HP Performance Center

for the Windows® operating systems

Software Version: 9.50

Monitor Reference

Document Number: T7331-90014
Document Release Date: January 2009
Software Release Date: January 2009

Legal Notices

Warranty

The only warranties for HP products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. HP shall not be liable for technical or editorial errors or omissions contained herein.

The information contained herein is subject to change without notice.

Restricted Rights Legend

Confidential computer software. Valid license from HP required for possession, use or copying. Consistent with FAR 12.211 and 12.212, Commercial Computer Software, Computer Software Documentation, and Technical Data for Commercial Items are licensed to the U.S. Government under vendor's standard commercial license.

Third-Party Web Sites

HP provides links to external third-party Web sites to help you find supplemental information. Site content and availability may change without notice. HP makes no representations or warranties whatsoever as to site content or availability.

Copyright Notices

Trademark Notices

Intel® is a trademark of Intel Corporation in the U.S. and other countries.

Java™ is a US trademark of Sun Microsystems, Inc.

Microsoft® and Windows® are U.S. registered trademarks of Microsoft Corporation.

Oracle® is a registered US trademark of Oracle Corporation, Redwood City, California.

UNIX® is a registered trademark of The Open Group.

Documentation Updates

This guide's title page contains the following identifying information:

- Software Version number, which indicates the software version.
- Document Release Date, which changes each time the document is updated.
- Software Release Date, which indicates the release date of this version of the software.

To check for recent updates, or to verify that you are using the most recent edition of a document, go to:

http://h20230.www2.hp.com/selfsolve/manuals

This site requires that you register for an HP Passport and sign-in. To register for an HP Passport ID, go to:

http://h20229.www2.hp.com/passport-registration.html

Or click the **New users - please register** link on the HP Passport login page.

You will also receive updated or new editions if you subscribe to the appropriate product support service. Contact your HP sales representative for details.

Support

You can visit the HP Software Support web site at:

http://www.hp.com/go/hpsoftwaresupport

This web site provides contact information and details about the products, services, and support that HP Software offers.

HP Software Support Online provides customer self-solve capabilities. It provides a fast and efficient way to access interactive technical support tools needed to manage your business. As a valued support customer, you can benefit by using the HP Software Support web site to:

- Search for knowledge documents of interest
- Submit and track support cases and enhancement requests
- Download software patches
- Manage support contracts
- Look up HP support contacts
- Review information about available services
- Enter into discussions with other software customers
- Research and register for software training

Most of the support areas require that you register as an HP Passport user and sign in. Many also require a support contract.

To find more information about access levels, go to:

http://h20230.www2.hp.com/new_access_levels.jsp

To register for an HP Passport ID, go to:

http://h20229.www2.hp.com/passport-registration.html

Table of Contents

Welcome to This Guide 13
How This Guide Is Organized 13
Who Should Read This Guide 16
Performance Center Documentation Library 16
Additional Online Resources 18
Documentation Updates 19

PART I: INTRODUCTION

Chapter 1: Introducing Performance Center Monitoring 23
Setting Up the Monitoring Environment 24
Configuring Performance Center Monitors on the Controller 26
Configuring the Remote Machine for SiteScope Monitors 26

PART II: RUN-TIME AND TRANSACTION MONITORING

Chapter 2: Run-Time Monitoring 39
Run-Time Graphs 39

Chapter 3: Transaction Monitoring 43
Transaction Monitor Graphs 43
Enabling the Transaction Monitor 44
Adding Transactions to a Script 45

PART III: WEB RESOURCE MONITORING

Chapter 4: Web Resource Monitoring 49
About Web Resource Monitoring 49
Hits per Second Graph 50
Throughput Graph 50
HTTP Responses per Second Graph 51
Pages Downloaded per Second Graph 53
Retries per Second Graph 54
Connections Graph 55
Connections per Second Graph 55
SSL Connections per Second Graph 56

PART IV: SYSTEM RESOURCE MONITORING

Chapter 5: Introducing System Resource Monitoring 59

Chapter 6: Antara FlameThrower Monitoring 61
Configuring the Antara FlameThrower Monitor 61
Antara FlameThrower Performance Counters 63

Chapter 7: Windows Resource Monitoring 75
Configuring the Windows Resources Monitor 76
Windows Resources Performance Counters 79

Chapter 8: SNMP Resource Monitoring 83
Configuring the SNMP Resources Monitor 83
Configuring the SiteScope SNMP by MIB Monitor 87

Chapter 9: SiteScope Resource Monitoring 91
Before Setting Up the SiteScope Monitor 91
Configuring the SiteScope Monitor 92

Chapter 10: Server Resource Monitoring 97
Setting Up the Server Resources Monitor 98
Configuring the Server Resources Monitor 98
Server Resources Performance Counters 100

Chapter 11: UNIX Resource Monitoring 103
Setting Up the Monitoring Environment 104
Configuring the UNIX Resources Monitor 106
UNIX Resources Performance Counters 108

PART V: NETWORK MONITORING

Chapter 12: Introducing Network Monitoring 111

Chapter 13: Network Delay Monitoring 113
Network Monitoring from a UNIX Source Machine 113
Configuring the Network Delay Time Monitor 117
Network Monitoring over a Firewall 121

PART VI: FIREWALL SERVER MONITORING

Chapter 14: Firewall Server Performance Monitoring 125
About the Firewall Server Monitor 125
Configuring the Check Point FireWall-1 Server Monitor 125
Check Point FireWall-1 Performance Counters 127

PART VII: WEB SERVER MONITORING

Chapter 15: Introducing Web Server Monitoring 131
About Web Server Resource Monitors 131
Monitoring Using a Proxy Server 132

Chapter 16: Apache Resource Monitoring 135
Configuring the Apache Monitor 135
Apache Performance Counters 137

Chapter 17: Microsoft IIS Monitoring 139
Configuring the Microsoft IIS Monitor 139
Microsoft IIS Performance Counters 142

Chapter 18: iPlanet/Netscape Monitoring 143
Configuring the iPlanet/Netscape Monitor 144
iPlanet/Netscape Performance Counters 147

Chapter 19: iPlanet (SNMP) Monitoring 149
Configuring the iPlanet (SNMP) Monitor 149
iPlanet (SNMP) Performance Counters 151

PART VIII: WEB APPLICATION SERVER RESOURCE MONITORING

Chapter 20: Introducing Web Application Server Resource Monitoring 159

Chapter 21: Ariba Monitoring 161
Configuring the Ariba Monitor 161
Ariba Performance Counters 164

Chapter 22: ATG Dynamo Monitoring 167
Configuring the ATG Dynamo Monitor 167
ATG Dynamo Performance Counters 170

Chapter 23: BroadVision Monitoring 173
Configuring the BroadVision Monitor 174
BroadVision Servers and Services 177
BroadVision Performance Counters 179

Chapter 24: ColdFusion Monitoring 187
Configuring the ColdFusion Monitor 188
ColdFusion Performance Counters 190

Chapter 25: iPlanet (NAS) Monitoring 191
Setting Up the Monitoring Environment 192
Configuring the iPlanet (NAS) Monitor 196
iPlanet (NAS) Performance Counters 198

Chapter 26: Microsoft Active Server Pages Monitoring 209
Configuring the Microsoft Active Server Pages Monitor 209
MS Active Server Pages Performance Counters 212

Chapter 27: Oracle9iAS HTTP Server Monitoring 213
Configuring the Oracle9iAS HTTP Server Monitor 213
Oracle9iAS HTTP Server Performance Counters 217

Chapter 28: SilverStream Monitoring 223
Configuring the SilverStream Monitor 223
SilverStream Performance Counters 226

Chapter 29: WebLogic (JMX) Monitoring 227
Setting Up the Monitoring Environment 227
Configuring the WebLogic (JMX) Monitor 231
BEA WebLogic (JMX) Performance Counters 235

Chapter 30: WebSphere Monitoring 237
Setting Up the Monitoring Environment 237
Configuring the WebSphere Monitor 238
WebSphere Performance Counters 243

Chapter 31: WebSphere Application Server Monitoring 251
Setting up the Monitoring Environment 252
Configuring the WebSphere Application Server Monitor 254

Chapter 32: WebSphere (EPM) Monitoring 259
Setting Up the Monitoring Environment 259
Configuring the WebSphere (EPM) Monitor 267

PART IX: DATABASE RESOURCE MONITORING

Chapter 33: Introducing Database Resource Monitoring 271

Chapter 34: SQL Server Monitoring 273
Configuring the SQL Server Monitor 273
SQL Server Performance Counters 276

Chapter 35: DB2 Database Server Monitoring 279
Setting Up the Monitoring Environment 279
Configuring the DB2 Monitor 281
DB2 Performance Counters 282

Chapter 36: Oracle Database Server Monitoring 297
Setting Up the Oracle Monitor Environment 298
Configuring Oracle Database Resource Monitoring 302
Oracle Performance Counters 306
Custom Queries 307

Chapter 37: Sybase Database Server Monitoring 309
Setting Up the Monitoring Environment 310
Configuring the Sybase Monitor 311
Sybase Performance Counters 314

PART X: J2EE PERFORMANCE MONITORING

Chapter 38: J2EE Performance Monitoring 323
About J2EE Performance Monitoring 323
Setting Up the Monitoring Enviroment 324
Configuring the J2EE Monitor 329
J2EE Performance Counters 331
Examples of Modifying Application Server Configuration 332

PART XI: APPLICATION DEPLOYMENT MONITORING

Chapter 39: Application Deployment Solutions 345
Setting Up the Monitoring Environment 346
Configuring the Citrix MetaFrame Server Monitor 347
Citrix MetaFrame Performance Counters 350

PART XII: ERP/CRM SERVER RESOURCE MONITORING

Chapter 40: Introducing ERP/CRM Server Resource Monitoring 359

Chapter 41: SAPGUI Monitoring....361
Setting Up the Monitoring Environment 362
Configuring the SAPGUI Monitor....362
SAPGUI Performance Counters....365

Chapter 42: SAP Portal Monitoring....367
Configuring the SAP Portal Monitor....367
SAP Portal Performance Counters....371

Chapter 43: Siebel Web Server Monitoring....373
Setting Up the Monitoring Environment....373
Configuring the Siebel Web Server Monitor....374
Siebel Web Server Performance Counters....378

Chapter 44: Siebel Server Manager Monitoring....379
Setting up the Monitoring Environment....379
Configuring the Siebel Server Manager Monitor....381
Siebel Server Manager Performance Counters....382

Chapter 45: PeopleSoft (Tuxedo) Monitoring....385
Setting Up the PeopleSoft (Tuxedo) Monitor....385
Configuring the PeopleSoft (Tuxedo) Monitor....386
PeopleSoft (Tuxedo) Performance Counters....389

PART XIII: APPLICATION COMPONENT MONITORING

Chapter 46: Introducing Application Component Monitoring....393

Chapter 47: Microsoft COM+ Monitoring....395
Setting Up the Monitoring Environment....395
Configuring the Microsoft COM+ Server Monitor....396
Microsoft COM+ Performance Counters....398

PART XIV: MIDDLEWARE PERFORMANCE MONITORING

Chapter 48: Middleware Performance Monitoring....405
Setting Up the Monitoring Environment....406
Configuring the Tuxedo Monitor....407
Tuxedo Monitor Measurements....410

PART XV: TROUBLESHOOTING MONITORS

Chapter 49: Troubleshooting Monitors 415
Cannot Create a Monitor 415
Cannot Save Measurements 417
Port Required for J2EE Monitor Is Already Taken 418
J2EE Monitor Initialization Errors 419
Cannot Save SiteScope Monitors with Many Counters 419

Index 421

Welcome to This Guide

Welcome to the *HP Performance Center Monitor Reference*.

Performance Center is HP's Web-enabled global load testing tool, which is specially designed to streamline the testing process and increase the test efficiency for multiple concurrent load tests across multiple geographic locations.

This guide describes how to configure the HP Performance Center monitors and SiteScope monitors, and lists each monitor's performance counters.

This chapter includes:

- How This Guide Is Organized on page 13
- Who Should Read This Guide on page 16
- Performance Center Documentation Library on page 16
- Additional Online Resources on page 18
- Documentation Updates on page 19

How This Guide Is Organized

The *HP Performance Center Monitor Reference* contains the following sections:

Part I Introduction

Introduces Performance Center monitoring and setting up monitoring environments.

Part II Run-Time and Transaction Monitoring

The **Run-Time** and **Transaction** monitors display information related to the Vusers and transactions during the load test run.

Part III Web Resource Monitoring

The **Web Resource** monitor measures statistics at the Web servers during load test runs. It provides information about the number of Web connections, throughput volume, HTTP responses, server retries, and downloaded pages during the load test.

Part IV System Resource Monitoring

The **System Resource** monitors monitor a machine's system resource usage during a load test run. They guage the Antara FlameThrower, Windows, Unix, SiteScope server resources used during a load test.

Part V Network Monitoring

The **Network** monitor displays information about the network delays on your system.

Part VI Firewall Server Monitoring

The **Firewall** monitor measures statistics at the firewall servers during the load test.

Part VII Web Server Monitoring

The **Web Server** resource monitors measure statistics on the Apache, Microsoft IIS, iPlanet (SNMP) and iPlanet/Netscape Web servers during the load test.

Part VIII Web Application Server Resource Monitoring

The **Web Application Server Resource** monitors measure statistics on the Ariba, ATG Dynamo, BroadVision, ColdFusion, iPlanet (NAS), MS Active Server Pages, Oracle9iAS, SilverStream, WebLogic (JMX), and WebSphere application servers during the load test.

Part IX Database Resource Monitoring

The **Database Server Resource** monitors measure statistics related to the SQL, DB2, Sybase, and Oracle servers.

Part X J2EE Performance Monitoring

The **Java Performance** monitor measure statistics of Java 2 Platform, Enterprise Edition (J2EE) objects using the J2EE server machine.

The **J2EE/.NET Diagnostics** monitors provide information to trace, time, and troubleshoot individual transactions through J2EE/.NET Web, application, and database servers. For information about the **J2EE/.NET Diagnostics** monitors, see the HP Diagnostics documentation.

Part XI Application Deployment Monitoring

The **Application Deployment Solutions** monitor measures statistics of the Citrix MetaFrame XP and 1.8 servers during a load test run.

Part XII ERP/CRM Server Resource Monitoring

The **ERP/CRM Server Resource** monitors measure statistics of the SAP, SAPGUI, SAP Portal, Siebel Server Manager, Siebel Web Server, and PeopleSoft (Tuxedo) server during a load test run.

Part XIII Application Component Monitoring

The **Application Component** monitors measure statistics of the Microsoft COM+ server during a load test run.

Part XIV Middleware Performance Monitoring

The **Middleware Performance** monitor measures statistics of the Tuxedo server during a load test run.

Part XV Troubleshooting Monitors

This section provides troubleshooting for problems related to working with the Performance Center monitors.

Who Should Read This Guide

This guide is for the following users of Performance Center:

- Performance Engineers
- Project Managers

This document assumes that you are moderately knowledgeable about enterprise application development and highly skilled in enterprise system and database administration.

Performance Center Documentation Library

Performance Center includes a complete set of documentation describing how to use the product. A comprehensive Documentation Library is available from the **Help** menu in the User and Administration Sites. PDFs can be read and printed using Adobe Reader, which can be downloaded from the Adobe Web site *(*http://www.adobe.com*)*. Printed documentation is also available on demand.

Accessing the Documentation

You can access Performance Center documentation as follows:

- In the User and Administration Sites, click the **Help** link in the top right corner of the window. In the bookmarks on the left, click **Home** to open the Performance Center Documentation Library front page which provides quick links to the online Performance Center Documentation Library.
- Printable guides can be accessed from **Start** > **Program Files** > **Performance Center** > **Documentation**.

Getting Started Documentation

- **Readme.** Provides last-minute news and information about Performance Center.

- **HP Performance Center System Configuration and Installation Guide.** Explains how to install and configure the Performance Center components. Available in PDF format only. Accessible from the HP Installation DVD.

- **HP Performance Center Quick Start**. A self-paced guide showing you how to use Performance Center to create, run, and monitor load tests. Available in PDF format only.

  Accessible from the General tab that is displayed when you open the User Site.

Performance Center User Guides

- **HP Performance Center User Guide.** Describes how to use Performance Center to create, schedule, run, and monitor load tests.
- **HP Performance Center Monitor Reference.** Describes how to set up the server monitor environment and configure Performance Center monitors for monitoring data generated during a load test run.

Performance Center Administration

- **HP Performance Center Administrator Guide.** Describes how to use Performance Center Administration Site for overall resource management, site management, system configuration, technical supervision, and user privileges in Performance Center projects.

Troubleshooting

- **HP Performance Center Troubleshooting Guide.** Provides information for troubleshooting problems while working with Performance Center Available in PDF fomat only.

Standalone Applications

The following documentation is available only on host machines, and if standalone components are installed:

- **HP Analysis User Guide.** Describes how to generate graphs and reports to analyze your load tests after they have run.

➤ **HP Virtual User Generator User Guide.** Describes how to create Vuser scripts through recording and programming.

Additional Online Resources

Troubleshooting and Knowledge Base accesses the Troubleshooting page on the HP Software Support Web site where you can search the Self-solve knowledge base. The URL for this Web site is http://h20230.www2.hp.com/troubleshooting.jsp.

HP Software Support accesses the HP Software Support Web site. This site enables you to browse the Self-solve knowledge base. You can also post to and search user discussion forums, submit support requests, download patches and updated documentation, and more. The URL for this Web site is www.hp.com/go/hpsoftwaresupport.

Most of the support areas require that you register as an HP Passport user and sign in. Many also require a support contract.

To find more information about access levels, go to:
http://h20230.www2.hp.com/new_access_levels.jsp

To register for an HP Passport user ID, go to:
http://h20229.www2.hp.com/passport-registration.html

HP Software Web site accesses the HP Software Web site. This site provides you with the most up-to-date information on HP Software products. This includes new software releases, seminars and trade shows, customer support, and more. The URL for this Web site is www.hp.com/go/software.

Documentation Updates

HP Software is continually updating its product documentation with new information.

To check for recent updates, or to verify that you are using the most recent edition of a document, go to the HP Software Product Manuals Web site (http://h20230.www2.hp.com/selfsolve/manuals).

Part I

Introduction

1

Introducing Performance Center Monitoring

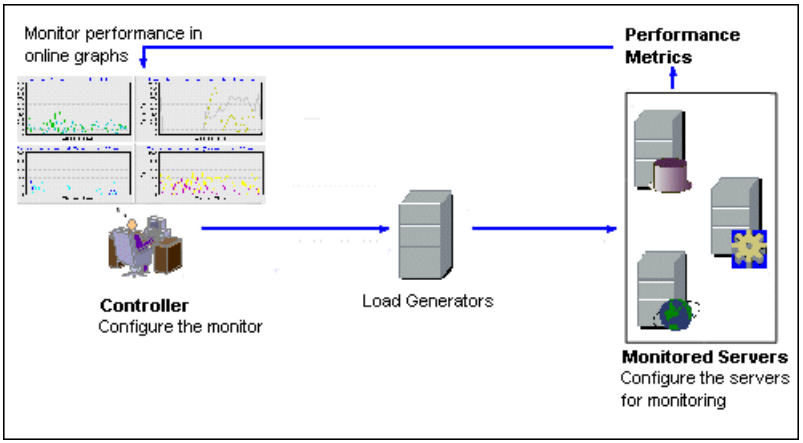
You can monitor a load test run using the Performance Center online monitors.

This chapter includes:

- Setting Up the Monitoring Environment on page 24
- Configuring Performance Center Monitors on the Controller on page 26
- Configuring the Remote Machine for SiteScope Monitors on page 26

Setting Up the Monitoring Environment

Before monitoring a load test run, you need to set up and configure the Performance Center monitoring components. Each monitor has different configuration requirements that are explained in the specific monitoring chapters. The diagram below illustrates the Performance Center monitoring process.



Before monitoring a server, perform the following steps:

- configure the monitoring environment on the server machine (if necessary)
- configure the monitor on the Controller machine

Configuring the Monitoring Environment on the Server Machine

To use the following monitors, you must first install or configure monitoring components on the server machine:

- Citrix
- DB2
- iPlanet (NAS)
- J2EE
- Microsoft .NET CLR
- Microsoft COM+
- Network Delay
- Oracle
- PeopleSoft (Tuxedo)
- SAPGUI
- SAP Portal
- Siebel Server Manager
- Siebel Web Server
- SiteScope
- Sybase
- Tuxedo
- UNIX
- WebLogic (JMX)
- WebSphere
- WebSphere Application Server
- WebSphere (EPM)

Configuring Performance Center Monitors on the Controller

To obtain performance data for a monitor, you need to configure the monitor (from the Controller), and indicate which statistics and measurements you want to monitor. You select these counters using the monitor's Add Measurements page.

To set up a monitor, you need to perform the following:

1 Select monitor.

Select the monitor that you want to measure from the Add Monitor page.

2 Select server.

Add a monitored machine to the Controller by selecting the server whose monitors you want to configure.

For SiteScope monitors, configure the remote machine.

3 Select measurements.

Configure the monitor by selecting the measurements that you want to monitor.

For more information on setting up the monitoring environment and configuring a monitor, see the specific monitoring chapter.

Configuring the Remote Machine for SiteScope Monitors

You configure the remote machine according to the platform your machine is running on. This section describes configuring remote machines running on NT and UNIX.

This section includes:

➤ "Configuring the NT Remote Machine" on page 27

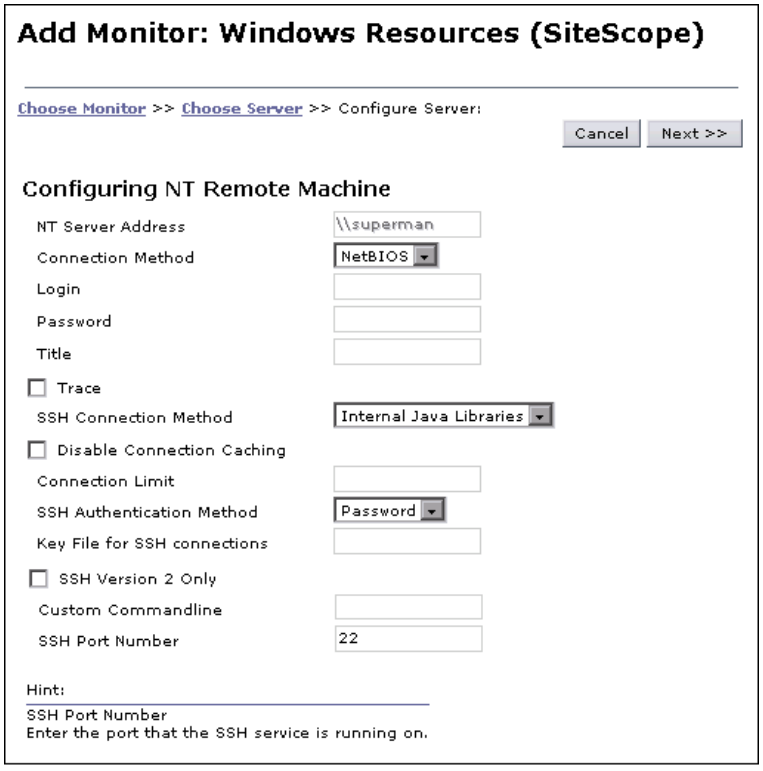
➤ "Configuring the UNIX Remote Machine" on page 31

Configuring the NT Remote Machine

For machines on a NT platform, when you add measurements to monitor for the first time, you need to configure the remote machine properties. You configure the remote machine properties in the Configuring NT Remote Machine page.

To configure the NT remote machine:

1 After you type monitor information for the monitor and click **Next**, the Configuring NT Remote Machine dialog box opens.



2 Enter the remote machine's configuration information.

- **NT Server Address**. The IP address or UNC style name of the NT server you wish to monitor. An IP host name will also work provided that the SiteScope server has a way to resolve this common name into an IP address (for example, by the use of a hosts file, DNS, or WINS/DNS integration).
- **Connection Method**. SiteScope can use one of two connection types for monitoring remote NT server resources. These are:
  - **NetBIOS**. The default server-to-server communication protocol for Windows NT and 2000 networks.
  - **SSH**. Secure Shell, a more secure communication protocol that can be installed on Windows NT/2000 based networks. This connection method normally requires installing SSH libraries on each server to which you want to connect.
- **Login**. The login for the remote server. If the server is within the same domain as the SiteScope machine, include the domain name in front of the user login name. For example, domainname\user. If you are using a local machine login account for machines within or outside the domain, include the machine name in front of the user login name. For example, machinename\user.
- **Password**. The password for the remote server or the passphrase for the SSH key file.

  When using SSH authentication with public/private key based authentication type the passphrase for the identity file here.
- **Title**. (Optional) A name by which the remote machine should be known. This name will appear in the drop-down list.
- **Trace**. Check this box to have trace messages to and from the subject server recorded to the SiteScope RunMonitor.log file.
- **SSH Connection Method**. The method to use for this connection. The currently supported methods are:
  - **Internal Java Libraries**. Connect using the Java SSH client integrated with SiteScope.

- **Plink**. Connect using an external SSH client. On NT, SiteScope ships with Plink. On UNIX or Linux SiteScope will use an installed client such as OpenSSH.

- **Disable Connection Caching**. Check this option to turn off connection caching for this remote. By default SiteScope caches open connections.
- **Connection Limit**. Controls the number of open connections that SiteScope will allow for this remote. If you have a large number of monitors configured to use this connection then set this number high enough to relieve the potential bottleneck.

Note: This setting does not effect the running of tests for a remote, tests will always create a new connection.

- **SSH Authentication Method**. The authentication method to use for SSH connections. The currently supported methods are:
  - **Password**. Authenticate using a password.
  - **Key File**. Authenticate using public/private key authentication. When this option is selected SiteScope uses the private key in the file **SiteScope/groups/identity** to authenticate. The corresponding public key must be listed in the **authorized_keys** file on the remote host.
- **Key File for SSH connections**. Select the file that contains the private key for this connection. The default key file is **SiteScope\groups\identity**. This setting only applies when the authentication method is **Key File**.
- **SSH Version 2 Only**. Check this option to force SiteScope to use SSH protocol version 2 only. This option only applies when using the integrated Java Client in SiteScope.
- **Custom Commandline**. Type a custom commandline for a remote using the External Client. This option can be used when needing to pass specific options to the external client being executed. Valid substitution variable are:
  - **$root$.** This will be translated to the SiteScope directory.
  - **$user$.** This will be translated to the user name entered into the remote.

- **$password$.** This will be translated to the password entered into the remote.
- **$host$.** This will be translated to the host name entered into the remote.

- **SSH Port Number**. Type the port that the remote SSH server is listening on. By default, the port number is 22.

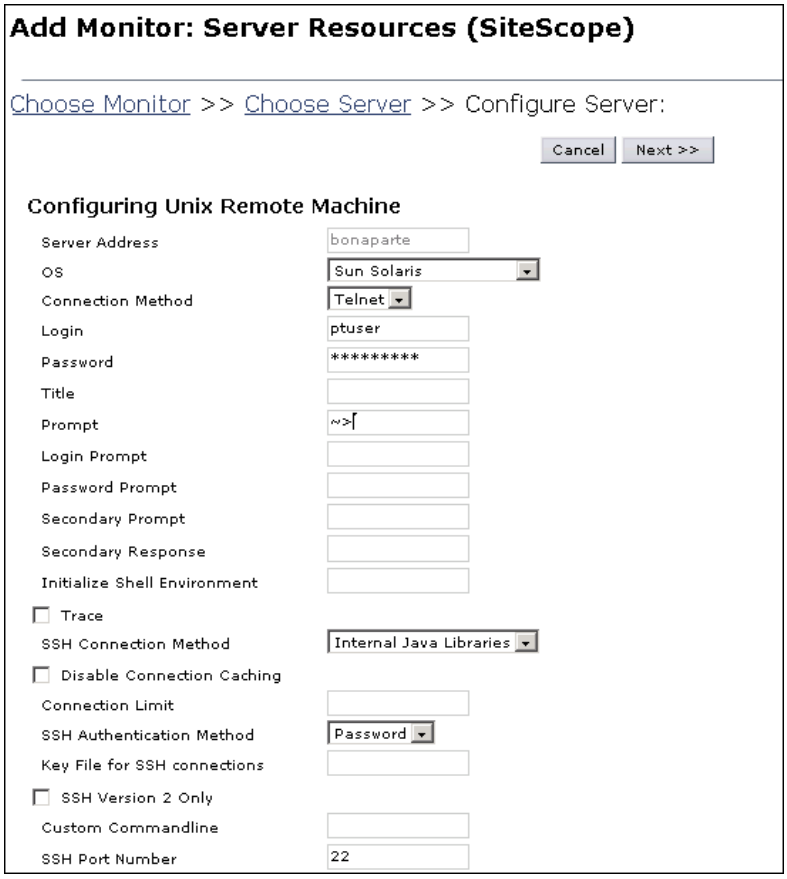
3 Click **Next**, and continue to add measurements in the Choose Measurements page.

