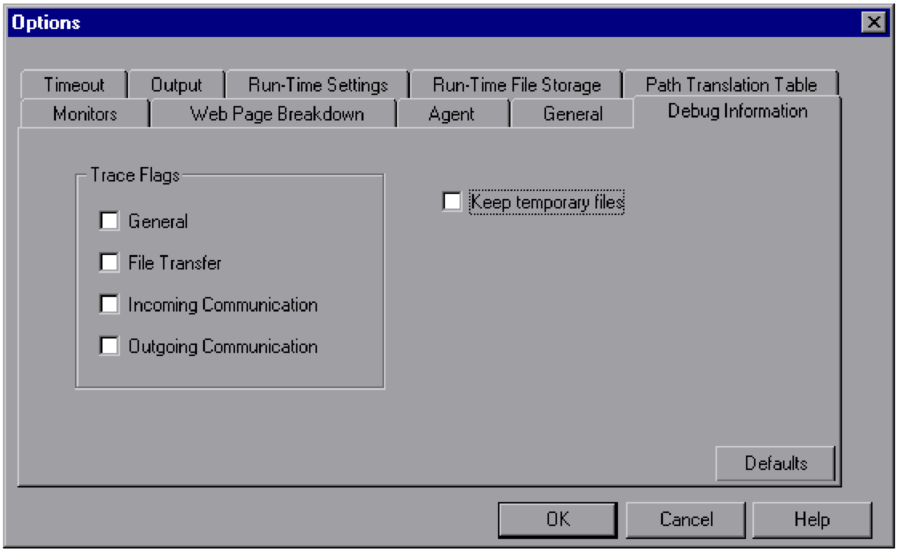
The Debug settings in the Options dialog box allow you to determine the extent of the trace to be performed during session execution. The debug information is written to the Output window.

The following trace flags are available: General, File Transfer, Incoming Communication, and Outgoing Communication. You only need to select the flags relating to your problem. For example, if you encounter specific problems with the transfer of files, select the File Transfer flag.

The ProTune agent and Console create some temporary files, which collect information such as the parameter file sent to the Vuser, the output compilation file, and the configuration file. The ProTune agent files are saved in *brr* folders in the TMP or TEMP directory of the agent machine. The Console files are saved in *lrr* folders in the TMP or TEMP directory of the Console machine. At the end of the session, all these files are automatically deleted. However, using the Debug Information Expert mode settings, you can instruct ProTune to keep these temporary files.

To set the Debug Information settings:

1 Choose **Tools** > **Options**. The Options dialog box appears. Select the **Debug Information** tab.



Trace Flags

□ General

□ File Transfer

□ Incoming Communication

□ Outgoing Communication

□ Keep temporary files

2 Select the check boxes for the desired trace flags.

3 To save the temporary run-time files, select the **Keep temporary files** check box.

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

Options - Output Settings

Basic Output settings are available when Expert mode is disabled. However, Expert mode provides additional settings for handling Vuser output:

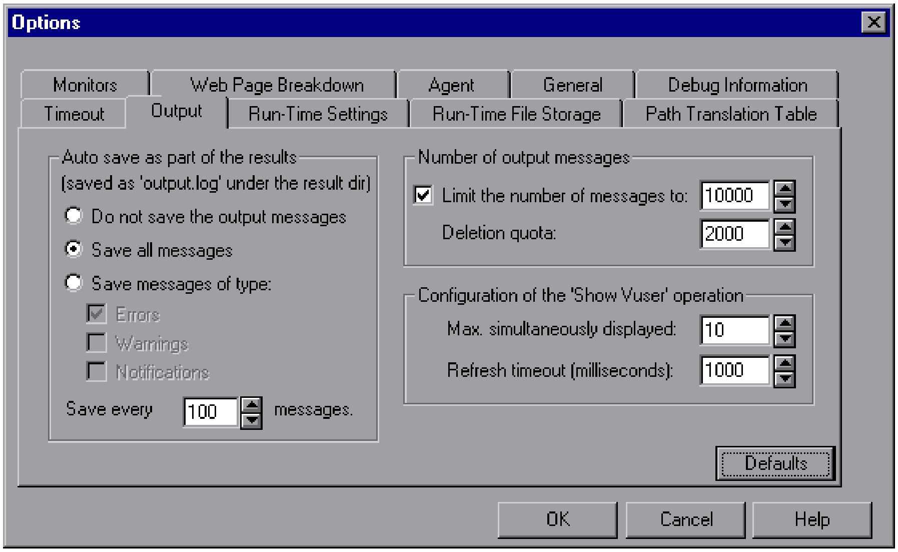
Save every *xxx* messages: Defines how often to auto-save the output message file. The default is every 100 messages.