Configuring the UNIX Remote Machine

For machines on a UNIX platform, when you add measurements to monitor for the first time, the Configuring Unix Remote Machine page opens.

To configure the UNIX remote machine:

1 After you enter monitor information for the monitor and click **Next**, the Configuring UNIX Remote Machine dialog box opens.



2 Enter the Unix Remote Machine information:

➤ **Server Address**. Displays the IP address or host name of the server that you entered in the Add Machine dialog box.

- **OS**. Select the operating system running on the remote server. The following versions of UNIX are supported:

AIX	OPENSERVER
FreeBSD	SCO
HP/UX	SGI Irix
HP/UX 64-bit	Sun Solaris
Linux	Tru64 5.x
MacOSX	Tru64 Pre 4.x (Digital)

- **Connection Method**. Select the method for connecting to the server. The supported methods are:
  - **Telnet**. Log in to the remote server using Telnet.
  - **SSH**. Log in to the remote server using the SSH protocol. This may require additional software and setup depending on the version of UNIX you are working with.
  - **Rlogin**. Log in to the remote server using the Rlogin protocol.
  - **HTTP**. Connect to an HTTP server on the remote server and run the command via a CGI. For this method, the Login and Password are optional and are used for authorizing the log on to the remote machine if required.
- **Login**. The login for the remote server.
- **Password**. Type the password for the remote server.
- **Title**. Type a name by which the remote machine should be known. This name will appear in the list in monitors that can connect to this server.
- **Prompt**. Type the prompt to be displayed when the system is ready to handle a command. The default is #.
- **Login Prompt**. Type the prompt to be displayed when the system is waiting for the login to be entered. The default is "ogin:".
- **Password Prompt**. Type the prompt to be displayed when the system is waiting for the password to be entered. The default is "assword:".

- **Secondary Prompt**. Type the prompt to be displayed if the telnet connection to the remote server causes the remote server to prompt for more information about the connection. Separate multiple prompt strings by commas (,). For example, for Telnet connections to some remote servers, the remote server may ask what terminal type should be emulated for the connection. In this case you might need to type Terminal type? as the secondary prompt. The response to the secondary prompt is entered in the Secondary Response field below.
- **Secondary Response**. Type the responses to secondary prompts required to establish connections with this remote server. Separate multiple responses with commas (,).
- **Initialize Shell Environment**. Type any shell commands to be executed at the beginning of the session. Separate multiple commands with a semicolon (;). This option allows you to specify shell commands to be executed on the remote machine directly after a Telnet or SSH session has been initiated. These commands can be used to customize the shell for each SiteScope remote.

  Examples:

  - The remote shell may not have the correct path set for SiteScope scripts to run. The following command will add the directory **/usr/local/bin** into the PATH of the current shell on the remote machine:

    ```
    export PATH=$PATH:/usr/local/sbin
    ```

  - The remote shell may not be initializing the pseudo terminal correctly. Type the following command to increase the terminal width to 1024 characters:

    ```
    stty cols 1024;${SHELL}
    ```

Note: Commands after a shell invocation will not be executed.

- There have been cases where the remote Telnet Server does not echo back the command line properly. This may cause strange behavior for monitors that rely on this behavior.

  Type the following command to force the remote terminal to echo:

  ```
  stty echo
  ```

- Certain UNIX shells have been known to behave erratically with SiteScope. This includes bash, ksh, and csh. Type the following command to change the shell to sh for the SiteScope connection:

  ```
  /bin/sh
  ```

- **Trace**. Select this option to trace messages to and from the remote server in the **RunMonitor.log** file.
- **SSH Connection Method**. Select the method to use to connect to the remote server.
  - **Internal Java Libraries**. Connect using the Java SSH client integrated with SiteScope.
  - **Plink**. Connect using an external SSH client. On Windows NT, SiteScope ships with Plink.
- **Disable Connection Caching**. Select this to disable SSH connection caching.
- **Connection Limit**. Type the maximum number of connections for this remote machine.
- **SSH Authentication Method**. Select the method to use to authenticate to the remote server (for SSH connections only).
  - **Password**. Authenticate using a password.
  - **Keyfile**. Authenticate using public/private key authentication. When this option is selected, SiteScope uses the private key in the file **SiteScope/groups/identity** to authenticate. The corresponding public key must be listed in the **authorized_keys** file on the remote host.
- **Key File for SSH connections**. Type the Key File for SSH connections.

- **SSH Version 2 Only**. Select this option to force SSH to only use SSH protocol version 2. This option is only supported when using the internal Java libraries connection method.
- **Custom Commandline**. Type the command for execution of the external SSH client. For substitutions with options listed above, use $host$, $user$, and $password$ respectively. This setting is supported only for connections using an external process.
- **SSH Port Number**. Type the port on which the SSH service is running.

After defining the server for SiteScope, you can test the settings by clicking on the test link. SiteScope attempts to display the working directory of the remote machine (the "pwd" command on UNIX, or "cd" on Windows NT), as a test to check that the remote machine can be accessed and can run commands properly.

3 Click **Next**, and continue to add measurements in the Choose Measurements page.

Part II

Run-Time and Transaction Monitoring

2

Run-Time Monitoring

While running a load test, you can use Performance Center's Run-Time monitor to view graphs of the run-time status.

This chapter includes:

➤ Run-Time Graphs on page 39

Run-Time Graphs

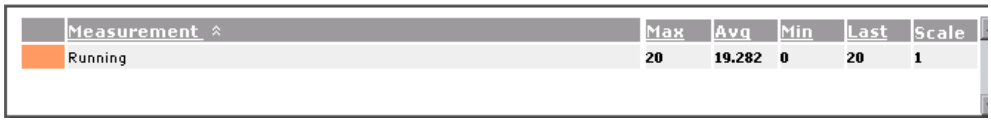
The **Run-Time** monitor provides information about the status of the Vusers participating in the load test, and the number and types of errors that the Vusers generate. In addition, the Run-Time monitor provides the User-Defined Data Points graph, which displays the real time values for user-defined points in a Vuser script.

The **Run-Time** monitor provides information for the following graphs:

- Running Vusers Graph
- Error Statistics Graph
- Vusers with Error Statistics Graph
- User-Defined Data Points Graph

Running Vusers Graph

The monitor's **Running Vusers** graph provides information about the status of the Vusers running in the current load test 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.



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 dialog box for a complete explanation of the error.

Error Statistics Graph

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

Vusers with Error Statistics Graph

The **Vusers with Error Statistics** graph provides details about the number of Vusers that generate errors during load test 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 *HP LoadRunner Online Function Reference*.

By default, Performance Center 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 load test. For more information, see the *HP LoadRunner Analysis User Guide*.

3

Transaction Monitoring

While running a load test, you can use Performance Center's Transaction monitors to view graphs of transaction performance.

This chapter includes:

- Transaction Monitor Graphs on page 43
- Enabling the Transaction Monitor on page 44
- Adding Transactions to a Script on page 45

Transaction Monitor Graphs

The **Transaction** monitor displays the transaction rate and response time during load test execution. For more information about transactions, see "Adding Transactions to a Script" on page 45.

The **Transaction** monitor provides the following graphs:

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

Transaction Response Time Graph

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 load test (x-axis).

Transactions per Second (Passed) Graph

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 load test (x-axis).

Transactions per Second (Failed, Stopped) Graph

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 load test (x-axis).

Total Transactions per Second (Passed) Graph

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 load test (x-axis).

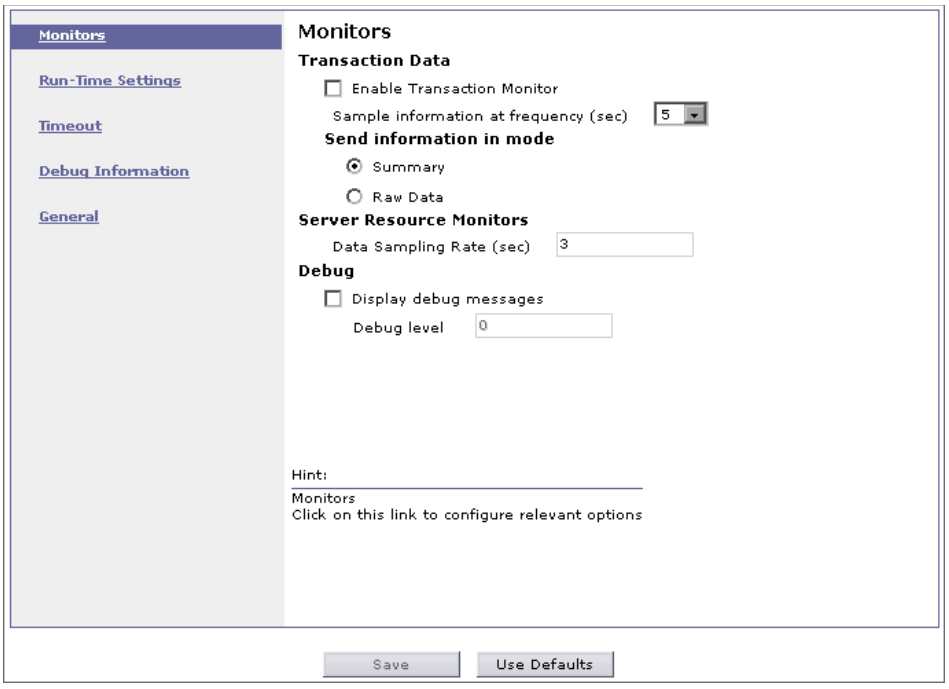
Enabling the Transaction Monitor

The Transaction monitor is enabled by default—it automatically begins monitoring Vuser transactions at the start of a load test. To conserve resources, you can disable the Transaction monitor.

To enable the transaction monitoring:

1 On the **Projects** menu, click **Options.**

2 In the left pane of the Options page, click **Monitors**.



3 Under Transaction Data, select **Enable Transaction Monitor**.

4 Click **Save**.

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	HP WinRunner
non-GUI Windows	HP Virtual User Generator (VuGen)

To add a transaction to a script:

1. On the **Projects** menu, click **Vuser Scripts** to open the Vuser Scripts page.
2.  Click the Download Script arrow. Performance Center prepares the script for download and opens the Ready for Download dialog box.
3. Click the **Download** button. The File Download dialog box opens.
4. Click **OK**. The Save As dialog opens.
5. Choose a file name and the location into which you want the file downloaded.
6. Click **Save** to download the file.
7. Using Vugen open and insert **Start** and **End Transaction** functions or markers throughout your script.

Note: Your version of VuGen may be enabled to upload scripts but not be enabled to download scripts. If you have trouble downloading scripts you may have to run the update for your version of VuGen. For more information, see the section about preparing VuGen to work with Performance Center in the *HP Performance Center User Guide*.

Part III

Web Resource Monitoring

4

Web Resource Monitoring

You can obtain information about the performance of your Web server using Performance Center's Web Resource monitor.

This chapter includes:

- About Web Resource Monitoring on page 49
- Hits per Second Graph on page 50
- Throughput Graph on page 50
- HTTP Responses per Second Graph on page 51
- Pages Downloaded per Second Graph on page 53
- Retries per Second Graph on page 54
- Connections Graph on page 55
- Connections per Second Graph on page 55
- SSL Connections per Second Graph on page 56

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 load test, 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, the number of server retries per second, the number of open TCP/IP connections, the number of new TCP/IP connections per second, and the number of SSL Connections per second.

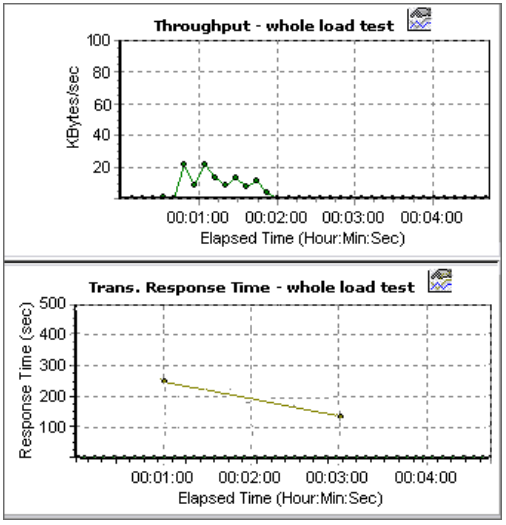
Hits per Second Graph

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 load test (x-axis). This graph can display the whole step, 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 load test run (x-axis). Throughput is measured in kilobytes and represents the amount of data that the Vusers received from the server at any given second. You can compare this graph to the Transaction Response Time graph to see how the throughput affects transaction performance.

In the following example, the Transaction Response time graph is compared with the Throughput graph. It is apparent from the graph that as the throughput decreases, the transaction response time also decreases. The peak throughput occurred at approximately 1 minute into the step. 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 load test run (x-axis), grouped by status code. You can group the results shown in this graph by script (using the "Group By" function) to locate scripts which generated error codes.

The following table displays a list of HTTP status codes:

Code	Description
200	OK
201	Created
202	Accepted
203	Non-Authoritative Information
204	No Content
205	Reset Content
206	Partial Content
300	Multiple Choices
301	Moved Permanently
302	Found
303	See Other
304	Not Modified
305	Use Proxy
307	Temporary Redirect
400	Bad Request
401	Unauthorized
402	Payment Required
403	Forbidden

Code	Description
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
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

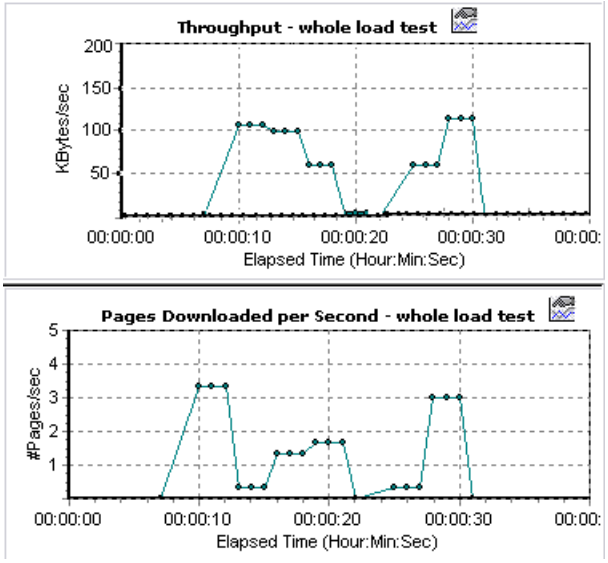
The **Pages Downloaded per Second** graph shows the number of Web pages (y-axis) downloaded from the server during each second of the load test run (x-axis). This graph helps you evaluate the amount of load Vusers generate, in terms of the number of pages downloaded.

Note: To view the Pages Downloaded per Second graph, you must select **Pages per second (HTML Mode only)** from the script's run-time settings Preferences tab before running your load test.

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 load test 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 load test (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.

Connections Graph

The **Connections** graph shows the number of open TCP/IP connections (y-axis) at each point in time of the load test (x-axis). Note that one HTML page may cause the browser to open several connections, when links on the page go to different Web addresses. Two connections are opened for each Web server.

This graph is useful in indicating when additional connections are needed. For example, if the number of connections reaches a plateau, and the transaction response time increases sharply, adding connections would probably cause a dramatic improvement in performance (reduction in the transaction response time).

Connections per Second Graph

The **Connections Per Second** graph shows the number of new TCP/IP connections (y-axis) opened each second of the load test (x-axis). This number should be a small fraction of the number of hits per second, because new TCP/IP connections are very expensive in terms of server, router and network resource consumption. Ideally, many HTTP requests should use the same connection, instead of opening a new connection for each request.

SSL Connections per Second Graph

The **SSL Connections per Second** graph shows the number of new and reused SSL Connections (y-axis) opened in each second of the load test (x-axis). An SSL connection is opened by the browser after a TCP/IP connection has been opened to a secure server.

Because creating a new SSL connection entails heavy resource consumption, you should try to open as few new SSL connections as possible; once you've established an SSL connection, you should reuse it. There should be no more than one new SSL connection per Vuser. If you set your run-time settings to simulate a new Vuser at each iteration (through the Browser Emulation tab in the Run-Time Settings menu), you should have no more than one new SSL connection per Vuser per iteration. Ideally, you should have very few new TCP/IP and SSL connections each second.

Part IV

System Resource Monitoring

5

Introducing System Resource Monitoring

You can monitor a machine's system resource usage during a load test run using Performance Center's System Resource monitors.

A primary factor in a transaction's response time is its system resource usage. Using the Performance Center resource monitors, you can monitor Antara FlameThrower, Windows, SNMP, SiteScope server resources, and UNIX resources on a machine during a load test run, and determine why a bottleneck occurred on a particular machine.

- The Antara FlameThrower monitor can measure the following performance counters: Layer, TCP, HTTP, SSL/HTTPS, Sticky SLB, FTP, SMTP, POP3, DNS, and Attacks.
- The Windows measurements correspond to the built-in counters available from the Windows Performance monitor.
- The SNMP monitor is available for monitoring machines using the Simple Network Management Protocol (SNMP). SNMP monitoring is platform independent.
- The SiteScope monitor can measure server, network, and processor performance counters. SiteScope can also be used to monitor SAP CCMS (Computer Center Management System) and SNMP server resources. For detailed information on all the performance counters that SiteScope can monitor, see the relevant SiteScope documentation.
- The Server Resources monitor shows the CPU, disk space, memory, or application resources used on remote Windows and UNIX servers during a load test.

- The UNIX measurements include those available by the **rstatd** daemon: average load, collision rate, context switch rate, CPU utilization, incoming packets error rate, incoming packets rate, interrupt rate, outgoing packets error rate, outgoing packets rate, page-in rate, page-out rate, paging rate, swap-in rate, swap-out rate, system mode CPU utilization, and user mode CPU utilization.

The System Resource monitors are automatically enabled when you execute a load test. 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 load test run.

6

Antara FlameThrower Monitoring

The Antara FlameThrower monitor displays statistics about the resource usage on the Antara FlameThrower server during the load test run.

The Antara FlameThrower monitor can measure the following performance counters: Layer, TCP, HTTP, SSL/HTTPS, Sticky SLB, FTP, SMTP, POP3, DNS, and Attacks.

This chapter includes:

- Configuring the Antara FlameThrower Monitor on page 61
- Antara FlameThrower Performance Counters on page 63

Configuring the Antara FlameThrower Monitor

You select the Antara FlameThrower resources you want to monitor using the Antara FlameThrower Choose Measurements page.

To configure the Antara FlameThrower monitor:

1 Open the Add Monitor window.

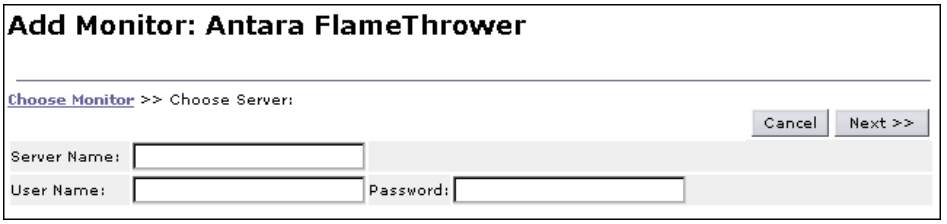
- **Before running a load test:** In the Load Test configuration page's **Monitors** tab, select the type of monitor profile to create:
  - To create a monitor profile for the current load test, click **Add Local Profile**.
  - To create a monitor profile that can be used for other load tests, click **Add Profile**, type a name and description of the profile and click **OK**.

  For details, see the section about configuring monitors in the *HP Performance Center User Guide*.

- **During a load test run:** On the Load Test run page:
  - Click the **Design** button. The Load Test Design dialog box opens.
  - Click **Monitors**.

  For details, see the section about creating monitor profiles while running a load test in the *HP Performance Center User Guide*.

2 Under **System Resource Graphs**, select **Antara FlameThrower**, then click **Next**. The Choose Server page opens.



3 Type the server name or IP address of the machine you want to monitor according to the following format:

```
<server name>:<port number>
```

For example, merc1:12135

4 Type the user login name and password.

5 Click **Next**. The Choose Measurements page opens, displaying the available measurements and server properties.

6 Select the measurements you want to monitor.

For a list of the available performance counters, see "Antara FlameThrower Performance Counters" on page 63.

7 Click **Save** to add the selected resource measurements to the monitor profile.

Antara FlameThrower Performance Counters

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.
SuccTCPConn Rate(/sec)	Number of SYN ACK packets received per second.
TCPConnLatency(millisec)	Interval between transmitting a SYN packet and receiving a SYN ACK reply packet in milliseconds.
MinTCPConn Latency(msec)	Minimum TCPConnectionLatency in milliseconds.
MaxTCPConn Latency(msec)	Maximum TCPConnectionLatency in milliseconds.
TCPSndConnClose	Total number of FIN or FIN ACK packets transmitted (Client).

Measurement	Description
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(msec)	The average HTTP Data Latency over the past second in milliseconds.
HTTPData Latency(msec)	Interval between transmitting a Request packet and receiving a response in milliseconds.
DataThroughput (bytes/sec)	The number of data bytes received from the HTTP server per second.
MinHTTPData Latency(msec)	Minimum HTTP Data Latency in milliseconds.
MaxHTTPData Latency(msec)	Maximum HTTP Data Latency in milliseconds.
MinData Throughput (bytes/sec)	Minimum HTTP Data Throughput in seconds.

Measurement	Description
MaxData Throughput (bytes/sec)	Maximum HTTP Data Throughput 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 Client Hello messages sent by the Client.
SSLConnection Rate(/sec)	Number of Client Hello 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.
SuccSSLConnectionRate(/sec)	Number of successful SSL connections established per second.
SSLAlertErrors	Number of SSL alert messages received by the client (e.g. bad_record_mac, decryption_failed, handshake_failure, and so on).
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 cookies that were not authenticated by the Server.
SuccCookie Authentication	The number of cookies 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 IDs that were not authenticated by the server.
SSLSessions Resumed	The number of Session IDs 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 Hellos 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 (msec)	Interval between transmitting an FTP User command packet and receiving a response in milliseconds.
MinFTPUserLatency(msec)	Minimum FTP Users Latency in milliseconds.
MaxFTPUserLatency(msec)	Maximum FTP Users Latency in milliseconds.
SuccFTPUsers	Total number of successful FTP User command replies received.

Measurement	Description
SuccFTPUserRate (/sec)	Number of successful 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 (msec)	Interval between transmitting a FTP PASS packet and receiving a response in milliseconds.
MinFTPPassLatency (msec)	Minimum FTP Pass Latency in milliseconds.
MaxFTPPassLatency(msec)	Maximum FTP Pass Latency in milliseconds.
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.
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.
FTPDataConnectionR ate(/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.

Measurement	Description
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 (msec)	Interval between transmitting a HELO packet and receiving a response in milliseconds.
MinSMTPHelo Latency(msec)	Minimum SMTP Helo Latency in milliseconds.
MaxSMTPHelo Latency(msec)	Maximum SMTP Helo Latency in milliseconds.
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(msec)	Interval between transmitting a Mail From packet and receiving a response in milliseconds.

Measurement	Description
MinSMTPMailFromLatency(msec)	Minimum SMTP Mail From Latency in milliseconds.
MaxSMTPMailFromLatency(msec)	Maximum SMTP Mail From Latency in milliseconds.
SuccSMTPMail Froms	Total number of successful Mail From replies received.
SuccSMTPMailFromRate(/sec)	Number of successful Mail From replies received per second.
SMTPRcptTos	Total number of Rcpt To packets transmitted.
SMTPRcptToRate (/sec)	Number of Rcpt To packets transmitted per second.
SMTPRcptTo Latency(msec)	Interval between transmitting a Rcpt To packet and receiving a response in milliseconds.
MinSMTPRcptTo Latency(msec)	Minimum SMTP Rcpt To Latency in milliseconds.
MaxSMTPRcptTo Latency(msec)	Maximum SMTP Rcpt To Latency in milliseconds.
SuccSMTPRcptTos	Total number of successful RcptTo replies received.
SuccSMTPRcptTo Rate(/sec)	Number of successful Rcpt To replies received per second.
SMTPDatas	Total number of Data packets transmitted.
SMTPDataRate(/sec)	Number of Data packets transmitted per second.
SMTPDataLatency (msec)	Interval between transmitting a Data packet and receiving a response in milliseconds.
MinSMTPData Latency(msec)	Minimum SMTP Data Latency in milliseconds.
MaxSMTPData Latency(msec)	Maximum SMTP Data Latency in milliseconds.

Measurement	Description
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 (msec)	Interval between transmitting a Pop3 User command packet and receiving a response in milliseconds.
MinPOP3User Latency(msec)	Minimum POP3UserLatency in milliseconds.
MaxPOP3User Latency(msec)	Maximum POP3UserLatency in milliseconds.
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.
POP3PassRate(/sec)	Number of Pop3 Pass command packets transmitted per second.
POP3PassLatency (msec)	Interval between transmitting a Pop3 Pass packet and receiving a response in milliseconds.
MinPOP3Pass Latency(msec)	Minimum POP3PassLatency in milliseconds.
MaxPOP3Pass Latency(msec)	Maximum POP3PassLatency in milliseconds.
SuccPOP3Passes	Total number of successful Pop3 Pass replies received.

Measurement	Description
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 (msec)	Interval between transmitting a Pop3 Stat packet and receiving a response in milliseconds.
MinPOP3Stat Latency(msec)	Minimum POP3StartLatency in milliseconds.
MaxPOP3Stat Latency(msec)	Maximum POP3StartLatency in milliseconds.
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 (msec)	Interval between transmitting a Pop3 List packet and receiving a response in milliseconds.
MinPOP3List Latency(msec)	Minimum POP3ListLatency in milliseconds.
MaxPOP3List Latency(msec)	Maximum POP3ListLatency in milliseconds.
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 (msec)	Interval between transmitting a Pop3 Retr packet and receiving a response in milliseconds.

Measurement	Description
MinPOP3Retr Latency(msec)	Minimum POP3RetrLatency in milliseconds.
MaxPOP3Retr Latency(msec)	Maximum POP3RetrLatency in milliseconds.
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.
SuccSecondaryDNSR equest	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.
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 milliseconds.
MaxDNSData Latency	Maximum DNS Data Latency in milliseconds.
CurDNSData Latency	Interval between sending a DNS request packet and receiving a response in milliseconds.

Measurement	Description
DNSDataRequest Rate(/sec)	Number of DNS Request packets transmitted per second.
NoOf ReTransmission	Total number of DNS Request packets received
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)
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

Windows Resource Monitoring

The Windows Resources monitor displays statistics about the Windows resource usage during the load test run. The Windows measurements correspond to the built-in counters available from the Windows Performance monitor.

This chapter includes:

- Configuring the Windows Resources Monitor on page 76
- Windows Resources Performance Counters on page 79

Configuring the Windows Resources Monitor

Windows NT and Windows 2000 measurements correspond to the built-in counters available from the Windows Performance monitor. You select the Windows resources you want to monitor using the Windows Resources Choose Measurements page.

To monitor Windows resources, we recommend using the SiteScope monitor engine. In this case, SiteScope must be installed on a server. Although SiteScope is installed on the Utility Server, we recommend installing it on the same machine as the Controller, or on a dedicated server in order to reduce load on the Utility Server.

Note: To monitor a remote Windows machine that does not use Windows domain security, you must authenticate the Performance Center host machine on the remote Windows machine. To authenticate the Performance Center host machine, create an account, or change the password of the account used to log in to the Performance Center host so that it matches the password and user name used to log in 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.

To monitor a Windows NT or 2000 machine through a firewall, use TCP, port 139.

To configure the Windows Resources monitor:

1 Open the Add Monitor window.

- **Before running a load test:** In the Load Test configuration page's **Monitors** tab, select the type of monitor profile to create:
  - To create a monitor profile for the current load test, click **Add Local Profile**.
  - To create a monitor profile that can be used for other load tests, click **Add Profile**, type a name and description of the profile and click **OK**.

For details, see the section about configuring monitors in the *HP Performance Center User Guide*.

➤ **During a load test run:** On the Load Test run page:

➤ Click the **Design** button. The Load Test Design dialog box opens.

➤ Click **Monitors**.

For details, see the section about creating monitor profiles while running a load test in the *HP Performance Center User Guide*.

2 Under **System Resource Graphs**, select **Windows Resources** or **Windows Resources (SiteScope)**, then click **Next**. The Choose Server page opens.

3 Type the name or IP address of the server whose resources you want to monitor. In addition:

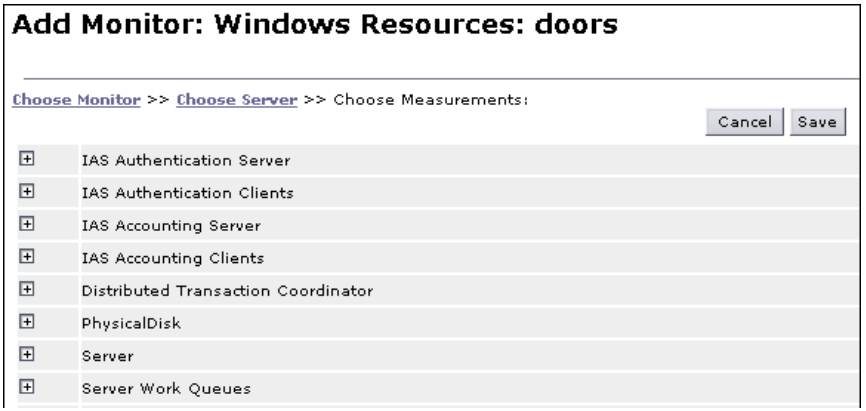
➤ For Windows Resources, type the user login name and password on the Windows machine.

➤ For Windows Resources (SiteScope), type the SiteScope machine name and port (default: 8888), and specify if you are using a Secure HTTP connection or a SiteScope user account. If you are using a SiteScope account, fill in the account name or number, and the user name and password defined to log in to the SiteScope account.

Click **Next**.

4 The first time you add measurements using the **Windows Resources (SiteScope)** monitor you need to configure the NT remote machine properties. Enter the remote machine's configuration information, as described in "Configuring the NT Remote Machine" on page 27.

5 The Choose Measurements page opens, displaying the available measurements and server properties.



6 Select the objects, counters, and instances that you want to monitor. The instance is relevant only if multiple instances of the highlighted counter are running.

For a list of available performance counters, see “Windows Resources Performance Counters” on page 79.

7 Click **Save** to add the selected resource measurements to the monitor profile.

Windows Resources Performance Counters

The following table describes some of the counters that can be measured:

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.

Object	Measurement	Description
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, then subtracting that value from 100%.
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.

Object	Measurement	Description
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.
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.

Object	Measurement	Description
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.

8

SNMP Resource Monitoring

The SNMP Resources monitor is available for monitoring any machine that runs an SNMP agent, using the Simple Network Management Protocol (SNMP). SNMP monitoring is platform independent.

This chapter includes:

- Configuring the SNMP Resources Monitor on page 83
- Configuring the SiteScope SNMP by MIB Monitor on page 87

Configuring the SNMP Resources Monitor

You select the SNMP resources you want to monitor using the SNMP Choose Measurements page.

To monitor SNMP resources, we recommend using the SiteScope monitor engine. In this case, SiteScope must be installed on a server. Although SiteScope is installed on the Utility Server, you should install it on the same machine as the Controller, or on a dedicated server in order to reduce load on the Utility Server.

Note: To monitor SNMP resources through a firewall, use ports 161 or 162.

To configure the SNMP Resources monitor:

1 Open the Add Monitor window.

- **Before running a load test:** In the Load Test configuration page's **Monitors** tab, select the type of monitor profile to create:

- To create a monitor profile for the current load test, click **Add Local Profile**.
- To create a monitor profile that can be used for other load tests, click **Add Profile**, type a name and description of the profile and click **OK**.

For details, see the section about configuring monitors in the *HP Performance Center User Guide*.

- **During a load test run:** On the Load Test run page:
  - Click the **Design** button. The Load Test Design dialog box opens.
  - Click **Monitors**.

  For details, see the section about creating monitor profiles while running a load test in the *HP Performance Center User Guide*.

2 Under **System Resource Graphs**, select an SNMP monitor:

- **SNMP (SiteScope)** - recommended
- **SNMP**

Click **Next**. The Choose Server page opens.

3 Type the name or IP address of the server whose resources you want to monitor. Performance Center connects to default SNMP port 161. To use a different port number, specify the machine name in the following format:

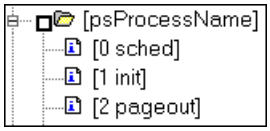
```
<server name>:<port number>
```

In addition:

- For **SNMP (SiteScope)**, type the SiteScope machine name and port (default: 8888), and specify if you are using a Secure HTTP connection or a SiteScope user account. If you are using a SiteScope account, fill in the account name or number, and the user name and password defined to log in to the SiteScope account.
- For SNMP object nodes with ten or more levels, click **Concatenate SNMP levels** to display the sub-levels as a single string, separated by periods (.). By default, the Choose Measurements page displays all nodes of the SNMP objects in a tree-level hierarchy.

To improve the level of measurement information for the SNMP monitor, click **Show String Item**. This lists measurements with string values (in addition to measurements with numeric values), and displays the string value as an identifying part of the measurement name.

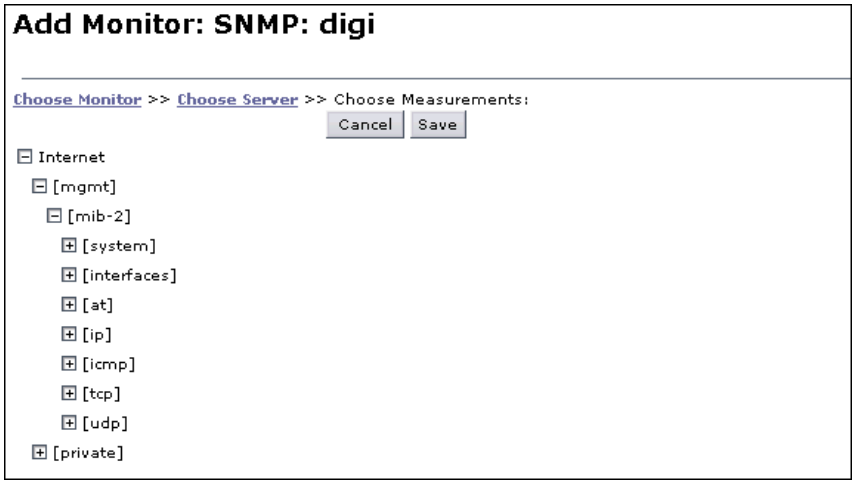
In the following example, the string value of ProcessName (sched) is displayed in addition to its instance ID (0):



Click **Next**.

4 The first time you add measurements using the **SNMP (SiteScope)** monitor you need to configure the monitor properties. Enter the monitor properties, as described in "Configuring the SiteScope SNMP by MIB Monitor" on page 87.

5 The Choose Measurements page opens, displaying the available measurements and server properties.



6 Select the measurements you want to monitor.

Note: The SNMP monitor can only monitor up to 25 measurements.

---

7 Click **Save** to add the selected resource measurements to the monitor profile.

You can modify the list of resources that you want to monitor at any point during the load test. Note that a load test does not have to be active in order for you to monitor the resources on a remote machine.

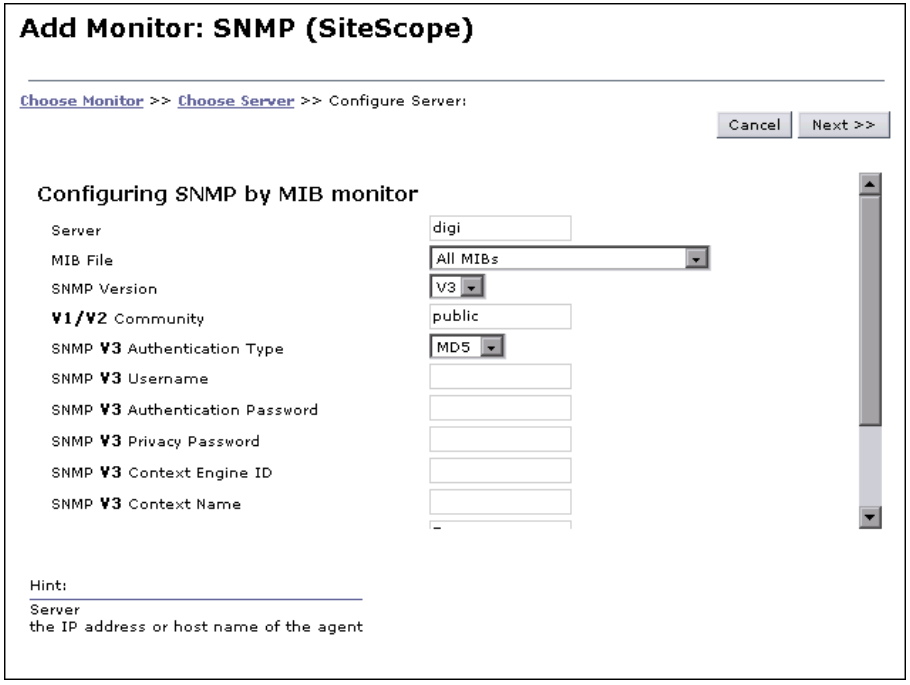
Note: It may take some time (approximately 2-3 minutes) to save measurements for the SNMP Resources monitor.

Configuring the SiteScope SNMP by MIB Monitor

The first time you add a measurement to the SNMP (SiteScope) monitor, you need to configure the monitor properties. After you have configured the monitor properties, you select the measurements to monitor on the machine.

To configure the SNMP by MIB monitor:

1 After you have selected the SNMP (SiteScope) monitor, and entered the monitor information, click **Next**. The Configuring SNMP by MIB Monitor page opens.



2 Type values for the monitor properties:

- **Server**. Type the name of the server you want to monitor.
- **MIB File**. Select the MIB file which contains the objects you are interested in monitoring. If you select a specific MIB file, then only the objects described in that MIB file are displayed. If you select **All MIBs**, then all objects retrieved from the agent during the MIB traversal will be displayed.

  If no MIB information is available for an object, it is still displayed, but with no textual name or description. To make this monitor aware of new or additional MIBs, simply place new MIB files in the **SiteScope/templates.mib** directory.
- **SNMP Version**. Select the version of SNMP to use when connecting.
- **V1/V2 Community**. Type the community string (valid only for version 1 or 2 connections).
- **SNMP V3 Authentication Type**. Select the type of authentication to use for version 3 connections.
- **SNMP V3 Username**. Type the user name for version 3 connections.
- **SNMP V3 Authentication Password**. Type the authentication password to use for version 3 connections.
- **SNMP V3 Privacy Password**. Type the privacy password if DES privacy encryption is desired for version 3 connections. Leave blank if you do not want privacy.
- **SNMP V3 Context Engine ID**. Type a hexadecimal string representing the Context Engine ID to use for this connection. This is applicable for SNMP V3 only.
- **SNMP V3 Context Name**. Type the Context Name to use for this connection. This is applicable for SNMP V3 only.
- **Timeout**. Type the total time, in seconds, that SiteScope should wait for all SNMP requests (including retries) to complete. The default is 5 seconds.
- **Retries**. Type the number of times each SNMP GET request should be retried before SiteScope considers the request to have failed. The default is 1 retry.

- **Port**. Type the port to use when requesting data from the SNMP agent. The default of 161 is the port on which an SNMP agent will typically be listening.
- **Update every**. Type how frequently the monitor should read the server statistics. The list to the right of the text box lets you specify time increments of seconds, minutes, hours, or days. You must specify a time increment of at least 15 seconds. The default is 30 seconds.

3 Click **Next**. Add the measurements you want to monitor as described in step 5 of "Configuring the SNMP Resources Monitor" on page 83.

9

SiteScope Resource Monitoring

The SiteScope Resources monitor graph shows the SiteScope resources measured during the load test.

Note: You can use SiteScope 8.8 or later to monitor multiple instances of the same process.

The SiteScope monitor can measure server, network, and processor performance counters. SiteScope can also be used to monitor SAP CCMS (Computer Center Management System) and SNMP server resources. For detailed information on all the performance counters that SiteScope can monitor, see the relevant SiteScope documentation.

This chapter includes:

- Before Setting Up the SiteScope Monitor on page 91
- Configuring the SiteScope Monitor on page 92

Before Setting Up the SiteScope Monitor

Before you configure the SiteScope monitor:

- Make sure that SiteScope has been installed on a server. Although you can install SiteScope on the Performance Center host machine, we recommend that you install it on a dedicated server.

➤ On the machine where SiteScope is installed, configure SiteScope to monitor the required servers. When you assign a name to a monitor, include the server name in the monitor name. This avoids any confusion as to which host the monitor belongs. Note that the monitor title cannot include "\" (for example, CPU Utilization on \\superman).

Note: SiteScope's default sampling rate is 10 minutes, and its minimum rate is 15 seconds.

➤ Verify that SiteScope is collecting the required data from the servers it is monitoring. From the SiteScope panel, select the monitor group polling the server machines, and check that the monitor displays a list of server measurements in the Status column.

Configuring the SiteScope Monitor

You select the SiteScope resources you want to monitor using the SiteScope Choose Measurements page.

To configure the SiteScope monitor:

1 Open the Add Monitor window.

➤ **Before running a load test:** In the Load Test configuration page's **Monitors** tab, select the type of monitor profile to create:

➤ To create a monitor profile for the current load test, click **Add Local Profile**.

➤ To create a monitor profile that can be used for other load tests, click **Add Profile**, type a name and description of the profile and click **OK**.

For details, see the section about configuring monitors in the *HP Performance Center User Guide*.

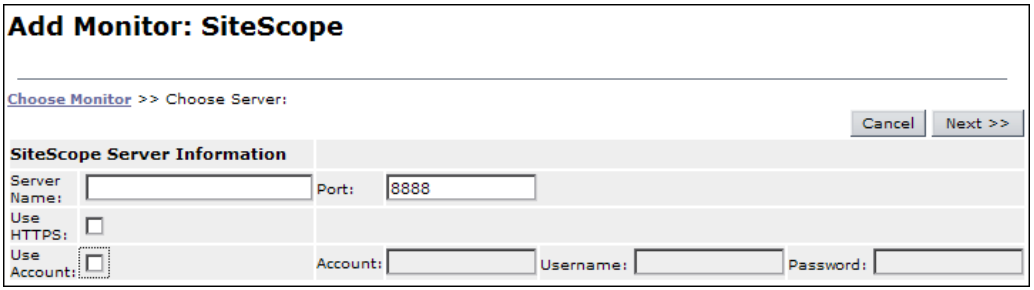
➤ **During a load test run:** On the Load Test run page:

➤ Click the **Design** button. The Load Test Design dialog box opens.

➤ Click **Monitors**.

For details, see the section about creating monitor profiles while running a load test in the *HP Performance Center User Guide*.

2 Under **System Resource Graphs**, select **SiteScope**, and click **Next**. The Choose Server page opens.



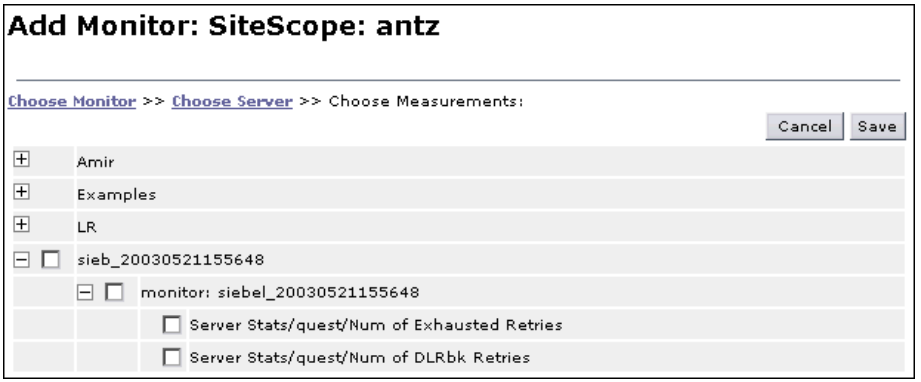
3 Type the SiteScope Server information as follows:

- **Server Name.** Type the SiteScope server name.
- **Port.** Type the SiteScope port (default: 8888).
- **Use HTTPS.** Specify if you are using a Secure HTTP connection.
- **Use Account.** By default, Performance Center connects to SiteScope using the default SiteScope Administrator user. If the default SiteScope Administrator password was changed, or if you want to connect using a different SiteScope user, select this option. Type the account name, and the user name and password defined to log in to the relevant SiteScope account.

Note: The administrator account name is **administrator**. For details about how to identify other SiteScope account names, see “Identifying the SiteScope Account Name” on page 94.

---

4 Click **Next**. The Choose Measurements page opens, displaying the available measurements and server properties.



5 Select the measurements you want to monitor.

6 Click **Save** to add the selected resource measurements to the monitor profile.

Note: During configuration of the SiteScope monitor, after selecting the measurements to monitor and clicking **Save**, instead of saving the monitor, Performance Center might go back to the Choose Monitor page. To prevent this from happening, you need to increase the value of the **AspMaxRequestEntityAllowed** property in the IIS metabase. For more information on troubleshooting this issue, see "Cannot Save Measurements" on page 417.

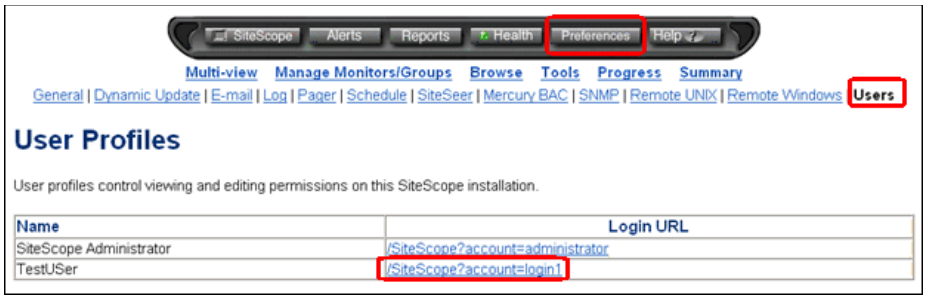
Identifying the SiteScope Account Name

This section describes how to identify the Sitescope account name for a particular SiteScope user.

To identify a SiteScope account name:

1 Log on to SiteScope as follows: **<SiteScope server>:8888/sitescope**.

2 Click the **Preferences** button, and then select the **Users** link.



3 The User Profiles page shows a table that displays the Login URL associated with each user name. Inside the Login URL you can identify the account name. In the above example, the account name for the user, **TestUSer** is **login1**.

10

Server Resource Monitoring

The Server Resources monitor shows the resources of monitors used on remote Windows and UNIX servers measured during the load test run, enabling you to determine the impact of Vuser load on the various server resources.

This chapter includes:

- Setting Up the Server Resources Monitor on page 98
- Configuring the Server Resources Monitor on page 98
- Server Resources Performance Counters on page 100

Setting Up the Server Resources Monitor

To obtain Server Resources data, you need to enable the Server Resources online monitor (from the Performance Center host) before running the load test, and indicate which statistics and measurements you want to monitor.

SiteScope must be installed on a server. Although SiteScope is installed on the Utility Server, we recommend installing it on the same machine as the Controller, or on a dedicated server in order to reduce load on the Utility Server.

Configuring the Server Resources Monitor

The Server Resources monitor includes the following monitors:

- **CPU Monitor**. Monitors CPU usage.
- **Disk Space Monitor**. Monitors disk space.
- **Memory Monitor**. Monitors pages per second and percentage of virtual memory used.
- **Service Monitor**. Verifies that specific processes are listed as running and checks CPU usage.

To configure the Server Resource monitor:

1 Open the Add Monitor window.

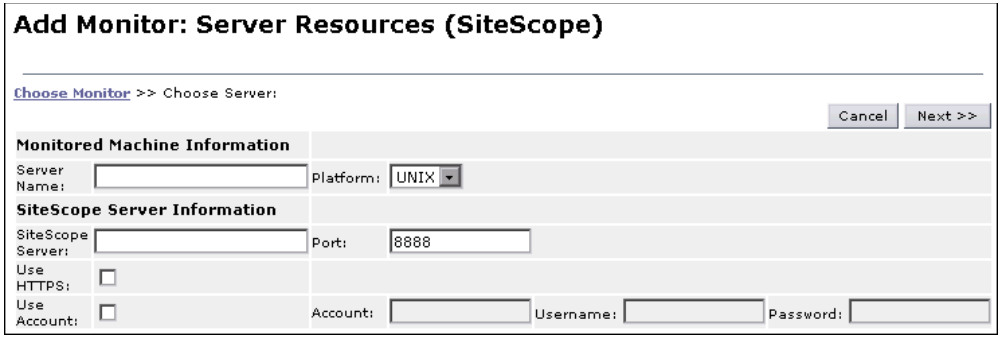
- **Before running a load test:** In the Load Test configuration page's **Monitors** tab, select the type of monitor profile to create:
  - To create a monitor profile for the current load test, click **Add Local Profile**.
  - To create a monitor profile that can be used for other load tests, click **Add Profile**, type a name and description of the profile and click **OK**.

  For details, see the section about configuring monitors in the *HP Performance Center User Guide*.
- **During a load test run:** On the Load Test run page:
  - Click the **Design** button. The Load Test Design dialog box opens.

➤ Click **Monitors**.

For details, see the section about creating monitor profiles while running a load test in the *HP Performance Center User Guide*.

2 Under **System Resource Graphs**, select **Server Resources (SiteScope)** and click **Next**. The Choose Server page opens.



3 Under **Monitored Machine Information**, type the name or IP address of the server whose resources you want to monitor, and the platform on which the server runs.

Under **SiteScope Server Information**, type the SiteScope machine name and port (default: 8888), and specify if you are using a Secure HTTP connection or a SiteScope user account. If you are using a SiteScope account, fill in the account name or number, and the user name and password defined to log in to the SiteScope account.

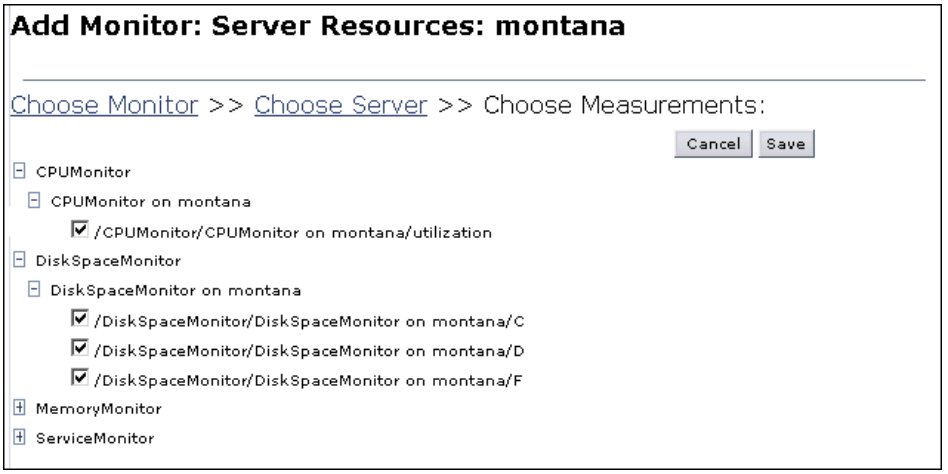
Click **Next**.

4 The first time you add measurements using the **Server Resources (SiteScope)** monitor you need to configure the remote machine properties.

- If you are monitoring a machine on an NT platform, the Configuring NT Remote Machine page opens.
- If you are monitoring a machine on a UNIX platform, the Configuring UNIX Remote Machine page opens.

Type the remote machine's configuration information, as described in "Configuring the Remote Machine for SiteScope Monitors" on page 26.

5 Click **Next**. The Choose Measurements page opens, displaying the available measurements and server properties.



6 Select the measurements you want to monitor.

7 Click **Save** to add the selected resource measurements to the monitor profile.

Server Resources Performance Counters

The following default measurements are available for the Server Resources machine:

Monitor	Measurements	Description
CPU Monitor	**Utilization**	Measures CPU utilization.
Disk Space Monitor	**Disk space**	Measures the percentage of disk space used.

Monitor	Measurements	Description
Memory Monitor	**MB free**	Measures the amount of disk space free, in MB.
	Pages/sec	Measures the number of virtual memory pages that are moved between main memory and disk storage.
	Percent used	Measures the percentage of memory and paging file space used.
Services Monitor		Monitors processes locally or on remote systems. Can be used to verify that specific processes are running.

11

UNIX Resource Monitoring

The UNIX Resources monitor shows the UNIX resources measured during the load test, and enables you to determine the impact of Vuser load on the various system resources.

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.

This chapter includes:

- Setting Up the Monitoring Environment on page 104
- Configuring the UNIX Resources Monitor on page 106
- UNIX Resources Performance Counters on page 108

Setting Up the Monitoring Environment

To monitor UNIX resources, you must configure the rstatd daemon. 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 you are prompted to include a version number, specify versions 3 and 4.

Configuring the UNIX Resources Monitor

You select the UNIX resources you want to monitor using the UNIX Resources Choose Measurements page.

To configure the UNIX Resources monitor:

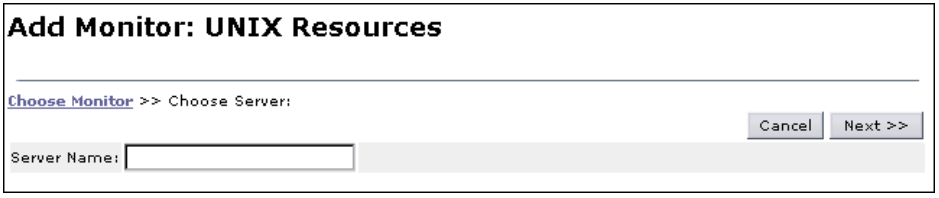
1 Open the Add Monitor window.

- **Before running a load test:** In the Load Test configuration page's **Monitors** tab, select the type of monitor profile to create:
  - To create a monitor profile for the current load test, click **Add Local Profile**.
  - To create a monitor profile that can be used for other load tests, click **Add Profile**, type a name and description of the profile and click **OK**.

  For details, see the section about configuring monitors in the *HP Performance Center User Guide*.
- **During a load test run:** On the Load Test run page:
  - Click the **Design** button. The Load Test Design dialog box opens.
  - Click **Monitors**.

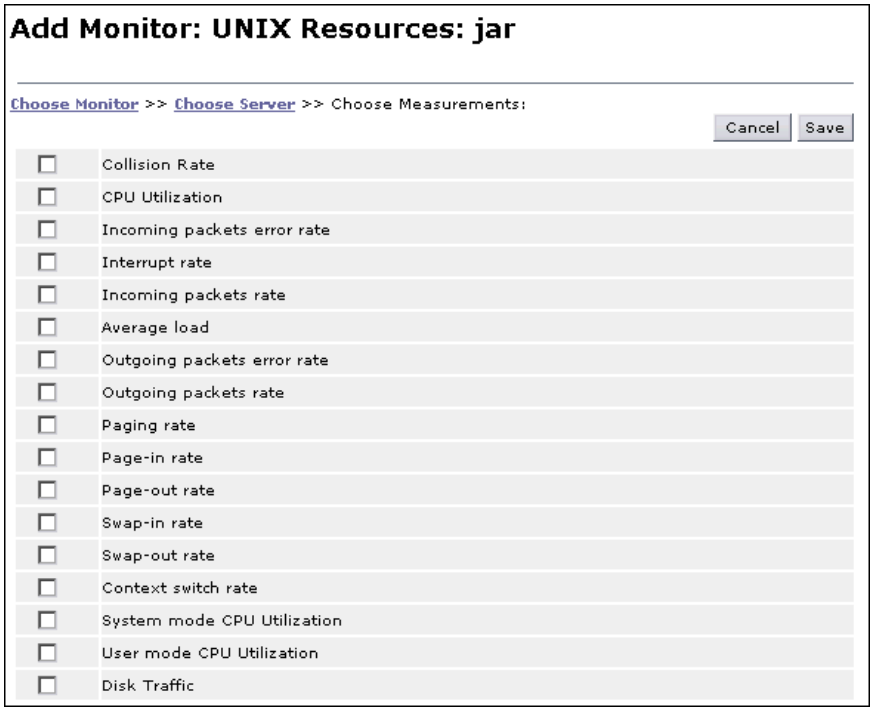
  For details, see the section about creating monitor profiles while running a load test in the *HP Performance Center User Guide*.

2 Under **System Resource Graphs**, select **UNIX Resources** and click **Next**. The Choose Server page opens.



3 Type the name or IP address of the server whose resources you want to monitor.

4 Click **Next**. The Choose Measurements page opens, displaying the available measurements and server properties.



5 Select the measurements you want to monitor.

For a list of available performance counters, see "UNIX Resources Performance Counters" on page 108.