Max simultaneously displayed: Specifies the maximum number of Vuser logs that may be displayed simultaneously, as well as the maximum number of active UNIX, GUI, RTE, or Web Vusers that the Console should display by opening up Run-Time Viewers on your machine. The default number is 10.

Refresh timeout: Defines how often to refresh the Vuser log. The default is every 1000 milliseconds.

To set the Output settings:

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



Options

Auto save as part of the results
(saved as 'output.log' under the result dir)

○ Do not save the output messages
● Save all messages
○ Save messages of type:
☑ Errors
☐ Warnings
☐ Notifications

Save every 100 messages.

Number of output messages

☑ Limit the number of messages to: 10000
Deletion quota: 2000

Configuration of the 'Show Vuser' operation

Max. simultaneously displayed: 10
Refresh timeout (milliseconds): 1000

2 Enter the number of messages after which ProTune will perform an auto-save operation.

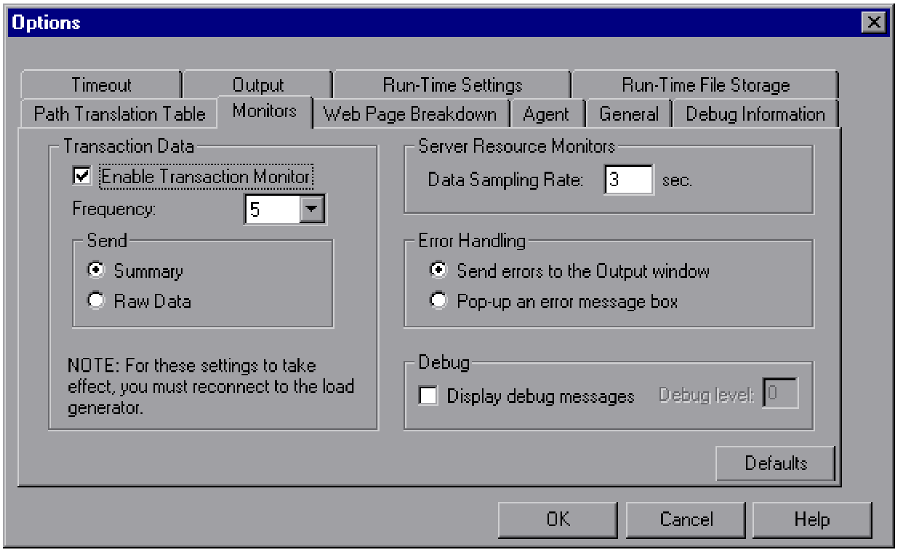
3 Enter values for the relevant Show Vuser options.

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

Options - Monitor Settings

Expert mode provides the following additional monitor setting:

Send Summary or Raw Data: sends a summary of the data collected back to the Console, or sends all of the data in raw form. Sending the data in raw form saves time because the data does not need to be processed. However, since all of the data is being transferred to the Console, it may cause more network traffic. If the transfer speed is significant to you, it is recommended that you choose **Summary**.



Options

Transaction Data

☑ Enable Transaction Monitor

Frequency: 5

Send

● Summary

○ Raw Data

NOTE: For these settings to take effect, you must reconnect to the load generator.

Server Resource Monitors

Data Sampling Rate: 3 sec.