6 Click **Save** to add the selected resource measurements to the monitor profile.

UNIX Resources Performance Counters

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 pagefiles and removed from physical memory, per second
Paging rate	Number of pages read to physical memory or written to pagefiles, 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

Part V

Network Monitoring

12

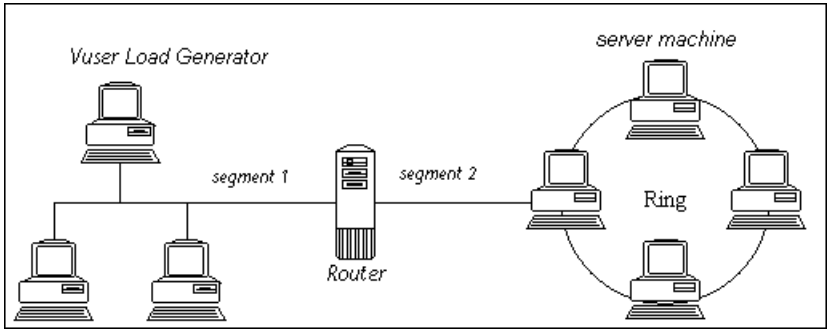
Introducing Network Monitoring

You can use Network monitoring to determine whether your network is causing a delay in the load test. You can also determine the problematic network segment.

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. 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.

13

Network Delay Monitoring

You can use Network monitoring to determine whether your network is causing a delay in the load test. You can also determine the problematic network segment.

This chapter includes:

- Network Monitoring from a UNIX Source Machine on page 113
- Configuring the Network Delay Time Monitor on page 117
- Network Monitoring over a Firewall on page 121

Note: To run the Network monitor, you must have administrator privileges on the Windows source machine (unless you are using the ICMP protocol).

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 UNIX source machine by assigning root permissions to the **merc_webtrace** process (see "Configuring the UNIX Source Machine" on page 114).
- Make the necessary adjustments to either connect to the source machine through RSH (see "Connecting to the UNIX Source Machine Through RSH" on page 115), or through the agent (see "Connecting to the UNIX Source Machine Through the Agent" on page 116).

Configuring the UNIX Source Machine

When Performance Center is installed locally, assign root permissions to the **merc_webtrace** process by adding an s-bit to **merc_webtrace**'s permissions

To configure the source machine, where Performance Center is installed locally:

1. Log in to the source machine as root.
2. Type: cd <Performance Center 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.

When Performance Center is installed on a network, the **merc_webtrace** process is on the network, not on the source machine disk. The procedure described below copies the **merc_webtrace** file to the local disk, configures **mdrv.dat** to recognize the process, and assigns root permissions to **merc_webtrace**.

To configure the source machine, where Performance Center is installed on the network:

1. Copy **merc_webtrace** from <**Performance Center installation**>/**bin** to anywhere on the local disk of the source machine. For example, to copy the file to the **/local/**<**Performance Center**> directory, type:

```
cp /net/tools/<Performance Center installation>/bin/merc_webtrace
/local/<Performance Center>
```

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/<Performance Center>), since all of them use the same **mdrv.dat**.

---

2 Add the following line to the <**Performance Center 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 <Performance Center 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 UNIX Source Machine Through RSH

If the Performance Center Host is connected to the source machine through RSH (default connection mode), then you do not need to activate the agent daemon. Before running the Network monitor the first time, you type an encrypted user name and password in the Network monitor configuration file.

To create an encrypted user name and password:

1 On the Performance Center Host machine, type: cd <Performance Center installation>/bin to change to the **bin** directory.

Note: In a network or workstation installation, <Performance Center installation>/bin is the location on the network in which you installed the Performance Center setup files.

---

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 <**Performance Center installation**>**/dat/monitors/ndm.cfg** file, in the [hosts] section:

```
Host = <encrypted string copied from clipboard>
```

6 Close and open the current load test. Performance Center will read the updated configuration file and recognize the source machine for monitoring.

Connecting to the UNIX Source Machine Through the Agent

If the Performance Center Host 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.

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 <**Performance Center installation**>**/bin** directory.

2 Make sure that the agent daemon is running whenever you activate the Network monitor.

3 To stop the Network Delay Monitor agent daemon, type:

```
m_daemon_setup - remove
```

Configuring the Network Delay Time Monitor

The Network Delay Time monitor shows the delays for the complete path between the source and destination machines (for example the database server and Vuser host). The graph maps the delay as a function of the elapsed load test time.

You configure the Network monitor from the Monitor tab of the Load Tests page before you begin running a load test. Using the Network Delay Monitor page, you select the network path you want to monitor and configure the network delay monitor settings. You can also configure the network settings from the Network Delay Monitor Properties page.

Note: To enable network monitoring, you must install the Performance Center agent on the source machine. You do not have to install the Performance Center agent on the destination machine.

To configure the Network Delay Time monitor:

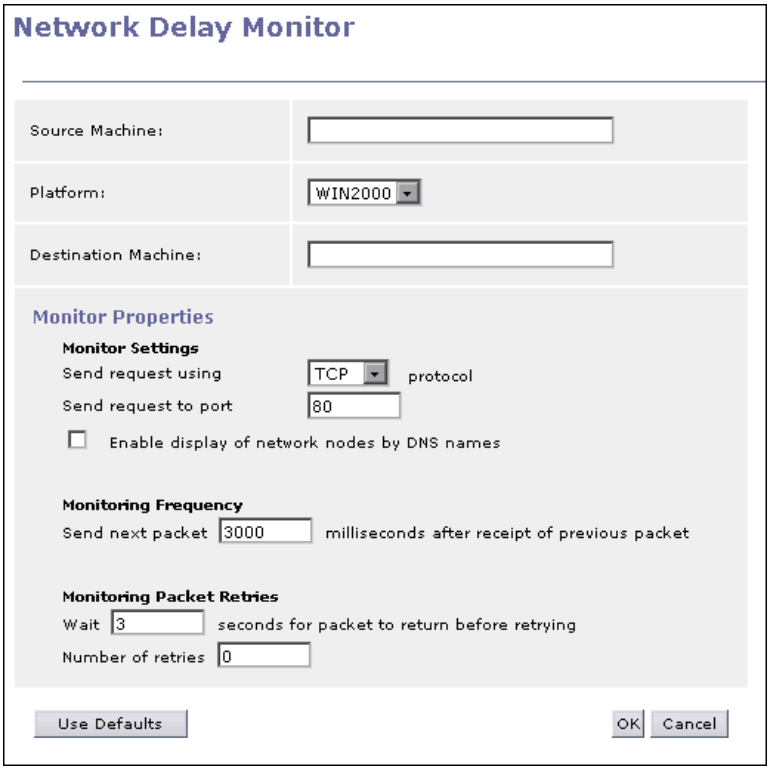
1 Open the Add Monitor window.

- **Before running a load test:** In the Load Test configuration page's **Monitors** tab, select the type of monitor profile to create:
  - To create a monitor profile for the current load test, click **Add Local Profile**.
  - To create a monitor profile that can be used for other load tests, click **Add Profile**, type a name and description of the profile and click **OK**.

  For details, see the section about configuring monitors in the *HP Performance Center User Guide*.
- **During a load test run:** On the Load Test run page:
  - Click the **Design** button. The Load Test Design dialog box opens.
  - Click **Monitors**.

  For details, see the section about creating monitor profiles while running a load test in the *HP Performance Center User Guide*.

2 Under **Network Graphs**, select **Network Delay Time** and click **Next**. The Network Delay Monitor page opens.



3 In the **Source Machine** box, type the name of the source machine, from which you want the network path monitoring to begin.

To run the Network Delay Monitor when there are firewalls between the Performance Center Host machine and the source machine, in the **Source Machine** box, type the server name or IP address of the source machine according to the format:

```
<MI Listener machine>:<source machine local key>
```

where source machine local key is the unique key that you chose when configuring the Performance Center agent on the source machine.

For example,

```
12.12.12.3:vds
```

For more information about working with firewalls in Performance Center, see the section about firewalls in the *HP Performance Center System Configuration and Installation Guide*.

4 Select the platform on which the machine runs.

5 In the **Destination Machine** box, type the name of the machine at the final destination of the path you want to monitor.

Note: If the destination machine is **localhost**, type the local machine's name and not **localhost**.

---

6 In the Monitor Settings box, select the protocol and type the port number being used by the network path. The Network monitor supports three protocols: TCP, UDP, and ICMP. The default protocol is recommended. In Windows, the default is TCP, and in UNIX, the default is UDP.

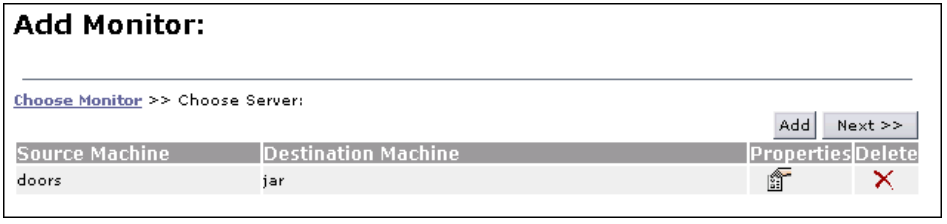
7 Select **Enable display of network nodes by DNS names** to view the DNS name of each node along the network path, in addition to its IP address. Note that selecting this option will decrease the speed of the Network monitor.

8 In the Monitoring Frequency box, select the number of milliseconds the monitor should wait between receiving a packet and sending out the next packet. The default value is 3000 milliseconds. If you have a long, steady load test, you can increase the interval by several seconds.

9 In the Monitoring Packet Retries box, select the maximum number of seconds that the monitor should wait for a packet to return before it retries to send the packet. The default value is 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.

10 Click **OK** to close the Network Delay Monitor page and save your settings. The name of the source and destination machines appears in the Choose Server page.



11 Click **Add** and repeat this procedure for each path you want to monitor.

To delete a machine, select it and click the **Delete** button.

To edit the Monitor Settings, click the **Edit Profile** button. The Network Delay Monitor Properties page opens. You can modify any of the monitor settings explained in steps 6 through 9.

12 Click **Next**, then click **Add** to add the Network Delay Monitor settings to the monitor profile.

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, check 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.

Part VI

Firewall Server Monitoring

14

Firewall Server Performance Monitoring

To isolate server performance bottlenecks, you can monitor the firewall server during a load test run.

This chapter includes:

➤ About the Firewall Server Monitor on page 125

➤ Configuring the Check Point FireWall-1 Server Monitor on page 125

➤ Check Point FireWall-1 Performance Counters on page 127

About the Firewall Server Monitor

The Firewall server online monitor measures the performance of a Firewall server during load test execution. To obtain performance data, you must activate the Firewall server monitor (before executing the load test) 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.

To configure the Check Point FireWall-1 server monitor:

1 Open the Add Monitor window.

➤ **Before running a load test:** In the Load Test configuration page's **Monitors** tab, select the type of monitor profile to create:

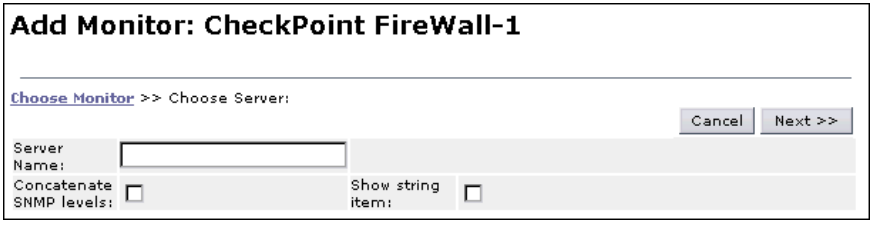
- To create a monitor profile for the current load test, click **Add Local Profile**.
- To create a monitor profile that can be used for other load tests, click **Add Profile**, type a name and description of the profile and click **OK**.

For details, see the section about configuring monitors in the *HP Performance Center User Guide*.

- **During a load test run:** On the Load Test run page:
  - Click the **Design** button. The Load Test Design dialog box opens.
  - Click **Monitors**.

  For details, see the section about creating monitor profiles while running a load test in the *HP Performance Center User Guide*.

2 Under **Firewalls**, select **Check Point FireWall-1** and click **Next**. The Choose Server page opens.

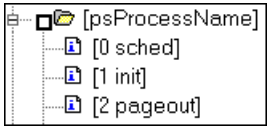


3 Type the name or IP address of the server whose resources you want to monitor. Performance Center connects to port 260, the default port for the Check Point FireWall-1 SNMP agent. If you do not want to connect to the default port, specify a machine name and port number using the format: <machine name>:<port number>

For object nodes with ten or more levels, click **Concatenate SNMP levels** to display the sub-levels as a single string, separated by periods (.). By default, the Choose Measurements page displays all nodes of the SNMP objects in a tree-level hierarchy.

To improve the level of measurement information for the SNMP monitor, click **Show String Item**. This enable measurements with string values to be listed (in addition to measurements with numeric values), and enables 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):



4. Click **Next**. The Choose Measurements page opens, displaying the available measurements and server properties.
5. Select the measurements you want to monitor.

   For a list of the available performance counters, see "Check Point FireWall-1 Performance Counters" on page 127.

Note: The Check Point FireWall-1 monitor can only monitor up to 25 measurements.

---

6. Click **Save** to add the selected resource measurements to the monitor profile.

Check Point FireWall-1 Performance Counters

The following default counters can be monitored:

Measurement	Description
fwRejected	The number of rejected packets.

Measurement	Description
fwDropped	The number of dropped packets.
fwLogged	The number of logged packets.

Part VII

Web Server Monitoring

15

Introducing Web Server Monitoring

Using Performance Center's Web Server Resource monitors, you can monitor the Apache, Microsoft IIS, iPlanet (SNMP), and iPlanet/Netscape servers during a load test run and isolate server performance bottlenecks.

This chapter includes:

- About Web Server Resource Monitors on page 131
- Monitoring Using a Proxy Server on page 132

About Web Server Resource Monitors

Web Server Resource monitors provide you with information about the resource usage of the Apache, Microsoft IIS, iPlanet (SNMP), and iPlanet/Netscape Web servers during load test execution. 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 load test.

Monitoring Using a Proxy Server

To monitor using the Apache and Netscape monitors, there must be a proxy server between the Performance Center Host and the monitored server. To enable this, you must define settings in the relevant configuration files, as follows:

Monitor	Configuration File
Apache	**<Performance Center root folder>\dat\monitors\apache.cfg**
Netscape	**<Performance Center root folder>\dat\monitors\Netscape.cfg**

Before defining settings, you need to determine whether you want Performance Center to obtain proxy settings from your Internet Explorer connection configuration, or from the proxy settings in the configuration file.

To have Performance Center 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 user name, password, or domain, type these parameters on the lines **proxyUsername**, **proxyPassword**, and **proxyDomain**.

To have Performance Center read proxy settings from the configuration file:

1. In the Proxy Settings section of the configuration file, type 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 user name, password, or domain, type these parameters on the lines: **proxyUsername**, **proxyPassword**, and **proxyDomain**.

To have Performance Center 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.

16

Apache Resource Monitoring

The Apache monitor displays statistics about the resource usage on the Apache server during the load test run.

This chapter includes:

➤ Configuring the Apache Monitor on page 135

➤ Apache Performance Counters on page 137

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 Open the Add Monitor window.

➤ **Before running a load test:** In the Load Test configuration page's **Monitors** tab, select the type of monitor profile to create:

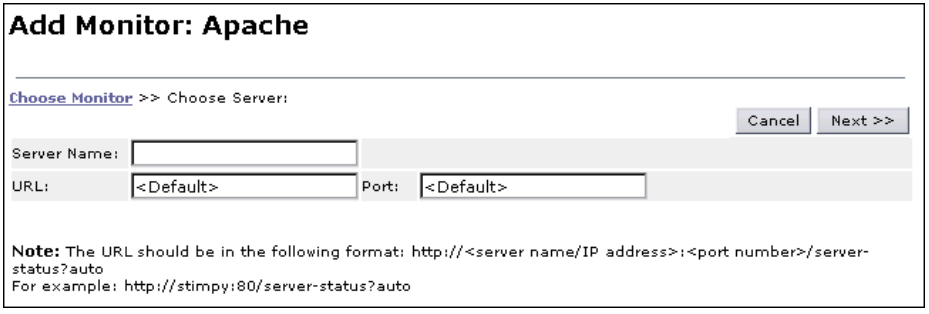
- To create a monitor profile for the current load test, click **Add Local Profile**.
- To create a monitor profile that can be used for other load tests, click **Add Profile**, type a name and description of the profile and click **OK**.

For details, see the section about configuring monitors in the *HP Performance Center User Guide*.

- **During a load test run:** On the Load Test run page:
  - Click the **Design** button. The Load Test Design dialog box opens.
  - Click **Monitors**.

  For details, see the section about creating monitor profiles while running a load test in the *HP Performance Center User Guide*.

2 Under **Web Server Resource Graphs**, select **Apache** and click **Next**. The Choose Server page opens.



3 Type the name or IP address of the server whose resources you want to monitor.

In the **URL** box, type the URL, or accept the default URL:

```
(/server-status?auto)
```

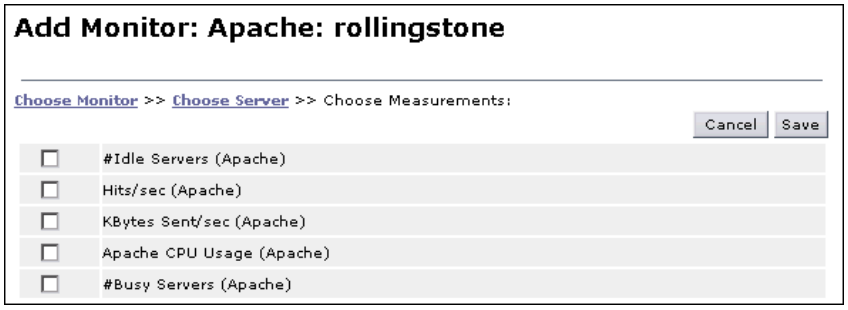
In the **Port** box, type the Port number (without the server name), or accept the default port.

Note: The default port number and URL can vary from one server to another. Contact your Web server administrator.

To monitor an Apache server through a firewall, use the Web server port (by default, port 80).

---

4 Click **Next**. The Choose Measurements page opens, displaying the available measurements and server properties.



5 Select the measurements you want to monitor.

For a list of the available performance counters, see "Apache Performance Counters" on page 137.

6 Click **Save** to add the selected resource measurements to the monitor profile.

Apache Performance Counters

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

Measurement	Description
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

17

Microsoft IIS Monitoring

The Microsoft IIS monitor displays statistics about the resource usage on the Microsoft IIS server during the load test run.

This chapter includes:

➤ Configuring the Microsoft IIS Monitor on page 139

➤ Microsoft IIS Performance Counters on page 142

Configuring the Microsoft IIS Monitor

You select measurements to monitor the Microsoft IIS server from the Microsoft IIS Choose Measurements page.

To configure the IIS server monitor:

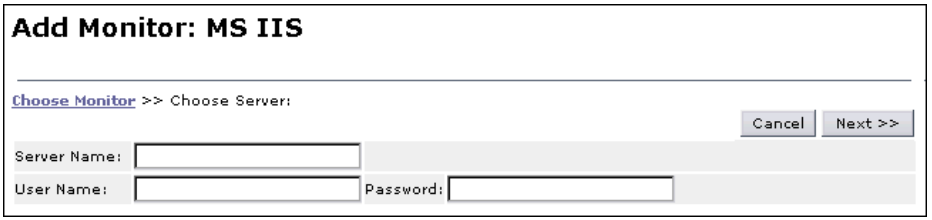
1 Open the Add Monitor window.

- **Before running a load test:** In the Load Test configuration page's **Monitors** tab, select the type of monitor profile to create:
  - To create a monitor profile for the current load test, click **Add Local Profile**.
  - To create a monitor profile that can be used for other load tests, click **Add Profile**, type a name and description of the profile and click **OK**.

  For details, see the section about configuring monitors in the *HP Performance Center User Guide*.
- **During a load test run:** On the Load Test run page:
  - Click the **Design** button. The Load Test Design dialog box opens.
  - Click **Monitors**.

For details, see the section about creating monitor profiles while running a load test in the *HP Performance Center User Guide*.

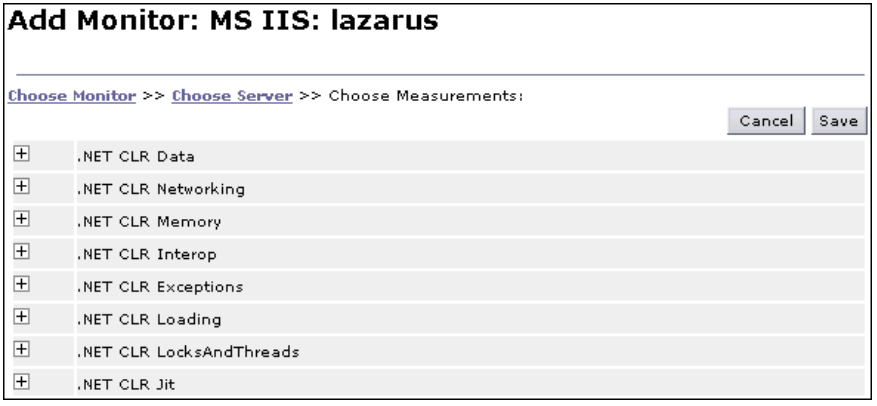
2 Under **Web Server Resource Graphs**, select **MS IIS** and click **Next**. The Choose Server page opens.



3 Type the name or IP address of the server whose resources you want to monitor.

4 Type the user login name and password.

5 Click **Next**. The Choose Measurements page opens, displaying the available measurements and server properties.



6 Select the objects, counters, and instances that you want to monitor.

For a list of the available performance counters, see "Microsoft IIS Performance Counters" on page 142.

7 Click **Save** to add the selected resource measurements to the monitor profile.

Note: To monitor an IIS server through a firewall, use TCP port 139.

Microsoft IIS Performance Counters

The following table describes some of the counters that can be measured:

Object	Measurement	Description
Web Service	**Bytes Sent/sec**	The rate at which the data bytes are sent by the Web service
Web Service	**Bytes Received/sec**	The rate at which the data bytes are received by the Web service
Web Service	**Get Requests/sec**	The rate at which HTTP requests using the GET method are made. Get requests are generally used for basic file retrievals or image maps, though they can be used with forms.
Web Service	**Post Requests/sec**	The rate at which HTTP requests using the POST method are made. Post requests are generally used for forms or gateway requests.
Web Service	**Maximum Connections**	The maximum number of simultaneous connections established with the Web service
Web Service	**Current Connections**	The current number of connections established with the Web service
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.

18

iPlanet/Netscape Monitoring

The iPlanet/Netscape monitor displays statistics about the resource usage on the iPlanet/Netscape server during the load test run.

This chapter includes:

- Configuring the iPlanet/Netscape Monitor on page 144
- iPlanet/Netscape Performance Counters on page 147

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
```

Notes:

- 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 monitor an iPlanet/Netscape server through a firewall, use the iPlanet/Netscape Administration server port. Configure this port during the server installation process.

To activate the iPlanet/Netscape monitor from the Performance Center Host:

1 Open the Add Monitor window.

- **Before running a load test:** In the Load Test configuration page's **Monitors** tab, select the type of monitor profile to create:
  - To create a monitor profile for the current load test, click **Add Local Profile**.
  - To create a monitor profile that can be used for other load tests, click **Add Profile**, type a name and description of the profile and click **OK**.

For details, see the section about configuring monitors in the *HP Performance Center User Guide*.

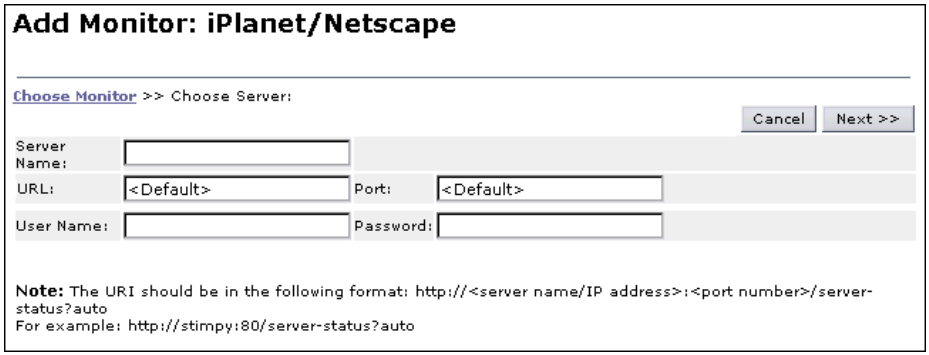
➤ **During a load test run:** On the Load Test run page:

➤ Click the **Design** button. The Load Test Design dialog box opens.

➤ Click **Monitors**.

For details, see the section about creating monitor profiles while running a load test in the *HP Performance Center User Guide*.

2 Under **Web Server Resource Graphs**, select **iPlanet/Netscape** and click **Next**. The Choose Server page opens.



3 Type the name or IP address of the server whose resources you want to monitor.

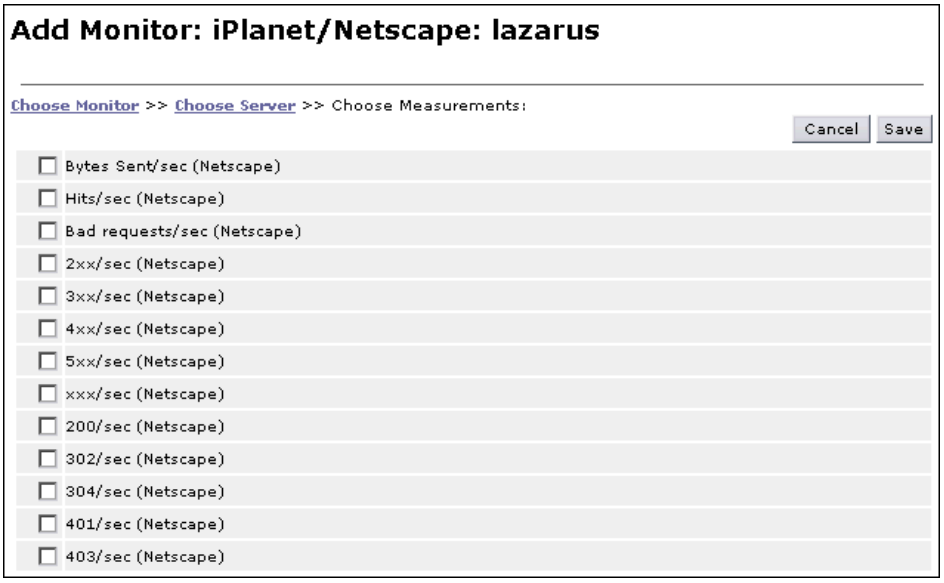
Note: The default port number and URL can vary from one server to another. Please consult the Web server administrator. In some server configurations, the URL must contain the administration server name and not the IP address.

➤ In the **URL** box, type the URL or accept the default URL,

```
(/https-<admin_server>/bin/sitemon?doit)
```

- In the **Port** box, type the Port number (without the server name), or accept the default port.
- Type the user login name and password. The user must have administrator permissions on the server.

4 Click **Next**. The Choose Measurements page opens, displaying the available measurements and server properties.



5 Select the required measurements.

For a list of the available performance counters, see "iPlanet/Netscape Performance Counters" on page 147.

6 Click **Save** to add the selected resource measurements to the monitor profile.

iPlanet/Netscape Performance Counters

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
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

19

iPlanet (SNMP) Monitoring

The iPlanet (SNMP) monitor uses the Simple Network Management Protocol (SNMP) to retrieve iPlanet (SNMP) server statistics.

This chapter includes:

➤ Configuring the iPlanet (SNMP) Monitor on page 149

➤ iPlanet (SNMP) Performance Counters on page 151

Configuring the iPlanet (SNMP) Monitor

You select measurements to monitor the iPlanet (SNMP) server using the iPlanet (SNMP) Choose Measurements page.

Note: To monitor a iPlanet (SNMP) server, use port 161 or 162, depending on the configuration of the agent.

To configure the iPlanet (SNMP) Resources monitor:

1 Open the Add Monitor window.

➤ **Before running a load test:** In the Load Test configuration page's **Monitors** tab, select the type of monitor profile to create:

➤ To create a monitor profile for the current load test, click **Add Local Profile**.

➤ To create a monitor profile that can be used for other load tests, click **Add Profile**, type a name and description of the profile and click **OK**.

For details, see the section about configuring monitors in the *HP Performance Center User Guide*.

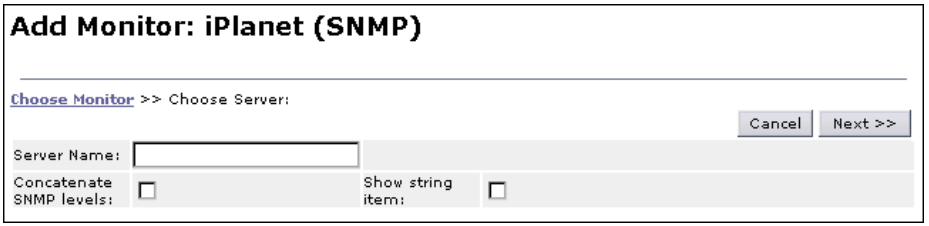
➤ **During a load test run:** On the Load Test run page:

➤ Click the **Design** button. The Load Test Design dialog box opens.

➤ Click **Monitors**.

For details, see the section about creating monitor profiles while running a load test in the *HP Performance Center User Guide*.

2 Under **Web Server Resource Graphs**, select **iPlanet (SNMP)** and click **Next**. The Choose Server page opens.



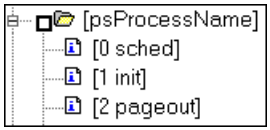
3 Type the name or IP address of the server whose resources you want to monitor. You need to define the port number if the iPlanet SNMP agent is running on a different port than the default SNMP port. Specify the machine name and port number in the format:

```
<server name>:<port number>
```

For object nodes with ten or more levels, click **Concatenate SNMP levels** to display the sub-levels as a single string, separated by periods (.). By default, the Choose Measurements page displays all nodes of the SNMP objects in a tree-level hierarchy.

To improve the level of measurement information for the iPlanet SNMP monitor, click **Show String Item**. This lists measurements with string values (in addition to measurements with numeric values), and displays the string value as an identifying part of the measurement name.

In the following example, the string value of ProcessName (sched) is displayed in addition to its instance ID (0):



4 Click **Next**. The Choose Measurements page opens, displaying the available measurements and server properties.

5 Select the measurements you want to monitor.

For a list of the available performance counters, see "iPlanet (SNMP) Performance Counters" on page 151.

Note: The iPlanet (SNMP) monitor can monitor up to 25 measurements at a time.

---

6 Click **Save** to add the selected resource measurements to the monitor profile.

iPlanet (SNMP) Performance Counters

The following table describes the measurements and server properties that can be monitored:

Measurement	Description
iwsInstanceTable	iPlanet Web Server instances
iwsInstanceEntry	iPlanet Web Server instances
iwsInstanceIndex	Server instance index
iwsInstanceId	Server instance identifier
iwsInstanceVersion	Server instance software version

Measurement	Description
iwsInstanceDescription	Description of server instance
iwsInstanceOrganization	Organization responsible for server instance
iwsInstanceContact	Contact information for persons responsible for server instance
iwsInstanceLocation	Location of server instance
iwsInstanceStatus	Server instance status
iwsInstanceUptime	Server instance uptime
iwsInstanceDeathCount	Number of times server instance processes have died
iwsInstanceRequests	Number of requests processed
iwsInstanceInOctets	Number of octets received
iwsInstanceOutOctets	Number of octets transmitted
iwsInstanceCount2xx	Number of 200-level (Successful) responses issued
iwsInstanceCount3xx	Number of 300-level (Redirection) responses issued
iwsInstanceCount4xx	Number of 400-level (Client Error) responses issued
iwsInstanceCount5xx	Number of 500-level (Server Error) responses issued
iwsInstanceCountOther	Number of other (neither 2xx, 3xx, 4xx, nor 5xx) responses issued
iwsInstanceCount200	Number of 200 (OK) responses issued
iwsInstanceCount302	Number of 302 (Moved Temporarily) responses issued
iwsInstanceCount304	Number of 304 (Not Modified) responses issued
iwsInstanceCount400	Number of 400 (Bad Request) responses issued
iwsInstanceCount401	Number of 401 (Unauthorized) responses issued

Measurement	Description
iwsInstanceCount403	Number of 403 (Forbidden) responses issued
iwsInstanceCount404	Number of 404 (Not Found) responses issued
iwsInstanceCount503	Number of 503 (Unavailable) responses issued
iwsInstanceLoad 1MinuteAverage	System load average for 1 minute
iwsInstanceLoad 5MinuteAverage	System load average for 5 minutes
iwsInstanceLoad 15MinuteAverage	System load average for 15 minutes
iwsInstanceNetwork InOctets	Number of octets transmitted on the network per second
iwsInstanceNetwork OutOctets	Number of octets received on the network per second
iwsVsTable	iPlanet Web Server virtual servers
iwsVsEntry	iPlanet Web Server virtual server
iwsVsIndex	Virtual server index
iwsVsId	Virtual server identifier
iwsVsRequests	Number of requests processed
iwsVsInOctets	Number of octets received
iwsVsOutOctets	Number of octets transmitted
iwsVsCount2xx	Number of 200-level (Successful) responses issued
iwsVsCount3xx	Number of 300-level (Redirection) responses issued
iwsVsCount4xx	Number of 400-level (Client Error) responses issued
iwsVsCount5xx	Number of 500-level (Server Error) responses issued

Measurement	Description
iwsVsCountOther	Number of other (neither 2xx, 3xx, 4xx, nor 5xx) responses issued
iwsVsCount200	Number of 200 (OK) responses issued
iwsVsCount302	Number of 302 (Moved Temporarily) responses issued
iwsVsCount304	Number of 304 (Not Modified) responses issued
iwsVsCount400	Number of 400 (Bad Request) responses issued
iwsVsCount401	Number of 401 (Unauthorized) responses issued
iwsVsCount403	Number of 403 (Forbidden) responses issued
iwsVsCount404	Number of 404 (Not Found) responses issued
iwsVsCount503	Number of 503 (Unavailable) responses issued
iwsProcessTable	iPlanet Web Server processes
iwsProcessEntry	iPlanet Web Server process
iwsProcessIndex	Process index
iwsProcessId	Operating system process identifier
iwsProcessThreadCount	Number of request processing threads
iwsProcessThreadIdle	Number of request processing threads currently idle
iwsProcessConnection QueueCount	Number of connections currently in connection queue
iwsProcessConnection QueuePeak	Largest number of connections that have been queued simultaneously
iwsProcessConnection QueueMax	Maximum number of connections allowed in connection queue
iwsProcessConnection QueueTotal	Number of connections that have been accepted
iwsProcessConnection QueueOverflows	Number of connections rejected due to connection queue overflow

Measurement	Description
iwsProcessKeepalive Count	Number of connections currently in keepalive queue
iwsProcessKeepaliveMax	Maximum number of connections allowed in keepalive queue
iwsProcessSizeVirtual	Process size in kbytes
iwsProcessSizeResident	Process resident size in kbytes
iwsProcessFraction SystemMemoryUsage	Fraction of process memory in system memory
iwsListenTable	iPlanet Web Server listen sockets
iwsListenEntry	iPlanet Web Server listen socket
iwsListenIndex	Listen socket index
iwsListenId	Listen socket identifier
iwsListenAddress	Address socket is listening on
iwsListenPort	Port socket is listening on
iwsListenSecurity	Encryption support
iwsThreadPoolTable	iPlanet Web Server thread pools
iwsThreadPoolEntry	iPlanet Web Server thread pool
iwsThreadPoolIndex	Thread pool index
iwsThreadPoolId	Thread pool identifier
iwsThreadPoolCount	Number of requests queued
iwsThreadPoolPeak	Largest number of requests that have been queued simultaneously
iwsThreadPoolMax	Maximum number of requests allowed in queue
iwsCpuTable	iPlanet Web Server CPUs
iwsCpuEntry	iPlanet Web Server CPU
iwsCpuIndex	CPU index
iwsCpuId	CPU identifier

Measurement	Description
iwsCpuIdleTime	CPU Idle Time
iwsCpuUserTime	CPU User Time
iwsCpuKernelTime	CPU Kernel Time

Part VIII

Web Application Server Resource Monitoring

20

Introducing Web Application Server Resource Monitoring

You can monitor a Web application server during a load test run and isolate application server performance bottlenecks using Performance Center's Web Application Server Resource monitors.

Web Application Server Resource monitors provide you with information about the resource usage of the following servers during a load test run:

- Ariba
- ATG Dynamo
- BroadVision
- ColdFusion
- iPlanet (NAS)
- Microsoft ASP
- Oracle9iAS HTTP
- SilverStream
- WebLogic (JMX)
- WebSphere

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 load test.

21

Ariba Monitoring

The Ariba monitor displays statistics about the resource usage on the Ariba server during the load test run.

This chapter includes:

➤ Configuring the Ariba Monitor on page 161

➤ Ariba Performance Counters on page 164

Configuring the Ariba Monitor

You select the measurements you want to monitor on the Ariba server using the Ariba Choose Measurements page.

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 Open the Add Monitor window.

➤ **Before running a load test:** In the Load Test configuration page's **Monitors** tab, select the type of monitor profile to create:

➤ To create a monitor profile for the current load test, click **Add Local Profile**.

➤ To create a monitor profile that can be used for other load tests, click **Add Profile**, type a name and description of the profile and click **OK**.

For details, see the section about configuring monitors in the *HP Performance Center User Guide*.

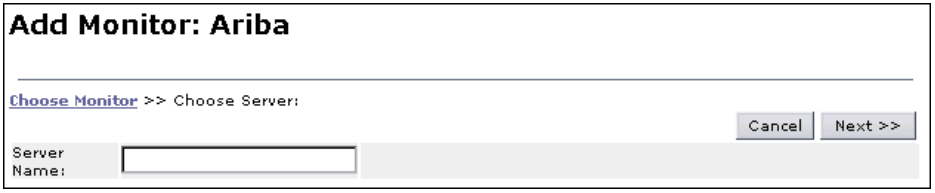
➤ **During a load test run:** On the Load Test run page:

➤ Click the **Design** button. The Load Test Design dialog box opens.

➤ Click **Monitors**.

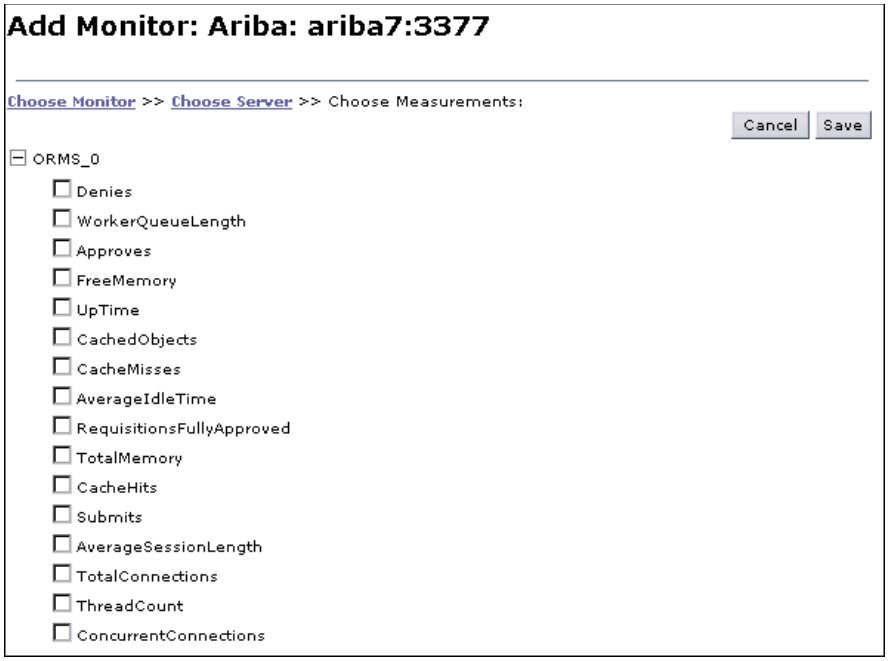
For details, see the section about creating monitor profiles while running a load test in the *HP Performance Center User Guide*.

2 Under **Web Application Server Resource Graphs**, select **Ariba** and click **Next**. The Choose Server page opens.



3 Type the server name or IP address in the format: <server name>:<port number>. For example, merc1:12130.

4 Click **Next**. The Choose Measurements page opens, displaying the available measurements and server properties.



5 Select the measurements you want to monitor.

For a list of the available performance counters, see "Ariba Performance Counters" on page 164.

6 Click **Save** to add the selected resource measurements to the monitor profile.

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 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.

Ariba Performance Counters

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

Measurement	Description
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 bytes) 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
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 Approvals submitted since previous sampling time
Denies	The cumulative count of the number of submitted Approvals denied since previous sampling time

Measurement	Description
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

22

ATG Dynamo Monitoring

The ATG Dynamo monitor uses SNMP to retrieve ATG Dynamo server statistics.

This chapter includes:

➤ Configuring the ATG Dynamo Monitor on page 167

➤ ATG Dynamo Performance Counters on page 170

Configuring the ATG Dynamo Monitor

You select the measurements you want to monitor on the ATG Dynamo server using the ATG Dynamo Choose Measurements page.

To configure the ATG Dynamo server monitor:

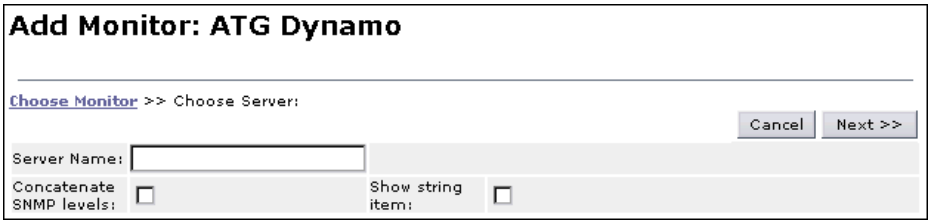
1 Open the Add Monitor window.

- **Before running a load test:** In the Load Test configuration page's **Monitors** tab, select the type of monitor profile to create:
  - To create a monitor profile for the current load test, click **Add Local Profile**.
  - To create a monitor profile that can be used for other load tests, click **Add Profile**, type a name and description of the profile and click **OK**.

  For details, see the section about configuring monitors in the *HP Performance Center User Guide*.
- **During a load test run:** On the Load Test run page:
  - Click the **Design** button. The Load Test Design dialog box opens.
  - Click **Monitors**.

For details, see the section about creating monitor profiles while running a load test in the *HP Performance Center User Guide*.

2 Under **Web Application Server Resource Graphs**, select **ATG Dynamo** and click **Next**. The Choose Server page opens.



3 Type the server name or IP address of the machine you want to monitor. Note that 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. Specify a machine name and port number using the format:

```
<server name:port number>
```

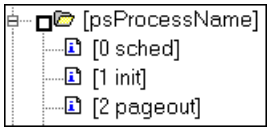
For example,

```
merc1:12130
```

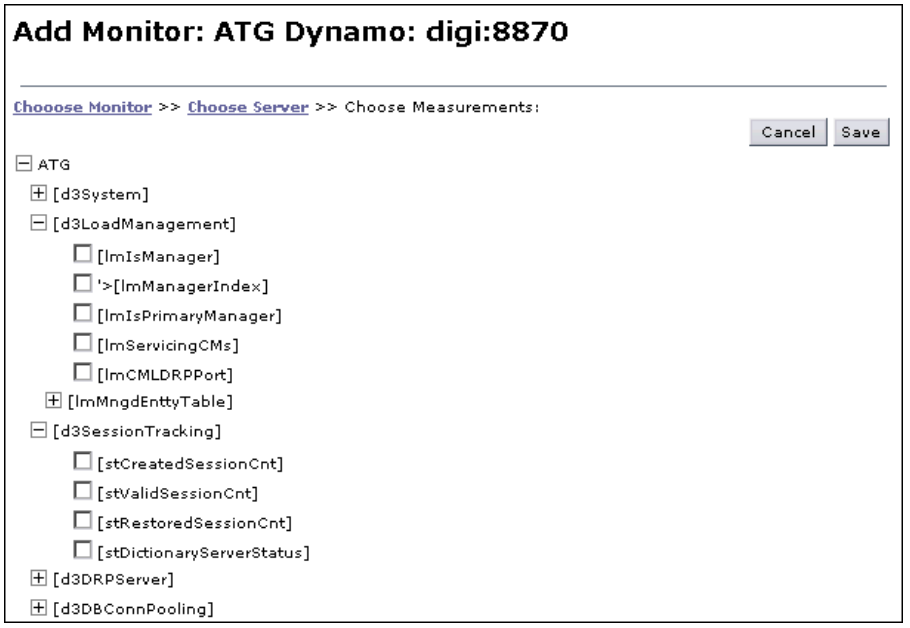
For object nodes with ten or more levels, click **Concatenate SNMP levels** to display the sub-levels as a single string, separated by periods (.). In the default view, the Choose Measurements page displays all nodes of the SNMP objects in a tree-level hierarchy.

To improve the level of measurement information for the ATG Dynamo monitor, click **Show String Item**. This lists measurements with string values (in addition to measurements with numeric values), and displays the string value as an identifying part of the measurement name.

In the following example, the string value of ProcessName (sched) is displayed in addition to its instance ID (0):



4 Click **Next**. The Choose Measurements page opens, displaying the available measurements and server properties.



5 Select the measurements you want to monitor.

For a list of the available performance counters, see "ATG Dynamo Performance Counters" on page 170.

Note: The ATG Dynamo monitor can only monitor up to 25 measurements.

---

6 Click **Save** to add the selected resource measurements to the monitor profile.

ATG Dynamo Performance Counters

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
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

Measurement	Description
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
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
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

23

BroadVision Monitoring

The BroadVision monitor displays performance statistics for all the servers/services available on the BroadVision application during a load test run.

This chapter includes:

- Configuring the BroadVision Monitor on page 174
- BroadVision Servers and Services on page 177
- BroadVision Performance Counters on page 179

Configuring the BroadVision Monitor

To monitor a BroadVision server, you must grant the client permission to invoke or launch services on the server. You select the measurements you want to monitor on the BroadVision server using the BroadVision Choose Measurements page.

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, type the following command at the BroadVision server command prompt:

```
# chmodit <server> i+all
```

To launch permission for all, type 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 Performance Center Host 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 Open the Add Monitor window.

➤ **Before running a load test:** In the Load Test configuration page's **Monitors** tab, select the type of monitor profile to create:

➤ To create a monitor profile for the current load test, click **Add Local Profile**.

➤ To create a monitor profile that can be used for other load tests, click **Add Profile**, type a name and description of the profile and click **OK**.

For details, see the section about configuring monitors in the *HP Performance Center User Guide*.

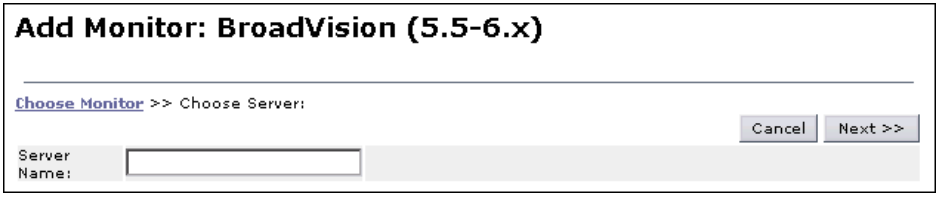
➤ **During a load test run:** On the Load Test run page:

➤ Click the **Design** button. The Load Test Design dialog box opens.

➤ Click **Monitors**.

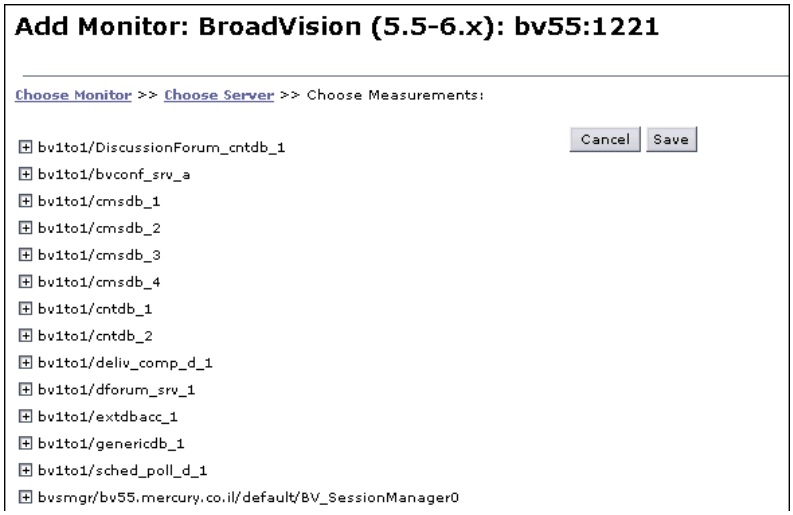
For details, see the section about creating monitor profiles while running a load test in the *HP Performance Center User Guide*.

2 Under **Web Application Server Resource Graphs**, select either **BroadVision(4.5-5.4)** or **BroadVision (5.5-6.X)** and click **Next**. The relevant Choose Server page opens.



3 Type the BroadVision server name or IP address with the port number according to the format:
<server name>:<port number>. For example, dnsqa:1221.

4 Click **Next**. The Choose Measurements page opens, displaying the available measurements and server properties.



5 Select the measurements you want to monitor.

For a list of the available performance counters, see "BroadVision Servers and Services" on page 177.

6 Click **Save** to add the selected resource measurements to the monitor profile.

BroadVision Servers and Services

The following table describes the servers and 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.

Server	Multiple Instances	Description
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 email 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.
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, then moving those records into an archive table.

Server	Multiple Instances	Description
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, 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.

BroadVision Performance Counters

Performance counters for each server or 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 update executions

BV_SRV_CTRL

SHUTDOWN	Shut-down trigger. Setting this to non-zero value causes the server to shut down.

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).

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	Count of bind() CORBA requests served
List	Count of list() CORBA requests served
New	Count of new NamingContext objects created
Rebnd	Count of rebind() CORBA requests served
Rsolv	Count of resolve() CORBA requests served
Unbnd	Count of unbind() CORBA requests served

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	Contains information about advertisements (current cache size, configured cache size, count of hits, count of misses, count of replacements).
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	Contains information about the category (current cache size, configured cache size, count of hits, count of misses, count of replacements).
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	Contains information about the external product (current cache size, configured cache size, count of hits, count of misses, count of replacements).
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	Contains information about securities (current cache size, configured cache size, count of hits, count of misses, count of replacements)
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	Script cache enabled. Initially 1 (on).
DUMP	Reference dump sequence. Increment this to cause all contexts to dump their references when they next run their next script. The dump information is appended to the jsctxdump.<pid> file in the BVLOG directory.
FLUSH	Cache flushing sequence. Increment this to cause all contexts to flush their script cache when they next run their next script.
METER	Metering enabled. Initially 0 (off).
TRACE	Tracing level. Range is BVLOG_LEVEL_MIN to BVLOG_LEVEL_MAX. Lower level means more tracing. Initial value is BVLOG_DEBUG.

JS_SCRIPT_STAT

ALLOC	Count of reference allocations
ERROR	Count of recoverable errors—how many times the Error object was set
FAIL	Count of failed script runs
JSPPERR	Count of preprocessor failures—also counted in FAIL
RELEASE	Count of reference releases
STOP	Count of stopped script runs
SUCC	Count of successful script runs
SYNTAX	Count of JavaScript compilation failures—also counted in FAIL

24

ColdFusion Monitoring

The ColdFusion monitor displays statistics about the resource usage on the ColdFusion server during the load test run.

This chapter includes:

- Configuring the ColdFusion Monitor on page 188
- ColdFusion Performance Counters on page 190

Configuring the ColdFusion Monitor

You select the measurements you want to monitor on the ColdFusion server using the ColdFusion Choose Measurements page.

Note: The ColdFusion monitor works through HTTP and supports UNIX platforms. 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 **<Performance Center 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 Open the Add Monitor window.

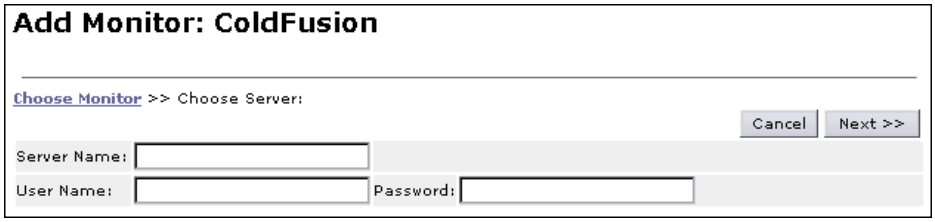
- **Before running a load test:** In the Load Test configuration page's **Monitors** tab, select the type of monitor profile to create:
  - To create a monitor profile for the current load test, click **Add Local Profile**.
  - To create a monitor profile that can be used for other load tests, click **Add Profile**, type a name and description of the profile and click **OK**.

  For details, see the section about configuring monitors in the *HP Performance Center User Guide*.
- **During a load test run:** On the Load Test run page:
  - Click the **Design** button. The Load Test Design dialog box opens.

➤ Click **Monitors**.

For details, see the section about creating monitor profiles while running a load test in the *HP Performance Center User Guide*.

2 Under **Web Application Server Resource Graphs**, select **ColdFusion** and click **Next**. The Choose Server page opens.



3 Type the name or IP address of the server whose resources you want to monitor.

Type the user login name and password.

4 Click **Next**. The Choose Measurements page opens, displaying the available measurements and server properties.

5 Select the measurements you want to monitor.

For a list of the available performance counters, see "ColdFusion Performance Counters" on page 190.

6 Click **Save** to add the selected resource measurements to the monitor profile.

ColdFusion Performance Counters

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.

25

iPlanet (NAS) Monitoring

The iPlanet (NAS) monitor uses the Simple Network Management Protocol (SNMP) to retrieve iPlanet (NAS) server statistics.

This chapter includes:

- Setting Up the Monitoring Environment on page 192
- Configuring the iPlanet (NAS) Monitor on page 196
- iPlanet (NAS) Performance Counters on page 198

Setting Up the Monitoring Environment

Before configuring the iPlanet (NAS) monitor, you need to set up the application server for SNMP monitoring.

This section includes:

- "SNMP Summary" on page 192
- "Setting up SNMP Monitoring of the iPlanet Application Server" on page 193

Note:

- This section offers a short explanation about setting up SNMP monitoring of the iPlanet Application Server. It is intended to supplement the iPlanet documentation. For more information about SNMP reporting architecture and theory, see the iPlanet documentation.
- The instructions below assume that SNMP statistics will be collected on the standard SNMP port 161.\

SNMP Summary

- Solaris has a native SNMP agent, **snmpdx**, that is started automatically at boot time by the script **/etc/rc3.d/S76snmpdx**. This daemon communicates on the standard SNMP port 161. The port number can be changed with the **-p <port>** option.
- Planet products are shipped with their own SNMP agents. The architecture is such that there is one **master agent** per host, which a network management station communicates with, and one or more **subagents** that collect data from various iPlanet products and forward statistics to the master agent. The master agent also defaults to communicating on port 161.
- To run both the Solaris SNMP agent and the iPlanet SNMP agent, a proxy must be used that makes the Sun agent look like a subagent to the iPlanet master agent.

Setting up SNMP Monitoring of the iPlanet Application Server

Use your SNMP management tool to query the SNMP master agent on port 161. You should see all the information provided by the Solaris SNMP agent as well as any iPlanet subagents that you have configured.

The next time that you boot Solaris, the Sun and iPlanet SNMP agents will be started automatically by the boot scripts that you have configured.

To set up the application server for SNMP monitoring:

1 Log in the the system.

Login to the system as root. (Only a root user can change the port number and run the agents).

2 Change the port number for the Solaris SNMP agent:

a Stop the SNMP agent by running /etc/rc2.d/K76snmpdx stop.

b Edit /etc/rc3.d/S76snmpdx to run the Solaris daemon on a non-standard port number, for example, 1161:

Replace

```
/usr/lib/snmp/snmpdx -y -c /etc/snmp/conf
```

with

```
/usr/lib/snmp/snmpdx -p 1161 -y -c /etc/snmp/conf
```

3 Configure and run the iPlanet agents, magt and sagt:

The master and proxy agents and startup scripts are found in **<ias install directory>\snmp**.

a In the script S75snmpagt, add a line to the environment variable GX_ROOTDIR so that it points to your iAS installation. For example, if the iPlanet Application Server is installed in /usr/iplanet/ias6/ias:

```
GX_ROOTDIR=/usr/iplanet/ias6/ias
exprt GX_ROOTDIR
```

b Copy the script S75snmpagt to /etc/rc3.d

c chmod 755 /etc/rc3.d/S75snmpagt

d In /etc/rc3.d/S75snmpagt /etc/rc2.d/K07snmpagt

e You can configure system information and traps.

In the example below, information has been added about the system owner and location, and SNMP traps have been sent to a network manager station ("mde.uk.sun.com").