Error Handling

● Send errors to the Output window

○ Pop-up an error message box

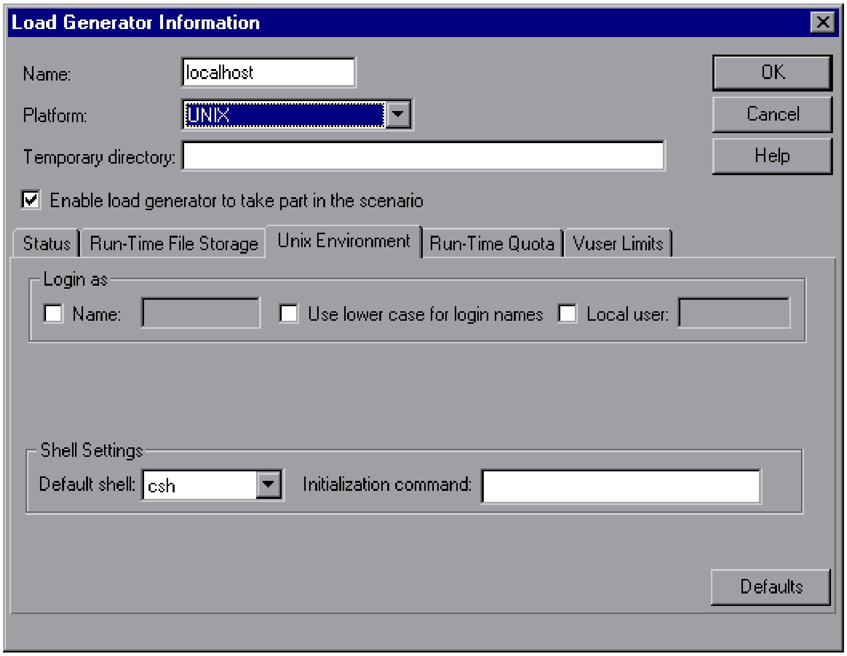
Debug

□ Display debug messages Debug level: 0

Load Generator Information - UNIX Environment Settings

Expert mode provides the following additional UNIX Environment setting:

Local User: UNIX load generators that use the *rsh* shell establish a connection as the current NT user (due to security considerations). To "mislead" rsh and log in as a user other than the current NT login, select the **Local user** check box and specify the desired UNIX login name. Since modifying the local user name is a security breach for *rsh*, this option should only be used when you encounter a problem connecting to the remote machine.



Load Generator Information

Name: localhost

Platform: UNIX

Temporary directory:

☑ Enable load generator to take part in the scenario

Login as

☐ Name:

☐ Use lower case for login names

☐ Local user:

Shell Settings

Default shell: csh

Initialization command:

Load Generator Information - Connection Log Settings

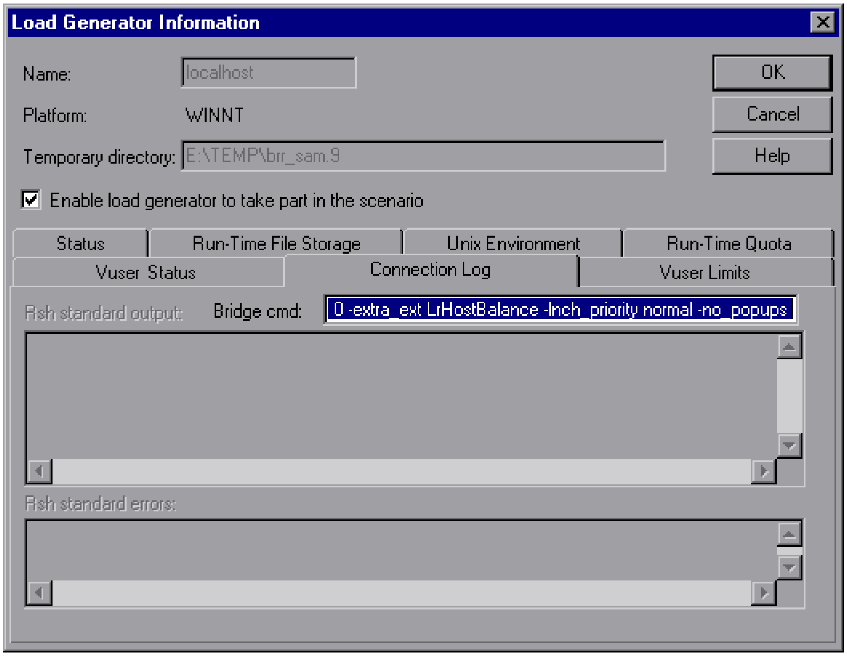
The Connection Log tab in the Load Generator dialog box allows you to view the standard output and standard errors generated as the Console connects to the selected UNIX load generator. You can also change the command that the Console sends to the remote bridge in order to connect to the load generator.

To set the Connection Log settings:

1 Click the **Generators** button, or select **Session** > **Load Generators**. The Load Generators dialog box opens.

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

3 Click the **Details** button. The Load Generator Information dialog box opens. Select the **Connection Log** tab.



Name: localhost

Platform: WINNT

Temporary directory: E:\TEMP\brr_sam.9

☑ Enable load generator to take part in the scenario

Bridge cmd: 0 -extra_ext LrHostBalance -lnch_priority normal -no_popups

You can view the rsh standard output and rsh standard errors generated as the Console sends the connection command to the selected UNIX load generator.

In the Bridge cmd box, enter a new command if you want to change the default bridge command being sent by the Console to the remote bridge in order to connect the UNIX load generator.

D

Working with Server Monitor Counters

When you configure the System Resource, Microsoft IIS, Microsoft ASP, ColdFusion, and SQL Server monitors, you are presented with a list of default counters that you can measure on the server you are monitoring. Using the procedure described below, you can create a new list of default counters by including additional counters, or deleting existing counters.

In addition, there are specific counters that are especially useful for determining server performance and isolating the cause of a bottleneck during an initial stress test on a server.

The following sections describe:

- Changing a Monitor's Default Counters
- Useful Counters for Stress Testing

Changing a Monitor's Default Counters

You can change the default counters for the System Resource, Microsoft IIS, Microsoft ASP, ColdFusion, or SQL Server monitors by editing the *res_mon.dft* file found in the ProTune/dat directory.

To change the default counters:

1. Open a new session and click the **Run** tab.
2. For each of the monitors, select the counters you want to measure.
3. Save the session and open the session *.lrs* file with an editor.

4 Copy the MonItemPlus section of the each counter you selected into the *res_mon.dft* file.

5 Count the number of new counters in the file and update the **ListCount** parameter with this number.

Useful Counters for Stress Testing

Certain counters are especially useful for determining server performance and isolating the cause of a bottleneck during an initial stress test on a server.

The following is a list of counters that are useful for monitoring Web server performance:

Object	Counter
Web Service	Maximum Connections
Web Service	Bytes Total/sec
Web Service	Current NonAnonymous Users
Web Service	Current Connections
Web Service	Not Found Errors
Active Server Pages	Requests/sec
Active Server Pages	Errors/sec
Active Server Pages	Requests Rejected
Active Server Pages	Request Not Found
Active Server Pages	Memory Allocated
Active Server Pages	Requests Queued
Active Server Pages	Errors During Script Run Time
Memory	Page Faults/sec
Server	Total Bytes/sec
Process	Private Bytes/Inetinfo

The following is a list of counters that are useful for monitoring SQL Server performance:

Object	**Counter**
SQLServer	User Connections
SQLServer	Cache Hit Ratio
SQLServer	Net-Network Reads/sec
SQLServer	I/O-Lazy Writes/sec
SQLServer-Locks	Total Blocking Locks
PhysicalDisk	Disk Queue Length

The following is a list of counters that are useful for monitoring both Web and SQL server performance:

Object	**Counter**
Processor	% Total Processor Time
PhysicalDisk	% Disk Time
Memory	Available Bytes
Memory	Pool Nonpaged Bytes
Memory	Pages/sec
Memory	Committed Bytes
System	Total Interrupts/sec
Object	Threads
Process	Private Bytes:_Total

Note: The % Disk Time counter requires that you run the diskperf -y utility at the command prompt and reboot your machine.