```
COMMUNITY       public
ALLOW ALL OPERATIONS
INITIAL sysLocation "Under Joe Bloggs' Desk in Headquarters"
INITIAL sysContact "Joe Bloggs
Email: Joe.Bloggs@Sun.COM
Voice: +1 650 555 1212"
MANAGER mde.uk.sun.com
SEND ALL TRAPS TO PORT 162
WITH COMMUNITY public
```

Note: There is no need to edit the proxy agent's configuration file (CONFIG_SAGT).

---

f Start the iPlanet agents by running the command:

```
/etc/rc3.d/S75snmpagt start
```

4 Start the Solaris SNMP agent:

Restart the Solaris SNMP agent by running the command:

```
/etc/rc3.d/S76snmpdx start
```

5 Configure iPlanet Application Server for SNMP statistics:

a Start the iPlanet Application Server admin tool ksvradmin.

b In the General View, select the instance name that you want to manage.

c Click the **SNMP** tab in the management frame.

d Select **Enable SNMP Administration and Monitoring** and **Enable SNMP Debug**.

e Type 60 in the **Connection Attempt Interval** field, and exit ksvradmin.

f Restart the iPlanet Application Server with the commands:

```
iascontrol stop
iascontrol kill
iascontrol start
```

g Check in the logfile **<iASInstallDir>/logs/ias.log** that the application server successfully connected to the master agent. You should see the following line:

```
kas> SNMP: Connected to master agen
```

6 Start SNMP subagents for iPlanet Web Server:

a Use your Web browser to access the iPlanet Web Server.

b Select the Web server you wish to administer, and click the **Manage** button.

c Select the **Monitor** tab, and click **SNMP Subagent Configuration** on the left side of the page.

d Type in the configuration information and set the radio button **Enable SNMP Statistics Collection** to "On".

e Click **SNMP Subagent Control**.

f Click the **Start** button.

7 Start SNMP subagents for iPlanet Directory Server:

a Use the Netscape Administration Console to manage the iPlanet Directory Server.

b Select the **Configuration** tab.

c Click the **SNMP** tab in the Configuration frame.

d Select the **Enable statistics collection** check box.

e Set "Master Host" to "localhost".

f Set "Master port" to 199.

g In the other fields, type the appropriate information.

h Click the **Start Subagent** button.

Configuring the iPlanet (NAS) Monitor

After you have configured the iPlanet SNMP Service, you must select the counters that you want the iPlanet (NAS) monitor to measure. You select these measurements using the iPlanet (NAS) Resources Choose Measurements page.

To configure the iPlanet (NAS) Resources monitor:

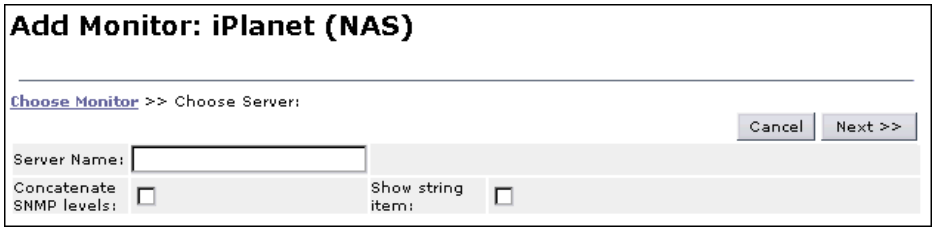
1 Open the Add Monitor window.

- **Before running a load test:** In the Load Test configuration page's **Monitors** tab, select the type of monitor profile to create:
  - To create a monitor profile for the current load test, click **Add Local Profile**.
  - To create a monitor profile that can be used for other load tests, click **Add Profile**, type a name and description of the profile and click **OK**.

  For details, see the section about configuring monitors in the *HP Performance Center User Guide*.
- **During a load test run:** On the Load Test run page:
  - Click the **Design** button. The Load Test Design dialog box opens.
  - Click **Monitors**.

  For details, see the section about creating monitor profiles while running a load test in the *HP Performance Center User Guide*.

2 Under **Web Application Server Resource Graphs**, select **iPlanet (NAS)** and click **Next**. The Choose Server page opens.



3 Type the server name or IP address of the machine you want to monitor. If the iPlanet SNMP agent is running on a different port than the default SNMP port, you need to define the port number. Specify the port number and the server name using the format:

```
<server name:port number>
```

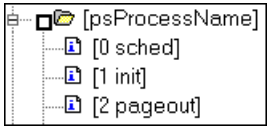
For example,

```
digi:8888
```

For object nodes with ten or more levels, click **Concatenate SNMP levels** to display the sub-levels as a single string, separated by periods (.). In the default view, the Choose Measurements page displays all nodes of the SNMP objects in a tree-level hierarchy.

To improve the level of measurement information for the iPlanet (NAS) monitor, click **Show String Item**. This lists measurements with string values (in addition to measurements with numeric values), and displays the string value as an identifying part of the measurement name.

In the following example, the string value of ProcessName (sched) is displayed in addition to its instance ID (0):



4 Click **Next**. The Choose Measurements page opens, displaying the available measurements and server properties.

5 Select the measurements you want to monitor.

For a list of the available performance counters, see "iPlanet (NAS) Performance Counters" on page 198.

Note: The iPlanet (NAS) monitor can only monitor up to 25 measurements.

---

6 Click **Save** to add the selected resource measurements to the monitor profile.

iPlanet (NAS) Performance Counters

The following tables describe the counters that can be monitored:

Netscape Performance Counters

Measurement	Description
nasKesEngConnRetries	The maximum number of times the administration server will try to connect to an engine.
nasKesEngMaxRestart	The maximum number of times the administration server will restart an engine after a failure.
nasKesEngAutoStart	Start all the engines at startup of the administration server.
nasKesConfigHeartBeat	Heart Beat.

KES Performance Counters

Measurement	Description
nasKesId	The ID of the KES this engine belongs to.
nasKesMinThread	The default minimum number of threads per engine.
nasKesMaxThread	The default maximum number of threads per engine.
nasKesLoadBalancerDisable	Enable or Disable the load balancer service.
nasKesCpuLoad	The total CPU usage on this host.
nasKesDiskLoad	The total disk usage on this host.
nasKesMemLoad	The total memory usage on this host.
nasKesRequestLoad	The number of requests on this NAS.
nasKesCpuLoad Factor	The relative importance of CPU usage in computing the server load. This number is specified as a percentage. The sum of all server load factors, CPULoad, DiskLoad, MemLoad and ExecReqs must equal 100%.
nasKesDiskLoadFactor	The relative importance of Disk usage in computing the server load. This number is specified as a percentage. The sum of all server load factors, CPULoad, DiskLoad, MemLoad and ExecReqs must equal 100%.
nasKesMemLoadFactor	The relative importance of Memory usage in computing the server load. This number is specified as a percentage. The sum of all server load factors, CPULoad, DiskLoad, MemLoad and ExecReqs must equal 100%.

Measurement	Description
nasKesAppLogicsRunningFactor	The relative importance of the number of times an AppLogic is run in computing the AppLogic execution performance. This figure is specified as a percentage. The sum of all agent load factors, ResultCached, AvgExecTime, LastExecTime, and ServerLoad must equal 100%.
nasKesResultsCachedFactor	The relative importance of the cached results of an AppLogic in computing the AppLogic execution performance. This figure is specified as a percentage. The sum of all agent load factors, ResultCached, AvgExecTime, LastExecTime, and ServerLoad must equal 100%
nasKesAvgExecTimeFactor	The relative importance of the average execution time of an AppLogic in computing the AppLogic execution performance. This figure is specified as a percentage. The sum of all agent load factors, ResultCached, AvgExecTime, LastExecTime, and ServerLoad must equal 100%.
nasKesLastExecTimeFactor	The relative importance of the last execution time of an AppLogic in computing the AppLogic execution performance. This figure is specified as a percentage. The sum of all agent load factors, ResultCached, AvgExecTime, LastExecTime, and ServerLoad must equal 100%.
nasKesHitsFactor	The relative importance of the number of AppLogics running in computing the AppLogic execution performance. This figure is specified as a percentage. The sum of all agent load factors, ResultCached, AvgExecTime, LastExecTime, and ServerLoad must equal 100%.

Measurement	Description
nasKesServerLoadFactor	The relative importance of the server load (computed using the four server load factors) in computing AppLogic execution performance. The sum of all agent load factors, ResultCached, AvgExecTime, LastExecTime, and ServerLoad must equal 100%.
nasKesBroadcastInterval	The length of time in seconds, between each broadcast attempt from the load balancer daemon.
nasKesApplogicBroadcastInterval	The length of time in seconds, between each broadcast of AppLogics load information across all the server in the cluster. This should be greater than nasKesBroacastInterval.
nasKesServerBroadcastInterval	The length of time in seconds, between each broadcast of server load information across all the server in the cluster. This should be greater than nasKesBroacastInterval.
nasKesServerLoadUpdateInterval	The length of time in seconds between each update of server load informations. A server load update applies the server load data that has been sampled up until the moment when the update occurs.
nasKesCpuLoadUpdateInterval	The length of time, in seconds, between each sampling of CPU usage.
nasKesDiskLoadUpdateInterval	The length of time, in seconds, between each sampling of disk usage.
nasKesMemLoadUpdateInterval	The length of time, in seconds, between each sampling of memory thrashes.
nasKesTotalReqsUpdateInterval	The length of time, in seconds, between each sampling of the number of requests.
nasKesMaxHops	The maximum number of times a request can be load-balanced to another server.

Measurement	Description
nasKesODBCReqMinThread	The minimum number of threads reserved to process asynchronous requests.
nasKesODBCReqMaxThread	The maximum number of threads reserved to process asynchronous requests.
nasKesODBCCacheMaxConns	The maximum number of connections opened between NAS and the database.
nasKesODBCCacheFreeSlots	The minimum number of cached connections established between NAS and the database.
nasKesODBCCacheTimeout	The time after which an idle connection is dropped.
nasKesODBCCacheInterval	The interval in seconds at which the cache cleaner will try to disconnect connections already idle for longer than the specified timeout.
nasKesODBCConnGiveupTime	Maximum time the driver will try to connect to the database.
nasKesODBCCacheDebug	Turns on the connection cache debug information.
nasKesODBCResultSetInitRows	The number of rows fetched at once from the database.
nasKesODBCResultSetMaxRows	The maximum number of rows the cached result set can contain.
nasKesODBCResultSetMaxSize	The maximum size of result set the driver will cache.
nasKesODBCSqlDebug	Turns on SQL debug information.
nasKesODBCEnableParser	Turns on SQL parsing.
nasKesORCLReqMinThread	The minimum number of threads reserved to process asynchronous requests.
nasKesORCLReqMaxThread	The maximum number of threads reserved to process asynchronous requests.

Measurement	Description
nasKesORCLCacheMaxConns	The maximum number of connections opened between NAS and the database.
nasKesORCLCacheFreeSlots	The minimum number of cached connections established between NAS and the database.
nasKesORCLCacheTimeout	The time after which an idle connection is dropped.
nasKesORCLCacheInterval	The interval in seconds at which the cache cleaner will try to disconnect connections already idle for longer than the specified timeout.
nasKesORCLConnGiveupTime	The maximum time the driver will spend trying to obtain a connection to Oracle.
nasKesORCLCacheDebug	Turns on the connection cache debug information.
nasKesORCLResultSetInitRows	The number of rows fetched at once from the database.
nasKesORCLResultSetMaxRows	The maximum number of rows the cached result set can contain.
nasKesORCLResultSetMaxSize	The maximum size of result set the driver will cache.
nasKesORCLSqlDebug	Turns on SQL debug information.
nasKesSYBReqMinThread	The minimum number of threads reserved to process asynchronous requests.
nasKesSYBReqMaxThread	The maximum number of threads reserved to process asynchronous request.
nasKesSYBCacheMaxConns	The maximum number of connections opened between NAS and the database.
nasKesSYBCacheFreeSlots	The minimum number of cached connections established between NAS and the database.

Measurement	Description
nasKesSYBCacheTimeout	The time after which an idle connection is dropped.
nasKesSYBCacheInterval	The interval time between cached connections.
nasKesSYBConnGiveupTime	The maximum time the driver will spend trying to obtain a connection to Sybase before giving up.
nasKesSYBCacheDebug	Turns on the connection cache debug information.
nasKesSYBResultSetInitRows	The number of rows fetched at once from the database.
nasKesSYBResultSetMaxRows	The maximum number of rows the cached result set can contain.
nasKesSYBResultSetMaxSize	The maximum size of result set the driver will cache.

Engine Performance Counters

Measurement	Description
nasEngKesPort	The port of the KXS this engine serves. This is supplied as part of the object ID and cannot be modified after creation.
nasEngPort	The TCP/IP port this engine is listening on. The port can only be specified at the creation of the engine. It is not allowed to modify it.
nasEngType	Type of the engine: executive(0), Java(1000), C++(3000).
nasEngId	The ID is an incremental number starting at 0. The ID cannot be modified.

Measurement	Description
nasEngName	The name of this engine. This is an informational string that contains kcs, kxs, or kjs.
nasEngNewConsole	Starts each engine in a new console window.
nasEngStatus	The status column used to add, remove, enable or disable an engine. To create an engine, one needs to set. This follows rfc1443.
nasEngMinThread	The default minimum number of threads per engine.
nasEngMaxThread	The default maximum number of threads per engine.
nasEngReqRate	The rate at which requests arrive.
nasEngTotalReq	The total number of requests processed since engine startup.
nasEngReqNow	The number of requests being processed.
nasEngReqWait	The requests waiting to be serviced.
nasEngReqReady	The requests that are ready to be serviced.
nasEngAvgReqTime	The average request processing time.
nasEngThreadNow	Number of threads in use by the request manager.
nasEngThreadWait	The number of idle threads.
nasEngWebReq Queue	The number of web requests that are queued.
nasEngFailedReq	The number of requests that failed.
nasEngTotalConn	The total number of connections opened.
nasEngTotalConnNow	The total number of connections in use.
nasEngTotalAccept	The total number of connections listening to incoming requests.

Measurement	Description
nasEngTotalAcceptNow	The total number of connections listening to incoming connections in use.
nasEngTotalSent	The total number of packets sent.
nasEngTotalSentBytes	The total number of bytes sent.
nasEngTotalRecv	The total number of packets received.
nasEngTotalRecvBytes	The total number of bytes received.
nasEngBindTotal	The number of AppLogic bound since startup.
nasEngBindTotalCached	The number of AppLogic cached since startup.
nasEngTotalThreads	Total number of threads created in this process.
nasEngCurrentThreads	Total number of threads in use in this process.
nasEngSleepingThreads	Number of threads sleeping in this process.
nasEngDAETotalQuery	Total number of queries executed since startup.
nasEngDAEQueryNow	The number of queries being processed.
nasEngDAETotalConn	The number of logical connections created since startup.
nasEngDAEConnNow	The number of logical connections in use.
nasEngDAECacheCount	The number of caches.
nasEngODBCQueryTotal	Total number of queries executed since startup.
nasEngODBCPreparedQueryTotal	Total number of odbc prepared queries executed since startup.
nasEngODBCConnTotal	Total number of connections opened since startup.
nasEngODBCConnNow	Number of connections currently opened.

Measurement	Description
nasEngORCLQueryTotal	Total number of queries executed since startup.
nasEngORCLPreparedQueryTotal	Total number of prepared queries executed since startup.
nasEngORCLConnTotal	Total number of connections established with Oracle since startup.
nasEngORCLConnNow	Number of connections opened with Oracle now.
nasEngSYBQueryTotal	Total number of queries the driver processed since startup.
nasEngSYBPreparedQueryTotal	Total number of prepared queries processed since startup.
nasEngSYBConnTotal	Total number of connections opened since startup.
nasEngSYBConnNow	Number of SYB connections opened now.
nasStatusTrapEntry	The KES definition.
nasTrapKesIpAddress	The IP Address of KES host.
nasTrapKesPort	The port of the main engine of this NAS.
nasTrapEngPort	The port of the engine generating this event.
nasTrapEngState	The port of the engine generating this event.

26

Microsoft Active Server Pages Monitoring

The Microsoft Active Server Pages (ASP) monitor displays statistics about the resource usage on the ASP server during the load test run.

This chapter includes:

➤ Configuring the Microsoft Active Server Pages Monitor on page 209

➤ MS Active Server Pages Performance Counters on page 212

Configuring the Microsoft Active Server Pages Monitor

You select the measurements you want to monitor on the Microsoft ASP application server using the MS Active Server Pages Choose Measurements page.

Note: To monitor an ASP server through a firewall, use TCP, port 139.

To configure the ASP monitor:

1 Open the Add Monitor window.

➤ **Before running a load test:** In the Load Test configuration page's **Monitors** tab, select the type of monitor profile to create:

➤ To create a monitor profile for the current load test, click **Add Local Profile**.

➤ To create a monitor profile that can be used for other load tests, click **Add Profile**, type a name and description of the profile and click **OK**.

For details, see the section about configuring monitors in the *HP Performance Center User Guide*.

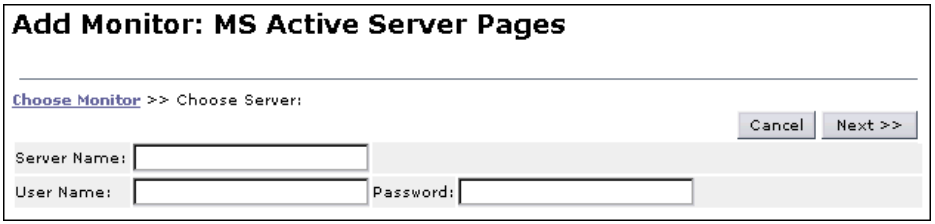
➤ **During a load test run:** On the Load Test run page:

➤ Click the **Design** button. The Load Test Design dialog box opens.

➤ Click **Monitors**.

For details, see the section about creating monitor profiles while running a load test in the *HP Performance Center User Guide*.

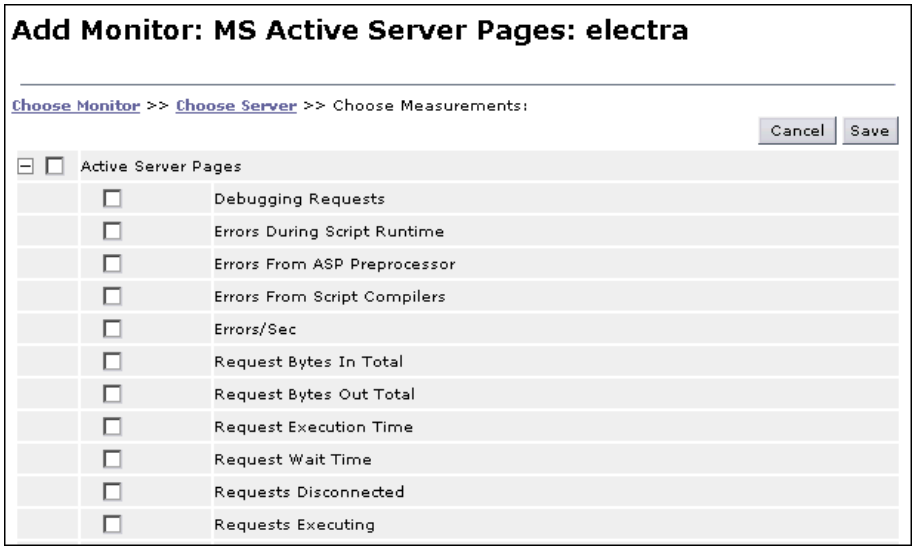
2 Under **Web Application Server Resource Graphs**, select **MS Active Server Pages** and click **Next**. The Choose Server page opens.



3 Type the name or IP address of the server whose resources you want to monitor.

Type the user login name and password.

4 Click **Next**. The Choose Measurements page opens, displaying the available measurements and server properties.



5 Select the measurements you want to monitor.

For a list of the available performance counters, see “MS Active Server Pages Performance Counters” on page 212.

6 Click **Save** to add the selected resource measurements to the monitor profile.

MS Active Server Pages Performance Counters

The following table describes some of the counters that can be measured:

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.

27

Oracle9iAS HTTP Server Monitoring

The Oracle9iAS HTTP Server monitor displays statistics about the resource usage on the Oracle9iAS HTTP server during the load test run.

This chapter includes:

- Configuring the Oracle9iAS HTTP Server Monitor on page 213
- Oracle9iAS HTTP Server Performance Counters on page 217

Configuring the Oracle9iAS HTTP Server Monitor

The Oracle9iAS HTTP Server (SiteScope) monitor allows you to monitor multiple parameters or counters with a single monitor instance. This allows you to watch server loading for performance, availability, and capacity planning.

Before Running the Load Test

You need to configure the Oracle9iAS HTTP Server online monitor, and select the default measurements you want to display. You select measurements to monitor the Oracle9iAS HTTP server using the Oracle HTTP Server Monitor Configuration dialog box.

For Oracle9iAS HTTP Server monitor:

- You must start running the Oracle9iAS HTTP server before you begin selecting the measurements you want to monitor.
- The port you use to monitor an Oracle9iAS HTTP server through a firewall depends on the configuration of your server.

For Oracle9iAS HTTP Server (SiteScope) monitor:

- Create a separate monitor instance for each Oracle9i application server in your environment.
- You must enable Web caching on the Oracle9i application server in order to use the Oracle9iAS monitor.
- The default run schedule for this monitor is every 10 minutes, but you can change it to run more or less often using the **Update every** setting.
- Although SiteScope is installed on the Utility Server, we recommend installing it on the same machine as the Controller, or on a dedicated server in order to reduce load on the Utility Server.

To configure the Oracle9iAS HTTP monitor:

1 Open the Add Monitor window.

- **Before running a load test:** In the Load Test configuration page's **Monitors** tab, select the type of monitor profile to create:
  - To create a monitor profile for the current load test, click **Add Local Profile**.
  - To create a monitor profile that can be used for other load tests, click **Add Profile**, type a name and description of the profile and click **OK**.

  For details, see the section about configuring monitors in the *HP Performance Center User Guide*.
- **During a load test run:** On the Load Test run page:
  - Click the **Design** button. The Load Test Design dialog box opens.
  - Click **Monitors**.

  For details, see the section about creating monitor profiles while running a load test in the *HP Performance Center User Guide*.

2 Under **Web Application Server Resource Graphs**, select **Oracle9iAS HTTP Server** or **Oracle9iAS HTTP Server (SiteScope)** and click **Next**. The Choose Server page opens.

3 Type the name or IP address of the server whose resources you want to monitor. In addition:

- For Oracle9iAS HTTP Server, type the user login name and password on the Oracle server.
- For Oracle9iAS HTTP Server (SiteScope), type the SiteScope machine name and port (default: 8888), and specify if you are using a Secure HTTP connection or a SiteScope user account. If you are using a SiteScope account, fill in the account name or number, and the user name and password defined to log in to the SiteScope account.

Click **Next**.

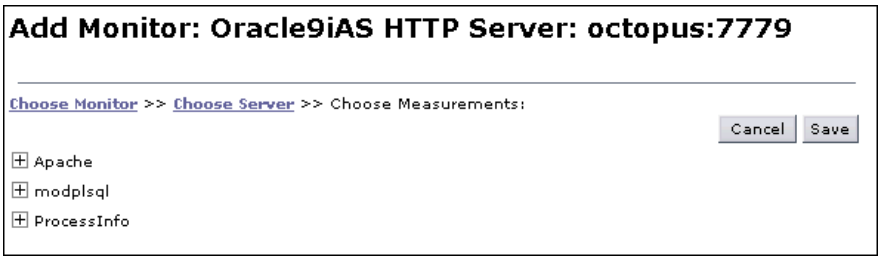
4 **(Oracle9iAS HTTP Server (SiteScope) monitor only)** The first time you add measurements using the Oracle9iAS HTTP Server (SiteScope) monitor you need to configure the monitor properties.

On the Configuring Oracle9iAS Monitor page that opens, enter the monitor properties, as follows:

- **Port**. Type the port number of the Oracle9iAS server.
- **URL**. The server administration URL appears by default, and has the following form:
  http://server:port/webcacheadmin?SCREEN_ID=CGA.Site.Stats&ACTION=Show
- **HTTP Proxy.** If you are using a proxy to access the Oracle9iAS server, type the domain name and port of an HTTP Proxy Server (for example, proxy.sitescope.com:8080).
- **Authorization User Name**. If the URL requires authentication, type the user name for the server administrator page.
- **Authorization Password**. If the URL requires authentication, type the password for the server administrator page.
- **Proxy Server User Name**. If the proxy server requires authorization, type the user name. Your proxy server must support Proxy-Authenticate for these options to function.
- **Update every**. Type how frequently (in seconds) the monitor should check the Oracle9iAS server. You must specify a time increment of at least 15 seconds. The default is 30 seconds.

Click **Next**.

5 The Choose Measurements page opens, displaying the available measurements and server properties.



6 Select the measurements you want to monitor.

For a list of the available performance counters, see “Oracle9iAS HTTP Server Performance Counters” on page 217.

7 Click **Save** to add the selected resource measurements to the monitor profile.

Oracle9iAS HTTP Server Performance Counters

The following tables describe the performance counters for the Oracle9i Server monitors.

- Oracle9iAS HTTP Server (SiteScope) Monitor Performance Counters
- Performance Center Oracle9iAS HTTP Server Monitor Performance Counters

Oracle9iAS HTTP Server (SiteScope) Monitor Performance Counters

---

Note: The Oracle9iAS HTTP Server (SiteScope) monitor can monitor a maximum of 10 counters at one time.

The following table describes the counters that can be monitored by the Oracle9iAS HTTP Server (SiteScope) monitor:

Measurement	Description
Up/Down Time (up/down)	Status of the application Web server.
Completed Requests (number/sec)	Number of requests that the application Web server processes per second.
Completed Requests(max/sec)	Maximum number of requests that the application Web server can process per second.
Completed Requests (avg/sec)	Average number of requests that the application Web server has processed per second.
Completed Requests (total)	Accumulated number of requests that the application Web server has processed.
Latency (avg this interval)	Average latency for 10 second intervals to process requests for Oracle Web Cache.
Latency (avg since start)	Average number of seconds to process requests for Oracle Web Cache since the application Web server started.

Measurement	Description
Load (now)	Current number of connections from Oracle Web Cache that the application Web server has open.
Load (max)	Maximum number of connections that the application Web server has had open at one time.
Active Sessions (now)	Current number of active connections from Oracle Web Cache to the application Web servers.
Active Sessions (max)	Maximum number of active connections that the application Web server has had open at one time.
Apology Pages Served (Network Error - number this second)	Current number of apology pages that the Oracle Web Cache is serving to Web browsers due to a network error.
Apology Pages Served (Network Error - Total)	Total number of apology pages that the Oracle Web Cache is serving to Web browsers due to a network error.
Apology Pages Served (Site Busy - number this second)	Current number of apology pages that the Oracle Web Cache is serving to Web browsers due to a busy Web site error.
Apology Pages Served (Site Busy - total)	Total number of apology pages that the Oracle Web Cache is serving to Web browsers due to a busy Web site error.
Application Web Server Backlog (now)	Current number of requests that the application Web server is processing for Oracle Web Cache.
Application Web Server Backlog (max)	Maximum number of requests that the application Web server has processed for Oracle Web Cache.

Performance Center Oracle9iAS HTTP Server Monitor Performance Counters

The following tables describe some of the modules and the counters that can be monitored by the native Performance Center Oracle9iAS HTTP Server monitor.

Modules

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

Measurement	Description
mod_ssl.c	Provides SSL support
mod_plsql.c	Handles requests for Oracle stored procedures
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

Measurement	Description
mod_jserv.c	Routes HTTP requests to JServ server processes. Balances load across multiple JServs by distributing new requests in round-robin order
mod_ose.c	Routes requests to the JVM embedded in Oracle's database server
http_core.c	Handles requests for static Web pages

Counters

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

Measurement	Description
connection.maxTime	The maximum amount of time spent servicing any HTTP connection
connection.minTime	The minimum amount of time spent servicing any HTTP connection
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)

28

SilverStream Monitoring

The SilverStream monitor displays statistics about the resource usage on the SilverStream server during the load test run.

This chapter includes:

➤ Configuring the SilverStream Monitor on page 223

➤ SilverStream Performance Counters on page 226

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 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 Open the Add Monitor window.

➤ **Before running a load test:** In the Load Test configuration page's **Monitors** tab, select the type of monitor profile to create:

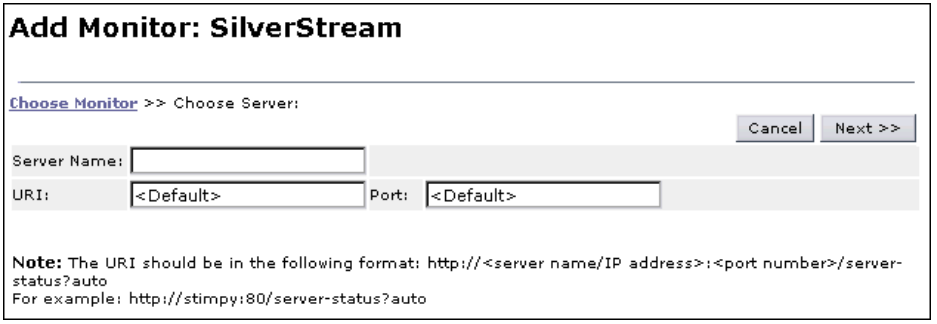
- To create a monitor profile for the current load test, click **Add Local Profile**.
- To create a monitor profile that can be used for other load tests, click **Add Profile**, type a name and description of the profile and click **OK**.

For details, see the section about configuring monitors in the *HP Performance Center User Guide*.

- **During a load test run:** On the Load Test run page:
  - Click the **Design** button. The Load Test Design dialog box opens.
  - Click **Monitors**.

  For details, see the section about creating monitor profiles while running a load test in the *HP Performance Center User Guide*.

2 Under **Web Application Server Resource Graphs**, select **SilverStream** and click **Next**. The Choose Server page opens.



3 Type the details of the server whose resources you want to monitor

a Type the name or IP address.

b In the URL box, type the URL or accept the default URL:

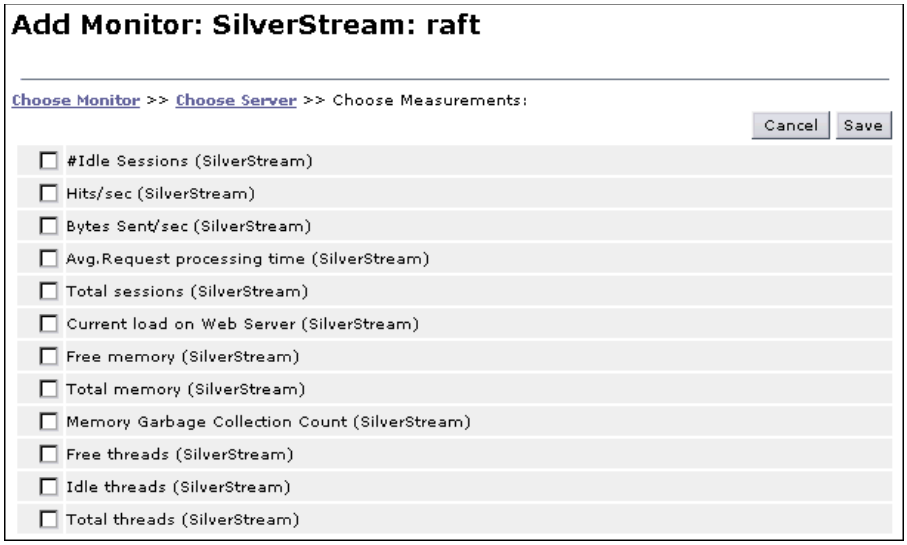
```
(/SilverStream/Statistics)
```

c In the Port box, type the Port number (without the server name), or accept the default port.

Note: The default port number and URL can vary from one server to another. Please consult the Web server administrator.

---

4 Click **Next**. The Choose Measurements page opens, displaying the available measurements and server properties.



5 Select the measurements you want to monitor.

For a list of the available performance counters, see “SilverStream Performance Counters” on page 226.

6 Click **Save** to add the selected resource measurements to the monitor profile.

Note: To monitor a SilverStream server through a firewall, use the Web server port (by default, port 80).

SilverStream Performance Counters

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.
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.

29

WebLogic (JMX) Monitoring

The BEA WebLogic (JMX) monitor uses the Java JMX interface to access runtime MBeans on the server. An MBean is a container that holds the performance data.

This chapter includes:

- Setting Up the Monitoring Environment on page 227
- Configuring the WebLogic (JMX) Monitor on page 231
- BEA WebLogic (JMX) Performance Counters on page 235

Setting Up the Monitoring Environment

Before using the WebLogic (JMX) monitor, you must:

- Install Java 1.3 or later on the Performance Center Host machine. You must install Java 1.3 or later to work with Weblogic 6.x or 7.x, and Java 1.4 to work with WebLogic 8.1.
- Set Permissions for user to be able to monitor MBeans
- Make sure that classes load directly from the file system if the servlet is disabled or not used. For more information, see "Loading Classes from the Server" on page 230.

Java 1.3.x:

1. Install Java 1.3 or later on the Performance Center Host 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.

2 Specify the path in the **<Performance Center 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
```

Java 1.4:

1 Install JDK 1.4 on the Performance Center Host machine. If Java 1.4 is already installed, but is not the default Java version being used, specify the full path to the updated version in the command line.