E

Working with Digital Certificates

A Digital Certificate is an electronic “credit card” that establishes your credentials when doing business or other transactions on the Web. It is issued by a Certification Authority (CA). It contains the IP address of the machine for which it was issued, a validation date, and the digital signature of the certificate-issuing authority.

This appendix describes:

- Using Digital Certificates with Firewalls
- Creating and Using Digital Certificates

Using Digital Certificates with Firewalls

When the MI Listener sends its Public Key to the ProTune agent, it always sends its certificate as well (this is the server-side certificate). The ProTune agent can be configured to authenticate the certificate which it received, as described in Chapter 12, “Working with Firewalls.” If the agent is configured to authenticate the certificate, it can verify whether the sender is really the machine that it claims to be by:

- Comparing the certificate's IP address with the sender’s IP address.
- Checking the validation date.
- Looking for the digital signature in its Certification Authorities list.

The MI Listener may also require the ProTune agent to send a certificate at any point in the session. This is called the client-side certificate, as described in the MI Listener Configuration Settings in Chapter 12, “Working with Firewalls.” If the ProTune agent owns a certificate, it sends it to the MI

Listener for the same authentication process. If the ProTune agent does not own a certificate, the communication might not be continued.

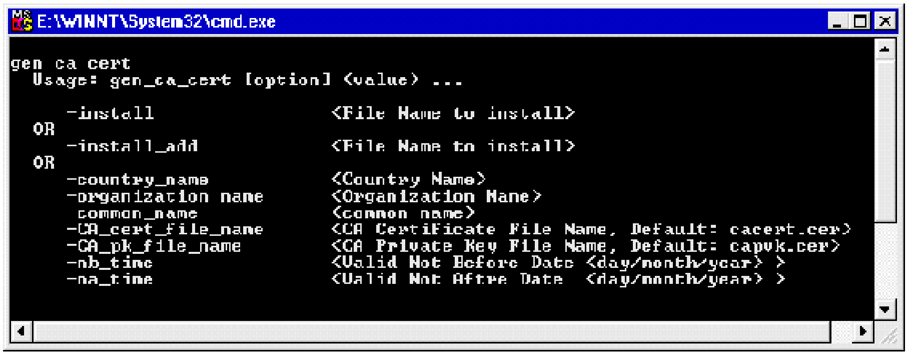
An SSL CA list and an SSL Certificate are included in each ProTune installation. This certificate is the same for all ProTune installations, which means that it can be obtained by third parties. Therefore, if you are interested in a more secure process, you should create your own Certificate Authority and include it in the list, and issue matching certificates for your machines.

Creating and Using Digital Certificates

You create a Certification Authority using the gen_ca_cert.exe (on UNIX platforms gen_ca_cert) utility, and a Digital Certificate using the gen_cert.exe (on UNIX platforms gen_cert) utility. Both utilities can be used on UNIX and Windows platforms, using a command-line interface.

To creating a Certificate Authority using gen_ca_cert:

1 To view the format and usage, run the *gen_ca_cert* utility from the <ProTune root folder>\launch_service\bin directory.



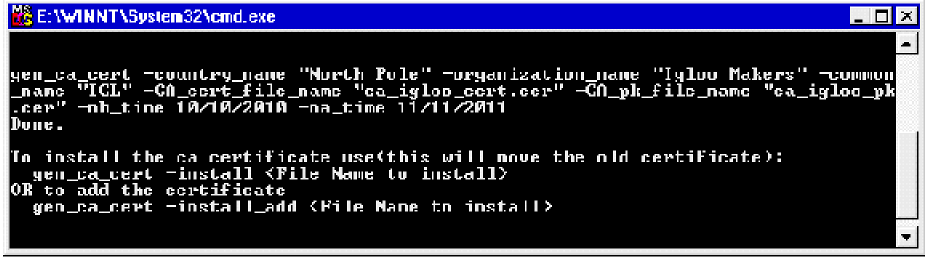
```
gen ca cert
  Usage: gen_ca_cert [option] <value> ...

     -install                  <File Name to install>
  OR
     -install_add              <File Name to install>
  OR
     -country_name             <Country Name>
     -organization name        <Organization Nane>
      common_name              <common name>
     -CA_cert_file_name        <CA Certificate File Name, Default: cacert.cer>
     -CA_pk_file_name          <CA Private Key File Name, Default: capvk.cer>
     -nb_time                  <Valid Not Before Date <day/month/year> >
     -na_time                  <Valid Not Aftre Date  <day/month/year> >
```

2 Create a new Certificate Authority by running the gen_ca_cert command with at least one of the options: -country_name <country name> -organization_name <organization name> and -common_name <the name of the CA>.