2 Copy **weblogic.jar** from the **<WebLogic Server installation>\..\lib** to **<Performance Center root folder>\classes**.

3 Remove **jmxri.jar** from **<Performance Center root folder>\classes**.

4 Specify the path in the **<Performance Center root folder>\dat\monitors\WebLogicMon.ini** file. Edit the JVM entry in the [WebLogicMon] section. For example:

```
JVM="E:\Program Files\JavaSoft\JRE\1.4\bin\javaw.exe
```

5 Edit the JavaVersion entry in the [WebLogicMon] section.

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 log on to the WebLogic (JMX) server:

Type the user name and password of a user with administrative privileges.

To set permissions for WebLogic version 6.x:

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 area 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 area 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, type 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**.

To set permissions for WebLogic versions 7.x and 8.x:

1 Open the WebLogic console (http://<host:port>/console).

2 In the tree on the left, select **Security** > **Realms** > **myrealm** > **Users**, and click **Configure a new User...** in the area on the right. The Create User: General tab opens.

3 In the Name box, type weblogic.admin.mbean, type a password, confirm the password, then click **Apply**.

4 In the Groups tab, type the name of any user or group you want to use for monitoring, then 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.

Note: For WebLogic version 8.1, you must load classes directly from the file system, as the WebLogic (JMX) monitor will not load the classes automatically.

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 **<Performance Center root folder>\classes**.
2. If the classes file is not located in the default **<Performance Center root folder>** folder, you need to specify the full path to it in the **<Performance Center 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 the measurements you want to monitor on the WebLogic (JMX) application server using the WebLogic JMX Choose Measurements page.

Although SiteScope is installed on the Utility Server, we recommend installing it on the same machine as the Controller, or on a dedicated server in order to reduce load on the Utility Server.

Note: The WebLogic (JMX) monitor utilizes a built-in server called the ClasspathServlet to load classes directly and automatically from the server. For information about loading classes from the server, see "Loading Classes from the Server" on page 230.

To configure the WebLogic (JMX) Monitor:

1 Open the Add Monitor window.

- **Before running a load test:** In the Load Test configuration page's **Monitors** tab, select the type of monitor profile to create:
  - To create a monitor profile for the current load test, click **Add Local Profile**.
  - To create a monitor profile that can be used for other load tests, click **Add Profile**, type a name and description of the profile and click **OK**.

  For details, see the section about configuring monitors in the *HP Performance Center User Guide*.
- **During a load test run:** On the Load Test run page:
  - Click the **Design** button. The Load Test Design dialog box opens.
  - Click **Monitors**.

  For details, see the section about creating monitor profiles while running a load test in the *HP Performance Center User Guide*.

2 Under **Web Application Server Resource Graphs**, select **WebLogic (JMX)** or **WebLogic (JMX) (SiteScope)** and click **Next**. The Choose Server page opens.

3 Type the server name or IP address of the machine you want to monitor according to the format:

```
<server name>:<port number>
```

For example,

```
mercury:8111
```

In addition:

- For WebLogic (JMX), type the user login name and password on the WebLogic JMX machine.
- For WebLogic (JMX) (SiteScope), type the SiteScope machine name and port (default: 8888), and specify if you are using a Secure HTTP connection or a SiteScope user account. If you are using a SiteScope account, fill in the account name or number, and the user name and password defined to log in to the SiteScope account.

Click **Next**.

4 **(WebLogic (JMX) (SiteScope) monitor only)** The first time you add measurements using the WebLogic (JMX) (SiteScope) monitor you need to configure the monitor properties.

In the Configuring SNMP by MIB Monitor page that opens, enter the monitor properties ,as follows:

- **Port Number**. Type the port number of WebLogic (JMX) server. The default port is 7001.
- **Password**. Type the password required to log on to the WebLogic server.
- **User Name**. Type the user name required to log on to the WebLogic server.
- **Secure Server**. Check this box if you are using a secure server connection option. If you select this option, you must type the applicable port number used by the WebLogic server for secure connections. The default secure server port is 7002.
- **Server**. Type the name of the server where the WebLogic JMX application is running.

➤ **WebLogic Jar File**. Type the absolute path name to the **weblogic.jar** file on the SiteScope machine. This file must be installed on the SiteScope server and can be downloaded from the WebLogic server. An example path is:

```
c:\bea\weblogic7\ebcc\lib\ext\weblogic.jar
```

This file is not strictly required for monitoring some earlier versions of WebLogic 6. In this case, leaving this box blank normally causes any necessary classes to be downloaded directly from the WebLogic server. This is not as efficient as loading the classes from the a ***.jar** file on the server where SiteScope is running.

Note: Do not install the **weblogic.jar** file in the SiteScope directory tree. For example, do not install it in the **<SiteScope install path>/SiteScope/java/lib/ext** directory as this will cause the Weblogic monitor to fail. You must create a separate directory for this file on the server where SiteScope is running.

➤ **WLCipher Jar File**. (This option is used only with the Secure Server (SSL) option.)

For some versions of the WebLogic Server, you need to type the absolute path to the **wlcipher.jar** file on the SiteScope machine.

For example:

```
c:bea\weblogic81\lib\wlciher.jar
```

➤ **WebLogic License File**. (This option is used only with the Secure Server (SSL) option.)

Type the absolute path to a BEA license file on the SiteScope machine. For example:

```
c:\bea\license.bea
```

- **Location of JVM**. Specify the location of a JVM where the WebLogic monitoring process should run.

Note: For monitors which do not use the Secure Server (SSL) option, this is not required. For monitors that do use SSL, the version of the JVM must be 1.4.1 or earlier.

- **Timeout**. Type the total time, in seconds, that SiteScope should wait for all WebLogic requests (including retries) to complete. The default is 180 seconds.
- **Update every**. Type how frequently the monitor should read the server statistics. The list to the right of the text box lets you select time increments of seconds, minutes, hours, or days. You must specify a time increment of at least 15 seconds. The default is 30 seconds.

Click **Next**.

5 The Choose Measurements page opens, displaying the available measurements and server properties.

6 Select the measurements you want to monitor.

For a list of the available performance counters, see "BEA WebLogic (JMX) Performance Counters" on page 235.

7 Click **Save** to add the selected resource measurements to the monitor profile.

BEA WebLogic (JMX) Performance Counters

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 HP's Load Testing Monitors Web site (http://www.mercury.com/us/products/performance-center/loadrunner/monitors/bealogic.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

30

WebSphere Monitoring

The WebSphere monitor displays statistics about the resource usage on the WebSphere 3.x, 4.x, or 5.x servers during the load test run.

This chapter includes:

- Setting Up the Monitoring Environment on page 237
- Configuring the WebSphere Monitor on page 238
- WebSphere Performance Counters on page 243

Setting Up the Monitoring Environment

The WebSphere 3.x, 4.x, and 5.x application servers have different monitor installation requirements.

To monitor WebSphere version 5.x, you need to deploy the performance servlet on the application server using the IBM WebSphere "Installing a New Application" wizard.

To deploy the performance servlet on the application server for WebSphere 5.x:

1. From the administrative console, click **Applications** > **Install New Application** in the console navigation tree.
2. For Path, specify the full path name of the source application file ("PerfServletApp.ear") on the server machine and click **Next**.
3. Select the **Generate Default Bindings** check box and click **Next**.
4. On the Install New Application page, click **Summary**, and select the **Cell/Node/Server** option. Click **Click here**.

5 On the **Map modules to application servers** panel, select the server onto which the application files will install from the **Clusters and Servers** list, and select **Module** to select all of the application modules.

6 Click **Next**, and in the Summary panel click **Finish**.

7 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 should be in the 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.

Configuring the WebSphere Monitor

You select measurements to monitor the WebLogic application server using the WebLogic Choose Measurements page.

If you are using the SiteScope monitor engine, SiteScope must be installed on a server. We recommend installing it on the same machine as the Controller, or on a dedicated server in order to reduce load on the Utility Server.

To configure the WebSphere or WebSphere 4.x-5.x monitor:

1 Open the Add Monitor window.

➤ **Before running a load test:** In the Load Test configuration page's **Monitors** tab, select the type of monitor profile to create:

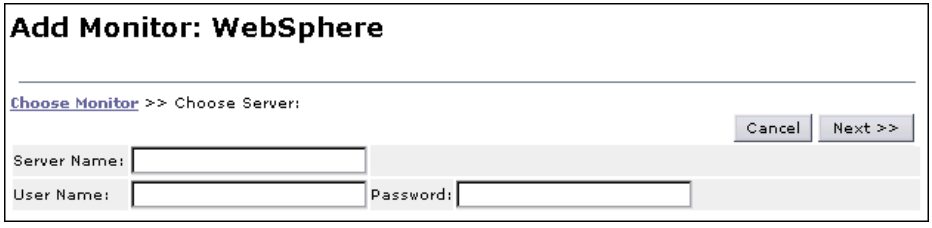
- To create a monitor profile for the current load test, click **Add Local Profile**.
- To create a monitor profile that can be used for other load tests, click **Add Profile**, type a name and description of the profile and click **OK**.

For details, see the section about configuring monitors in the *HP Performance Center User Guide*.

- **During a load test run:** On the Load Test run page:
  - Click the **Design** button. The Load Test Design dialog box opens.
  - Click **Monitors**.

  For details, see the section about creating monitor profiles while running a load test in the *HP Performance Center User Guide*.

2 Under **Web Application Server Resource Graphs**, select **WebSphere** or **WebSphere 4.x-5.x** and click **Next**. The Choose Server page opens.

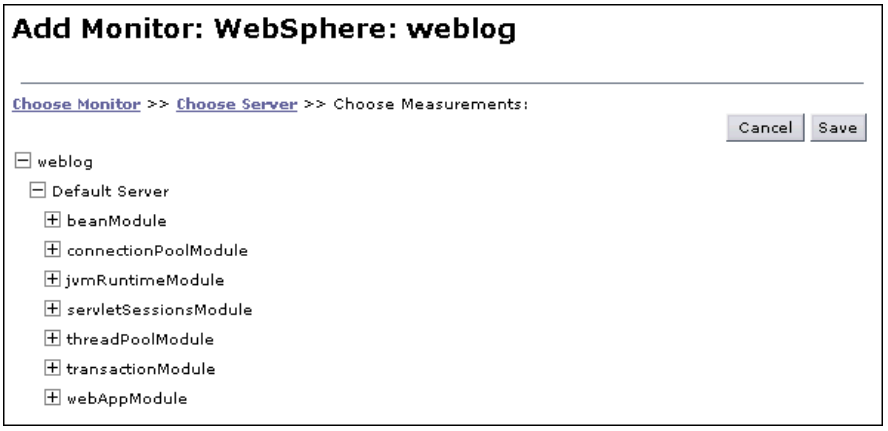


3 Type the details of the server whose resources you want to monitor.

a Type the name or IP address.

b Type the user login name and password.

4 Click **Next**. The Choose Measurements page opens, displaying the available measurements and server properties.



5 Select the measurements you want to monitor.

For a list of the available performance counters, see "WebSphere Performance Counters" on page 243.

6 Click **Save** to add the selected resource measurements to the monitor profile.

Note: The port you use to monitor a WebSphere server through a firewall depends on the configuration of your server.

Specifying a Web Alias for the Servlet Directory

You can monitor as many application servers as you want by specifying a Web alias for the servlet directory.

To specify another Web alias for the servlet directory:

By default, Performance Center 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, Performance Center will request the XML file in the 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 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 format:

```
http://<server:port_number>/<servlet_dir_alias>/servlet
```

For example,

```
http://mercury/servlet3/servlet
```

Enabling the EPM Counters on the WebSphere 3.5.x Server

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, select 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 select 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 Performance 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

Measurement	Description
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
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

Measurement	Description
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
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
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 percentage of the time that the number of threads in the pool reached or exceeded the desired maximum number

Measurement	Description
ThreadCreates	The number of threads created
ThreadDestroys	The number of threads destroyed
ConfiguredMaxSize	The configured maximum number of pooled threads

DBConnectionMgr

These are resources related to the database connection manager. The manager consists of a series of data sources, as well as a top-level aggregate of each of the performance metrics.

Measurement	Description
DataSource	Resources related to a specific data source specified by the "name" attribute
ConnectionCreates	The number of connections created
ConnectionDestroys	The number of connections released
ConnectionPoolSize	The average size of the pool, i.e., number of connections
ConnectionAllocates	The number of times a connection was allocated
ConnectionWaiters	The average number of threads waiting for a connection
ConnectionWaitTime	The average time, in seconds, of a connection grant
ConnectionTime	The average time, in seconds, that a connection is in use
ConnectionPercentUsed	The average percentage of the pool that is in use
ConnectionPercentMaxed	The percentage of the time that all connections are in use

TransactionData

These are resources that pertain to transactions.

Measurement	Description
NumTransactions	The number of transactions processed
ActiveTransactions	The average number of active transactions
TransactionRT	The average duration of each transaction
BeanObjectCount	The average number of bean object pools involved in a transaction
RolledBack	The number of transactions rolled back
Commited	The number of transactions committed
LocalTransactions	The number of transactions that were local
TransactionMethodCount	The average number of methods invoked as part of each transaction
Timeouts	The number of transactions that timed out due to inactivity timeouts
TransactionSuspended	The average number of times that a transaction was suspended

ServletEngine

These are resources that are related to servlets and JSPs.

Measurement	Description
ServletsLoaded	The number of servlets currently loaded
ServletRequests	The number of requests serviced
CurrentRequests	The number of requests currently being serviced
ServletRT	The average response time for each request
ServletsActive	The average number of servlets actively processing requests

Measurement	Description
ServletIdle	The amount of time that the server has been idle (i.e., time since last request)
ServletErrors	The number of requests that resulted in an error or an exception
ServletBeanCalls	The number of bean method invocations that were made by the servlet
ServletBeanCreates	The number of bean references that were made by the servlet
ServletDBCalls	The number of database calls made by the servlet
ServletDBConAlloc	The number of database connections allocated by the servlet
SessionLoads	The number of times the servlet session data was read from the database
SessionStores	The number of times the servlet session data was stored in the database
SessionSize	The average size, in bytes, of a session data
LoadedSince	The time that has passed since the server was loaded (UNC time)

Sessions

These are general metrics regarding the HTTP session pool.

Measurement	Description
SessionsCreated	The number of sessions created on the server
SessionsActive	The number of currently active sessions
SessionsInvalidated	The number of invalidated sessions. May not be valid when using sessions in the database mode
SessionLifetime	Contains statistical data of sessions that have been invalidated. Does not include sessions that are still alive

31

WebSphere Application Server Monitoring

The WebSphere Application Server monitor allows you to monitor the availability and server statistics of a IBM WebSphere Application Server 3.5.x, 4.x, and 5.x. The error and warning thresholds for the monitor can be set on as many as ten WebSphere Application Server performance statistics.

The WebSphere Application Server monitor monitors the server performance statistics from IBM WebSphere servers using the performance monitoring interfaces provided with WebSphere. You can monitor multiple parameters or counters with a single monitor instance. This allows you to watch server loading for performance, availability, and capacity planning. Create a separate WebSphere Application Server monitor instance for each WebSphere application server in your environment.

To obtain data for this graph, you need to configure the WebSphere Application Server monitor and select the measurements you want to display before running the load test.

This chapter includes:

- Setting up the Monitoring Environment on page 252
- Configuring the WebSphere Application Server Monitor on page 254

Setting up the Monitoring Environment

Before you can use the WebSphere Application Server monitor, you need to configure the server environment.

Although SiteScope is installed on the Utility Server, we recommend installing it on the same machine as the Controller, or on a dedicated server in order to reduce load on the Utility Server.

For WebSphere 3.5.x and 4.x

- You must first install the IBM WebSphere Administrator's Console on the SiteScope server if you are monitoring WebSphere versions 3.5.x or 4.x. When installing the Administrator's Console:
  - Select the **Custom** installation option.
  - In the Choose Application Server Components dialog box, select **Administrator's Console** and **IBM JDK 1.2.2**.
  - You will need to specify the machine you want to monitor during the installation.
- You must enable the WebSphere servers to be monitored.
  - For WebSphere 3.5.x, enable EPM Counters on the WebSphere server.
  - For WebSphere 4.x and 5.x, enable PMI Counters or enable the Performance Monitoring Service on the WebSphere server.

  You enable the counters for the application you want to monitor through the WebSphere Administrator's Console.

  For WebSphere 4.x:

  - When you have selected resources to monitor, select the **Performance** option.
  - In the dialog box that opens, expand the **Performance Modules** tree. In order to manage different levels of performance data, select the performance modules and select a performance level, and click **Set**.
- Alternatively, on WebSphere 3.5.x, you can set the EPM Specification to:

```
epm=high:epm.beanMethodData=none
```

through the WebSphere Administrator's Console.

➤ If security has been enabled on the WebSphere server, the server security ring must be copied to the admin client.

For WebSphere 5.x

Note: To monitor WebSphere version 5.x, the necessary WebSphere libraries must be available on the SiteScope server. This means that a WebSphere 5.x client must be installed on the SiteScope server.

To configure the server environment:

1. Select **Servers** > **Application Servers**, and select the server to be monitored from the Application Server list.
2. In the Configuration tab's **Additional Properties** list, click **Performance Monitoring Service**.
3. Select the **Start Up** check box.
4. In the **Initial specification level** section select **Standard** or **Custom**.
5. Click **Apply**.

To install the correct client software on a SiteScope server:

1. When installing WebSphere 5.x, select the following options from the Custom Options menu:
   - ➤ Administration (or admin console)
   - ➤ Performance Analysis

Note: Certain trial versions of IBM WebSphere do not include the Performance Analysis option required by the Sitescope WebSphere Application Server monitor. The SiteScope monitor will only work when a complete WebSphere production installation is available.

---

2. Copy all of the files from the **<WebSphere 5.x Application Server installation>\lib** folder to the **<client installation>\lib** folder (see step 1 above).

3. The WebSphere 5.x server and client settings have to match. This means that the SiteScope WebSphere Application Server Monitor will not be able to monitor a WebSphere 5.1 application server if the client libraries are from a WebSphere 5.0 application server, and vice versa.

   Client libraries should be installed in separate folders with clearly distinct directory names, such as WebSphere50 and WebSphere51, to avoid confusion and SiteScope setup errors.

4. The **sas.props** file should be replaced with **soap.props** for WebSphere 5.x installations.

Note: The WebSphere 5.x SiteScope monitor uses the WebSphere JMX interface, so the port number used to communicate with the application server is the SOAP port number. By default, the SOAP port number is **8880**.

---

5. If security has been enabled on the WebSphere server, the server security ring must be copied to the admin client.

General information

If security has been enabled on the WebSphere server, you must copy the security keyring from the WebSphere server to SiteScope. A keyring is a certification used by the server to identify the client.

Configuring the WebSphere Application Server Monitor

After you have configured the server environment, you need to configure the WebSphere Application Server monitor and select the measurements you want to display.

To configure the WebSphere Application Server monitor:

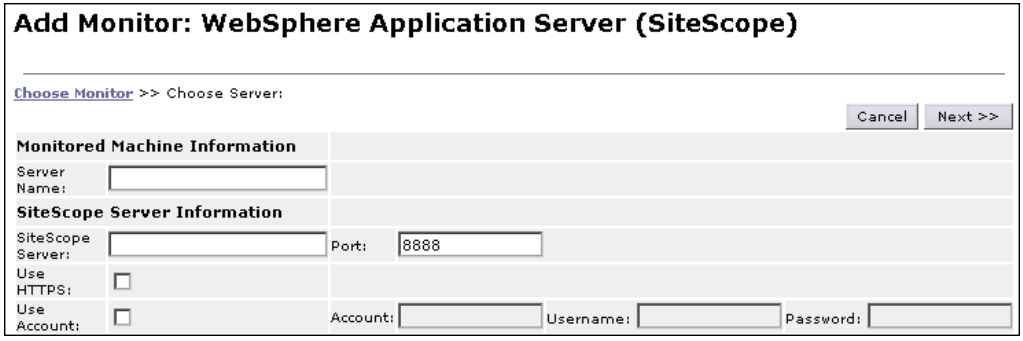
1. Open the Add Monitor window.

- **Before running a load test:** In the Load Test configuration page's **Monitors** tab, select the type of monitor profile to create:
  - To create a monitor profile for the current load test, click **Add Local Profile**.
  - To create a monitor profile that can be used for other load tests, click **Add Profile**, type a name and description of the profile and click **OK**.

  For details, see the section about configuring monitors in the *HP Performance Center User Guide*.
- **During a load test run:** On the Load Test run page:
  - Click the **Design** button. The Load Test Design dialog box opens.
  - Click **Monitors**.

  For details, see the section about creating monitor profiles while running a load test in the *HP Performance Center User Guide*.

2 Under **Web Application Server Resource Graphs**, select **WebSphere Application Server (SiteScope)** and click **Next**. The Choose Server page opens.



3 Under **Monitored Machine Information**, type the name or IP address of the server whose resources you want to monitor.

Under **SiteScope Server Information**, type the SiteScope machine name and port (default: 8888), and specify whether you are using a Secure HTTP connection or a SiteScope user account. If you are using a SiteScope account, fill in the account name or number, and the user name and password defined to log in to the SiteScope account.

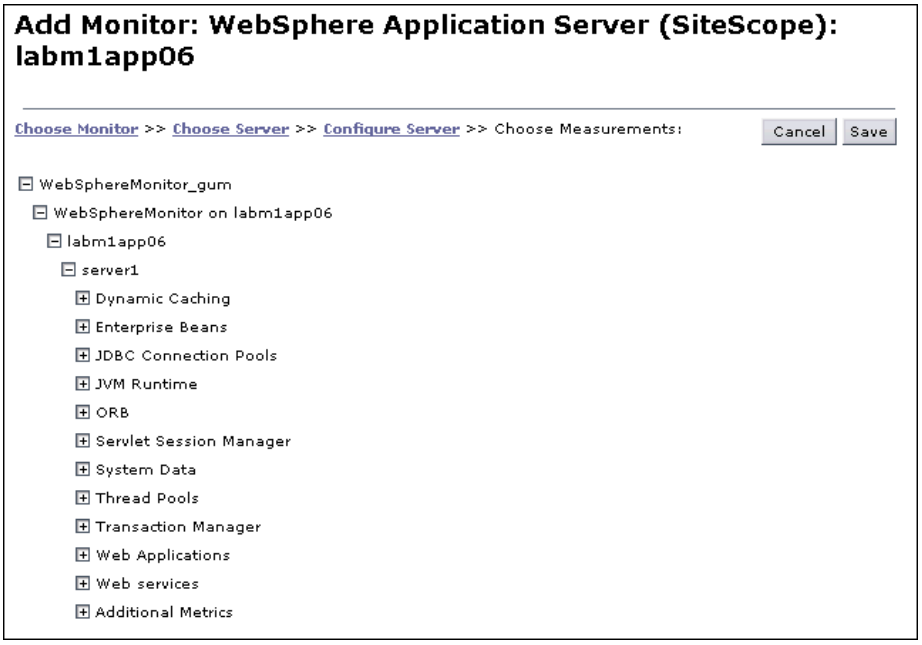
4 Click **Next**. The Configuring WebSphere Application Server monitor page opens.

Type values for the monitor properties:

- **Client Properties File**. Type the name of the custom client properties file. For WebSphere 5.x+, you should select an appropriate soap.client.props file. By default, the /properties/soap.client.props file is used.
- **Version**. Select the version of the WebSphere server.

- **WebSphere Directory**. Type the path of the WebSphere directory. This directory should contain at least an Admin Console installation.
- **Port Number**. Type the port number of the WebSphere server. This should be the SOAP port for WebSphere 5.x+. The default port number is 8880.
- **Classpath**. Type any extra classpath elements needed for the monitor program.
- **Password**. Type the password that SiteScope should use to log on to WebSphere server.
- **Security Realm**. Type the security realm for the WebSphere server (3.5x only).
- **User Name**. Type the user name that SiteScope should use to log on to WebSphere server.
- **Server**. Type the name of the server where the WebSphere application is running. Do not type back-slashes (\\) that indicate a UNC path as part of the name of the server.
- **Update every**. The number entered here indicates the amount of time, in seconds, between one monitor check and the next. By default the monitor updates every 30 seconds.

5 Click **Next**. The Choose Measurements page opens, displaying the available measurements and server properties.



6 Select the measurements you want to monitor. There is a large number of counters available for the WebSphere Application Server Monitor. The list of available counters will vary depending on which version of WebSphere you are running.

For a list of counters that may be available, see "WebSphere Performance Counters" on page 243.

7 Click **Save** to add the selected resource measurements to the monitor profile.

32

WebSphere (EPM) Monitoring

The WebSphere (EPM) monitor displays statistics about the resource usage on the WebSphere 3.5.x server during the load test run.

This chapter includes:

- Setting Up the Monitoring Environment on page 259
- Configuring the WebSphere (EPM) Monitor on page 267

Setting Up the Monitoring Environment

To monitor the IBM WebSphere application server (3.5.x), you must first install the IBM WebSphere Administrator's Console on the Performance Center Host machine and enable the EPM counters. You can then configure the WebSphere (EPM) monitor before running the load test.