This process creates two files in the directory from which the utility was run: the CA Certificate (cacert.cer), and the CA Private Key (capvk.cer). To provide different file names, use the -CA_cert_file_name and the -CA_pk_file_name options respectively.

By default, the CA is valid for three years, from the time that the CA is generated. To change the validation dates, use the options -nb_time <beginning of validity in dd/mm/yyyy format> and/or -na_time <ending of validity in dd/mm/yyyy format>.

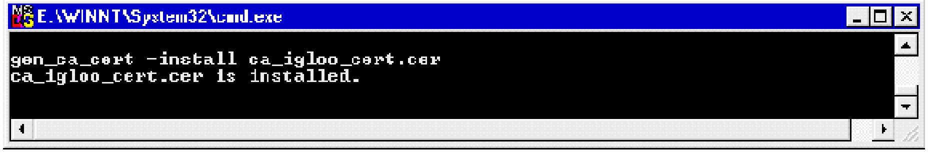
The following example creates two files: *ca_igloo_cert.cer* and *ca_igloo_pk.cer* in the current directory:



```
gen_ca_cert -country_name "North Pole" -organization_name "Igloo Makers" -common
_name "ICL" -CA_cert_file_name "ca_igloo_cert.cer" -CA_pk_file_name "ca_igloo_pk
.cer" -nb_time 10/10/2010 -na_time 11/11/2011
Done.

To install the ca certificate use(this will move the old certificate):
  gen_ca_cert -install <File Name to install>
OR to add the certificate
  gen_ca_cert -install_add <File Name to install>
```

3 To install this CA, use the -install <name of certificate file> option. This option replaces any previous CA list and creates a new one that includes only this CA.

To add the new CA to the existing CA list, use the -install_add <name of certificate file>.

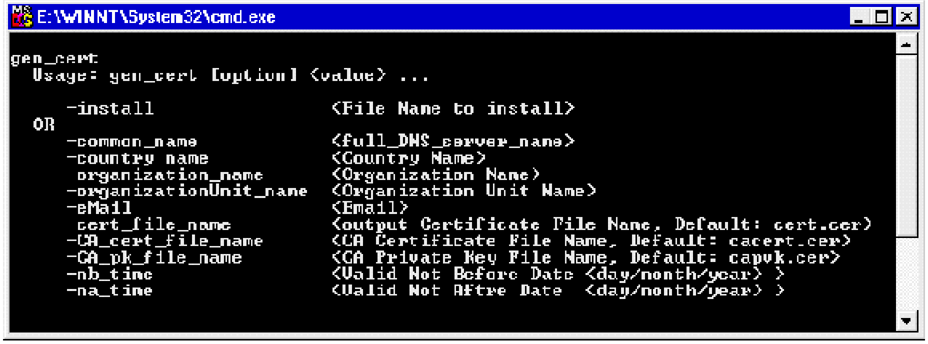


```
gen_ca_cert -install ca_igloo_cert.cer
ca_igloo_cert.cer is installed.
```

4 The -install and -install_add options install the certificate file only. Keep the private key file in a safe place and use it only for issuing certificates.

To create a Digital Certificate using gen_cert:

1 To view the format and usage, run the *gen_cert* utility from the <ProTune root folder>\launch_service\bin directory.



```
gen_cert
  Usage: gen_cert [option] <value> ...

     -install                   <File Name to install>
  OR
     -common_name               <full_DNS_server_name>
     -country name              <Country Name>
      organization_name         <Organization Name>
     -organizationUnit_name     <Organization Unit Name>
     -eMail                     <Email>
      cert_file_name            <output Certificate File Name, Default: cert.cer>
     -CA_cert_file_name         <CA Certificate File Name, Default: cacert.cer>
     -CA_pk_file_name           <CA Private Key File Name, Default: capvk.cer>
     -nb_time                   <Valid Not Before Date <day/month/year> >
     -na_time                   <Valid Not Aftre Date  <day/month/year> >
```

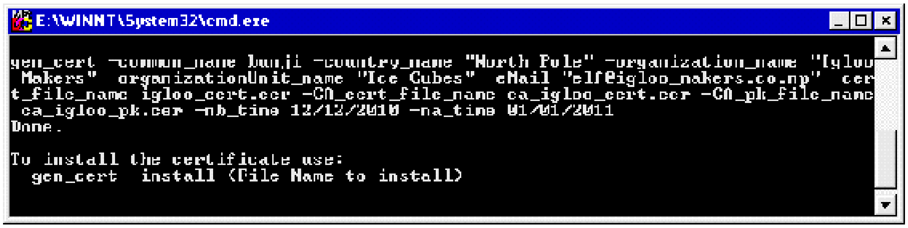
2 Create a new Digital Certificate by running the gen_cert command with at least one of the options: -country_name <country name>, -organization_name <organization name>, -organization_unit_name <organization unit name>, -eMail <email address> and -common_name <the name, full name or IP address of the machine>.

The CA Certificate and the CA Private Key files are necessary for the creation of the certificate. By default, it is assumed that they are in the current directory, and are named *cacert.cer* and *capvk.cer* respectively. In any other case, use the -CA_cert_file_name and -CA_pk_file_name options to give the correct files and locations.

In this process, the certificate file is created in the directory from which the utility was run. By default, the file name is *cert.cer*. To provide a different name, use the -cert_file_name option.

By default, the CA is valid for three years, from the time that the CA is generated. To change the validation dates, use the -nb_time <beginning of validity in dd/mm/yyyy format> and/or -na_time <ending of validity in dd/mm/yyyy format> options.

The following example creates the *igloo_cert.cer* file in the current directory:



```
gen_cert -common_name bunji -country_name "North Pole" -organization_name "Igloo
 Makers"  organizationUnit_name "Ice Cubes"  eMail "elf@igloo_makers.co.np"  cer
t_file_name igloo_cert.cer -CA_cert_file_name ca_igloo_cert.cer -CA_pk_file_name
 ca_igloo_pk.cer -nb_time 12/12/2010 -na_time 01/01/2011
Done.

To install the certificate use:
  gen_cert  install (File Name to install)
```

3 If you wish to install this certificate, use the -install <name of certificate file> option. This option replaces any previous certificate, as it is possible to own only one certificate per machine.

Index

Host Resolution Functions Copyright Agreement

Copyright (c) 1980, 1983, 1985, 1987, 1988, 1989, 1990, 1993

The Regents of the University of California. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. All advertising materials mentioning features or use of this software must display the following acknowledgement:

   This product includes software developed by the University of California, Berkeley and its contributors.
4. Neither the name of the University nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS ‘‘AS IS” AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Portions Copyright (c) 1993 by Digital Equipment Corporation.

Permission to use, copy, modify, and distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies, and that the name of Digital Equipment Corporation not be used in advertising or publicity pertaining to distribution of the document or software without specific, written prior permission.

THE SOFTWARE IS PROVIDED "AS IS" AND DIGITAL EQUIPMENT CORP. DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL DIGITAL EQUIPMENT CORPORATION BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Copyright (c) 1996 by Internet Software Consortium.

Permission to use, copy, modify, and distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND INTERNET SOFTWARE CONSORTIUM DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL INTERNET SOFTWARE CONSORTIUM BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

MERCURY INTERACTIVE

Mercury Interactive Corporation
1325 Borregas Avenue
Sunnyvale, CA 94089 USA

Main Telephone: (408) 822-5200
Sales & Information: (800) TEST-911, (866) TOPAZ-4U
Customer Support: (877) TEST-HLP
Fax: (408) 822-5300

Home Page: www.mercuryinteractive.com
Customer Support: support.mercuryinteractive.com