This section includes:

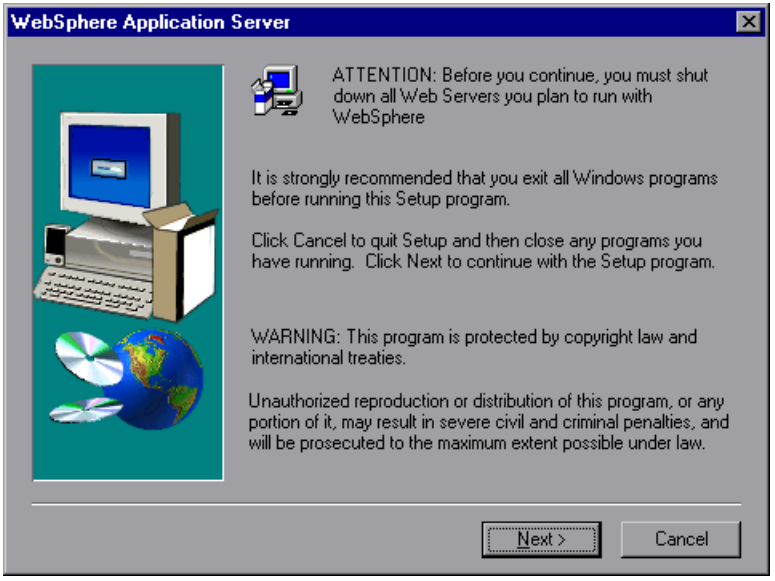
- "Installing the IBM WebSphere Administrator's Console" on page 260
- "Copying the Security Keyring" on page 265
- "Enabling EPM Counters on the WebSphere 3.5.x Server" on page 266

Installing the IBM WebSphere Administrator's Console

You must first install the IBM WebSphere Administrator's Console on the Performance Center Host machine.

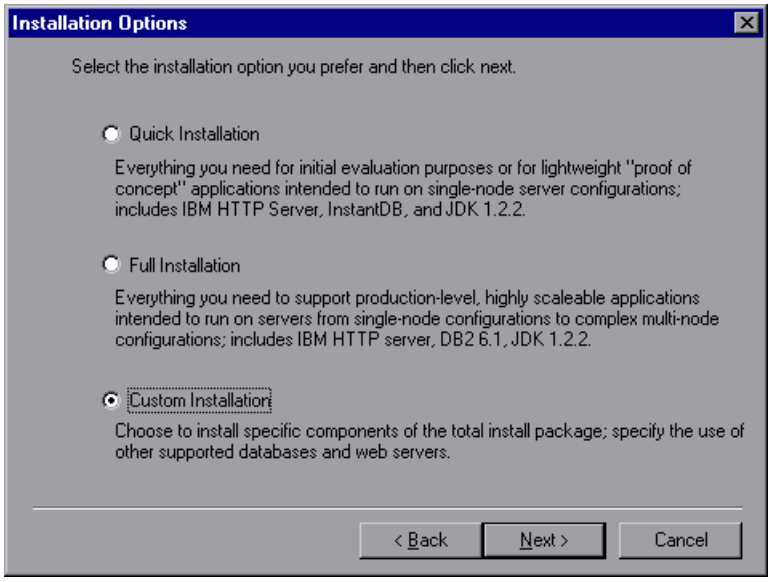
To install the IBM WebSphere Administrator's Console:

1 Start the WebSphere installation program from the WebSphere 3.5 Windows NT distribution CD-ROM. The WebSphere Application Server dialog box opens.

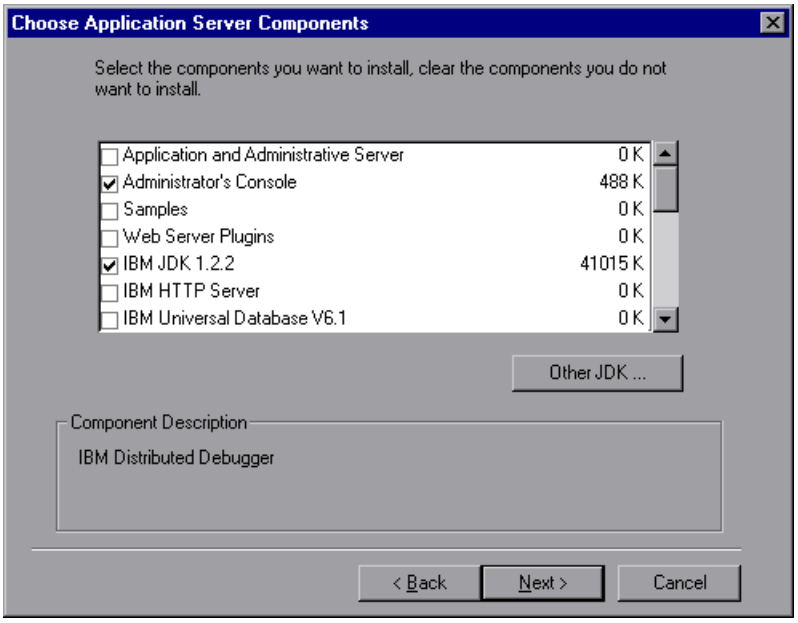


2 Disregard the instruction to shut down all Web servers that you plan to run with WebSphere. This is not relevant to the Administrator's Console installation. Follow the remaining instructions.

3 Click **Next** to proceed. The Installation Options dialog box opens.

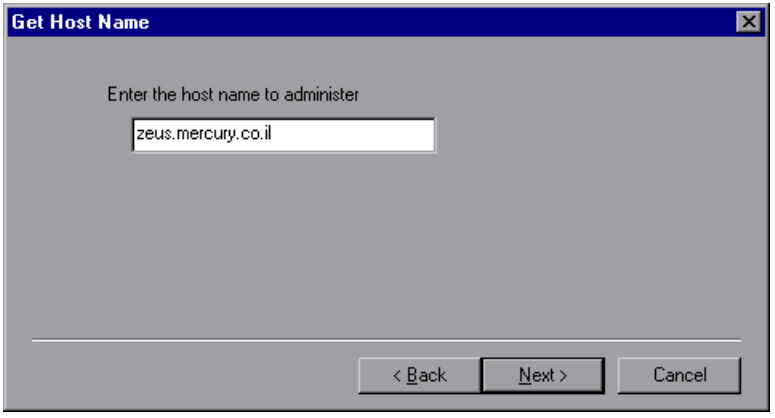


4 Select **Custom Installation**, and click **Next**. The Choose Application Server Components dialog box opens.



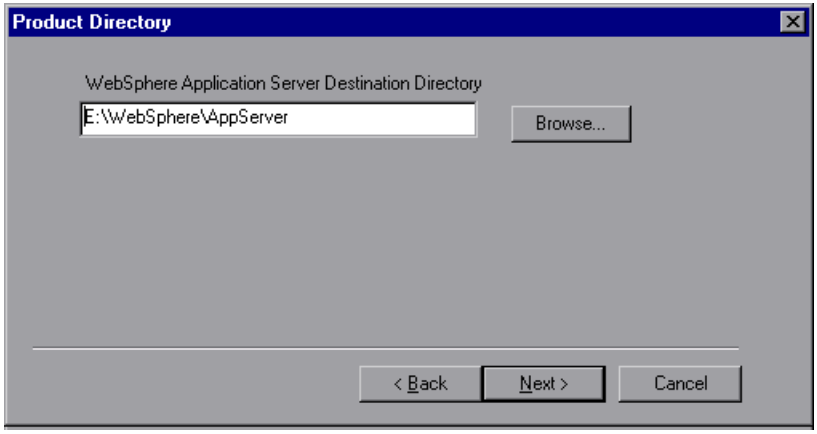
5 Select **Administrator's Console** and **IBM JDK 1.2.2**. Clear all the other options.

6 Click **Next**. The Get Host Name dialog box opens.



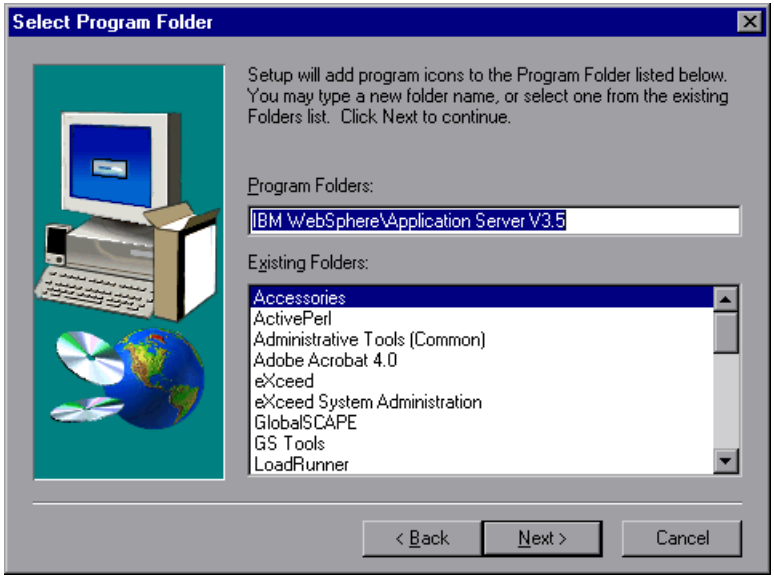
7 Type the name of the machine that you want to monitor.

8 Click **Next**. The Product Directory dialog box opens.



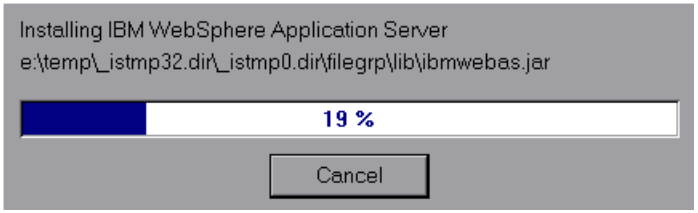
9 Specify the folder in which to install the Administrator's Console. To select a different location, click **Browse**, select a folder other than the default folder, and click **OK**.

10 Click **Next**. The Select Program Folder dialog box opens.

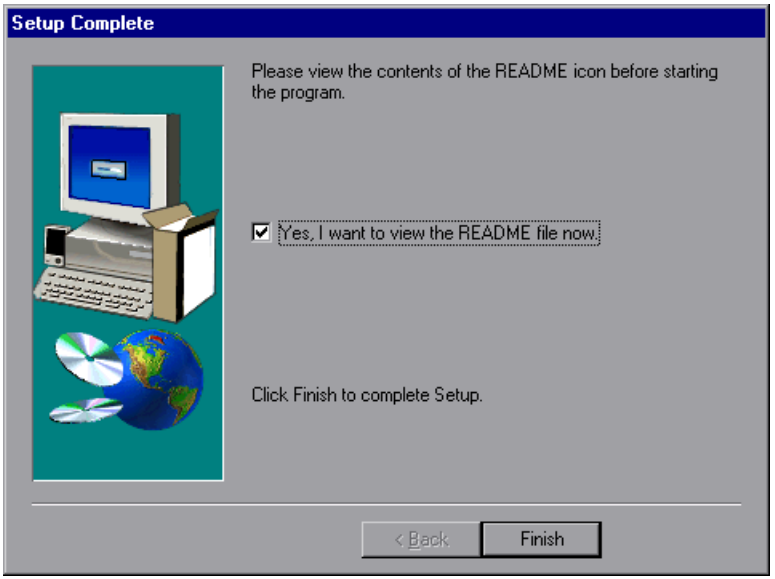


11 Specify a program folder, or accept the default folder, IBM WebSphere\Application Server V3.5.

12 Click **Next**. The installation process begins. To pause or quit the installation, click **Cancel**.

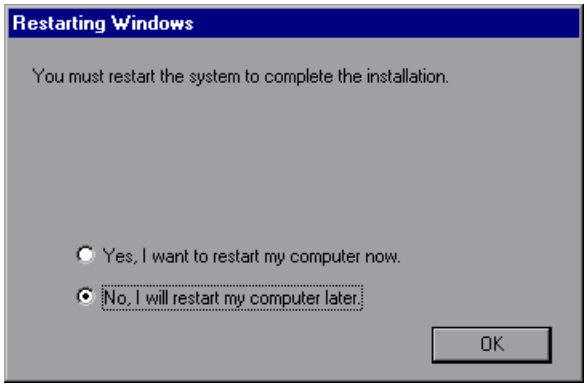


When the installation is complete, the Setup Complete dialog box opens.



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.



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 <**Performance Center root folder**>**\dat\monitors\WebSphere35Mon.ini** file in a text editor.

3 Locate the following line:

```
JVM_CLASSES4=C:\WebSphere\AppServer\lib\353Keyring.ja
```

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. To manage different levels of performance data, right-click the performance modules and select a performance level. Click the **Set** button.

Alternatively, check 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

Configuring the WebSphere (EPM) Monitor

Once you have installed the WebSphere Administrator's Console and enabled the EPM counters, you select measurements to monitor the WebSphere EPM server using the WebSphere EPM Choose Measurements page.

To activate the WebSphere EPM monitor:

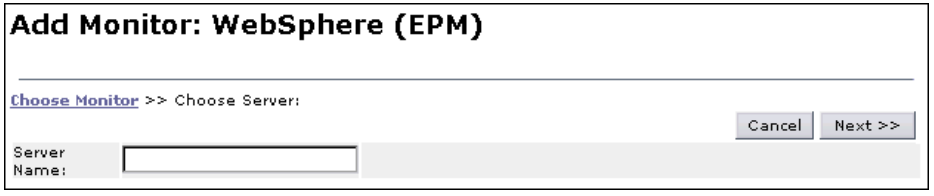
1 Open the Add Monitor window.

- **Before running a load test:** In the Load Test configuration page's **Monitors** tab, select the type of monitor profile to create:
  - To create a monitor profile for the current load test, click **Add Local Profile**.
  - To create a monitor profile that can be used for other load tests, click **Add Profile**, type a name and description of the profile and click **OK**.

  For details, see the section about configuring monitors in the *HP Performance Center User Guide*.
- **During a load test run:** On the Load Test run page:
  - Click the **Design** button. The Load Test Design dialog box opens.
  - Click **Monitors**.

  For details, see the section about creating monitor profiles while running a load test in the *HP Performance Center User Guide*.

2 Under **Web Application Server Resource Graphs**, select **WebSphere EPM** and click **Next**. The Choose Server page opens.



3 Type the name or IP address of the server whose resources you want to monitor.

4. Click **Next**. The Choose Measurements page opens, displaying the available measurements and server properties.
5. Select the measurements you want to monitor.

   For a list of the available performance counters, see "WebSphere Performance Counters" on page 243.
6. Click **Save** to add the selected resource measurements to the monitor profile.

Part IX

Database Resource Monitoring

33

Introducing Database Resource Monitoring

To isolate server performance bottlenecks, you can monitor the database server resource usage of an SQL, DB2, Sybase, and Oracle database server during a load test run.

To obtain performance data, you need to activate the online monitor for the server and specify which resources you want to measure before running the load test.

For more information, see:

- Chapter 34, “SQL Server Monitoring”
- Chapter 35, “DB2 Database Server Monitoring”
- Chapter 36, “Oracle Database Server Monitoring”
- Chapter 37, “Sybase Database Server Monitoring”

34

SQL Server Monitoring

The SQL Server monitor measures the standard Windows resources on the SQL server machine.

This chapter includes:

- Configuring the SQL Server Monitor on page 273
- SQL Server Performance Counters on page 276

Configuring the SQL Server Monitor

To obtain SQL Server performance data, you need to activate the online monitor for the server and specify which resources you want to measure before running the load test. You select measurements to monitor using the SQL Server Choose Measurements page.

We recommend using the SiteScope monitor engine. In this case, SiteScope must be installed on a server. Although SiteScope is installed on the Utility Server, we recommend installing it on the same machine as the Controller, or on a dedicated server in order to reduce load on the Utility Server.

Note: To monitor an SQL server through a firewall, use TCP, port 139.

To configure the SQL server monitor:

1 Open the Add Monitor window.

- **Before running a load test:** In the Load Test configuration page's **Monitors** tab, select the type of monitor profile to create:
  - To create a monitor profile for the current load test, click **Add Local Profile**.
  - To create a monitor profile that can be used for other load tests, click **Add Profile**, type a name and description of the profile and click **OK**.

  For details, see the section about configuring monitors in the *HP Performance Center User Guide*.
- **During a load test run:** On the Load Test run page:
  - Click the **Design** button. The Load Test Design dialog box opens.
  - Click **Monitors**.

  For details, see the section about creating monitor profiles while running a load test in the *HP Performance Center User Guide*.

2 Under **Database Server Resource Graphs**, select **SQL Server** or **SQL Server (SiteScope)** and click **Next**. The Choose Server page opens.

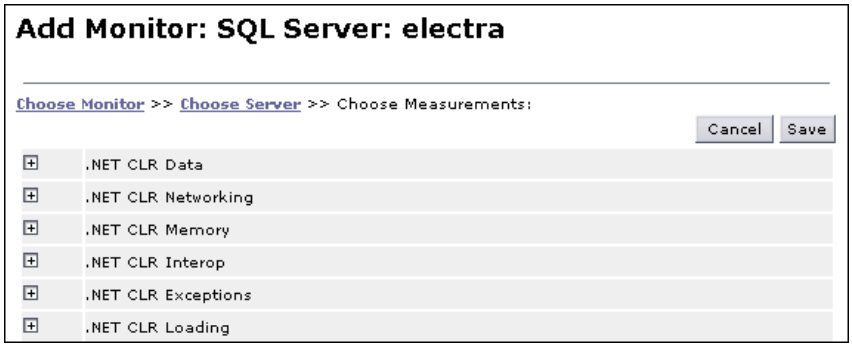
3 Type the name or IP address of the SQL server whose resources you want to monitor.

- For SQL Server, type the user login name and password on the SQL server.
- For SQL Server (SiteScope), type the SiteScope machine name and port (default: 8888), and specify whether you are using a Secure HTTP connection or a SiteScope user account. If you are using a SiteScope account, fill in the account name or number, and the user name and password defined to log in to the SiteScope account.

Click **Next**.

4 The first time you add measurements using the **SQL Server (SiteScope)** monitor you need to configure the NT remote machine properties. Enter the remote machine's configuration information, as described in "Configuring the NT Remote Machine" on page 27.

5 The Choose Measurements page opens, displaying the available measurements and server properties.



6 Select the measurements you want to monitor.

For a list of the available performance counters, see "SQL Server Performance Counters" on page 276.

7 Click **Save** to add the selected resource measurements to the monitor profile.

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 the section that describes useful counters for stress testing in the *HP LoadRunner Analysis User Guide*.

SQL Server Performance Counters

The following table describes some of the counters that can be measured 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, 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 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 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

35

DB2 Database Server Monitoring

The DB2 Database Server monitor measures the resource usage on a DB2 database during a load test run.

Note: If there is no application working with a database, you can only monitor the database manager instance.

This chapter includes:

- Setting Up the Monitoring Environment on page 279
- Configuring the DB2 Monitor on page 281
- DB2 Performance Counters on page 282

Setting Up the Monitoring Environment

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 Performance Center Host machine.
2. Install all the client files and libraries on the Utility Server.
3. Select **Start** > **Programs** > **DB2 for Windows NT** > **Control Center**. Type your DB2 server user name and password (with administrative privileges).
4. In the console that opens, right-click **Systems**, and select **Add**.

5 Enter the following settings in the dialog box:

- **System Name**. The server name.
- **Remote Instance**. DB2.
- **Host Name**. The host server name.
- **Service Name**. The DB2 server port. The default value is 50000.

6 Click **Retrieve** and **OK**.

Note: If you receive an error message after clicking **Retrieve**, repeat steps 4 and 5, and click **OK**.

---

7 Expand the <server name> node in the console tree.

8 Right-click **Instance**, and select **Add**.

9 Enter the following settings in the dialog box:

- **Remote Instance**. DB2
- **Instance Name**. The database instance to be called from the Controller.
- **Host Name**. The host server name.
- **Service Name**. The DB2 server port. The default value is 50000.

10 Click **OK** and close the Control Center.

Note: You can only work with a single Database Manager instance during each monitoring session.

Configuring the DB2 Monitor

To obtain performance data, you need to activate the DB2 monitor for the server and specify which resources you want to measure before running the load test. You select measurements to monitor using the DB2 Choose Measurements page.

To configure the DB2 monitor:

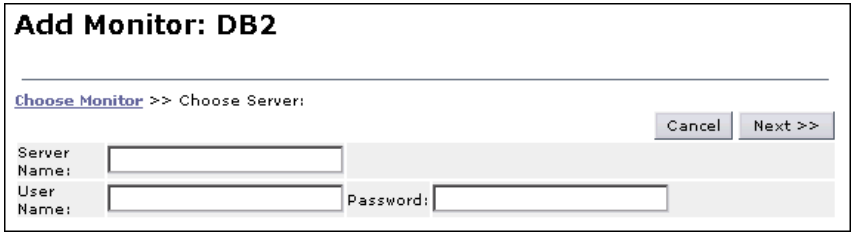
1 Open the Add Monitor window.

- **Before running a load test:** In the Load Test configuration page's **Monitors** tab, select the type of monitor profile to create:
  - To create a monitor profile for the current load test, click **Add Local Profile**.
  - To create a monitor profile that can be used for other load tests, click **Add Profile**, type a name and description of the profile and click **OK**.

  For details, see the section about configuring monitors in the *HP Performance Center User Guide*.
- **During a load test run:** On the Load Test run page:
  - Click the **Design** button. The Load Test Design dialog box opens.
  - Click **Monitors**.

  For details, see the section about creating monitor profiles while running a load test in the *HP Performance Center User Guide*.

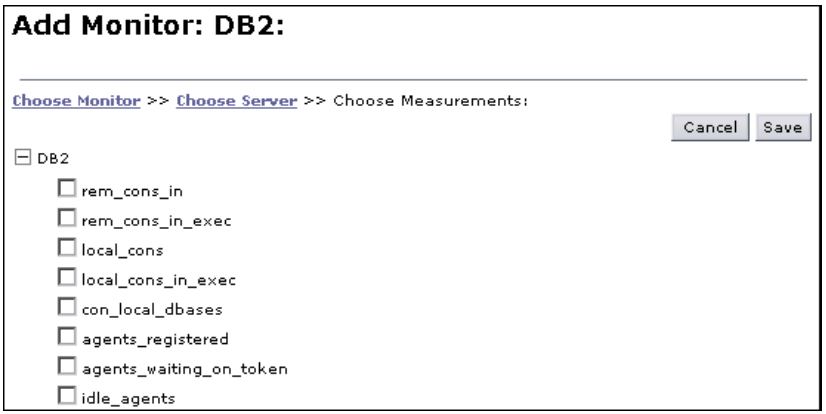
2 Under **Database Server Resource Graphs**, select **DB2** and click **Next**. The Choose Server page opens.



3 Type the name or IP address of the server whose resources you want to monitor.

Type the user login name and password.

4 Click **Next**. The Choose Measurements page opens, displaying the available measurements and server properties.



5 Select the measurements you want to monitor.

For a list of the available performance counters, see "DB2 Performance Counters" on page 282.

6 Click **Save** to add the selected resource measurements to the monitor profile.

DB2 Performance Counters

The following sections describe the default counters that can be monitored:

- "DatabaseManager Counters" on page 283
- "Database Counters" on page 284
- "Application Counters" on page 291

DatabaseManager Counters

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 Counters

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 pre-defined 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 Counters

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.

36

Oracle Database Server Monitoring

The Oracle database server resource monitor measures statistics for Oracle database servers. During a test run, you use this monitor to isolate Oracle database server performance bottlenecks.

The Oracle monitor displays information from Oracle V$ tables: Session statistics, V$SESSTAT, system statistics, V$SYSSTAT, and other table counters defined by the user in the custom query.

The SiteScope Oracle JDBC monitor monitors the server performance statistics from Oracle database servers. You can monitor multiple parameters or counters with a single monitor instance. This allows you to watch server loading for performance, availability, and capacity planning. You can create a separate Oracle JDBC Monitor instance for each Oracle database server in your environment.

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.

This chapter includes:

- Setting Up the Oracle Monitor Environment on page 298
- Configuring Oracle Database Resource Monitoring on page 302
- Oracle Performance Counters on page 306
- Custom Queries on page 307

Setting Up the Oracle Monitor Environment

Although SiteScope is installed on the Utility Server, we recommend installing it on the same machine as the Controller, or on a dedicated server in order to reduce load on the Utility Server.

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 before you can configure the monitor.

To set up the SiteScope Oracle JDBC monitor environment:

1 You must have a copy of the applicable Oracle JDBC database driver file (for example, **classes12.zip**) on the SiteScope server.

Copy the downloaded driver file into the **<SiteScope install path>\SiteScope\java\lib\ext** subdirectory. DO NOT unzip the file.

Stop and restart the SiteScope service after copying the driver file to the SiteScope machine.

Note: More than one driver file is available for download. Some drivers support more than one version of Oracle database (for example, the **classes12.zip** Oracle JDBC thin driver) while others only support a particular version. If you are monitoring a recent version of Oracle database, you should download the latest version of the database driver.

---

2 You must supply the correct Database Connection URL, a database user name and password when setting up the monitor. The syntax of the Database Connection URL usually has the form of:

```
jdbc:oracle:thin:@<tcp address>:<tcp port>:<database sid>
```

For example to connect to the ORCL database on a machine using port 1521 you would use:

```
jdbc:oracle:thin:@206.168.191.19:1521:ORCL
```

Note: The colon and @ symbols must be included as shown.

---

3 You must specify the Oracle Database Driver that was installed on the SiteScope server when setting up the monitor. The Database Driver for the Oracle thin JDBC driver is:

```
oracle.jdbc.driver.OracleDriver
```

4 You should only have one Oracle client installed on the SiteScope machine. If there is more that one client installed, SiteScope may report an error and be unable to connect to the database.

5 You must have an Oracle user login that SiteScope will use to access the Oracle server. In order to retrieve the Oracle database counters, the user that SiteScope will use for the Oracle JDBC Monitor should be able to execute all the SQL statements as found in the file **SiteScope\templates.applications\commands.oraclejdbc**.

To set up the Performance Center Oracle monitor environment:

1 Check that the Oracle client libraries are installed on the Performance Center Host and Utility server machines.

2 Verify that **%OracleHome%\bin** is included in the path environment variable. If it is not, add it.

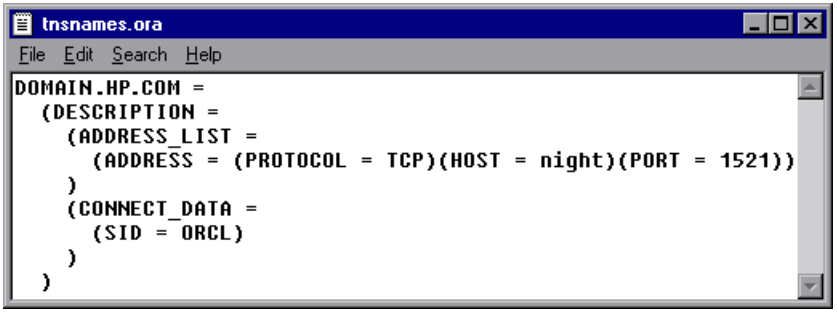
3 Configure the **tnsnames.ora** file on the Performance Center Host machine so that the Oracle client can communicate with the Oracle servers you plan to monitor.

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:



Note: Only the 32-bit Oracle client should be installed on the Performance Center Host machine running the Oracle monitor. If you have a 16-bit and a 32-bit Oracle client installation on the Performance Center Host machine, the 16-bit installation should be uninstalled.

---

4. Obtain a user name and password for the service from your database administrator, and check that the Performance Center Host 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 Performance Center Host 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 Check 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 each Performance Center Host machine and the Utility server. Attempt to log in to the Oracle servers with the desired user name/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 Performance Center 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.

If a problem occurs in setting up the Oracle environment, view the error message issued by the Oracle server.

Configuring Oracle Database Resource Monitoring

After you have set up the Oracle monitor environment, you can select the objects you want to measure.

To configure the Oracle monitor:

1 Open the Add Monitor window.

- **Before running a load test:** In the Load Test configuration page's **Monitors** tab, select the type of monitor profile to create:
  - To create a monitor profile for the current load test, click **Add Local Profile**.
  - To create a monitor profile that can be used for other load tests, click **Add Profile**, type a name and description of the profile and click **OK**.

  For details, see the section about configuring monitors in the *HP Performance Center User Guide*.
- **During a load test run:** On the Load Test run page:
  - Click the **Design** button. The Load Test Design dialog box opens.
  - Click **Monitors**.

  For details, see the section about creating monitor profiles while running a load test in the *HP Performance Center User Guide*.

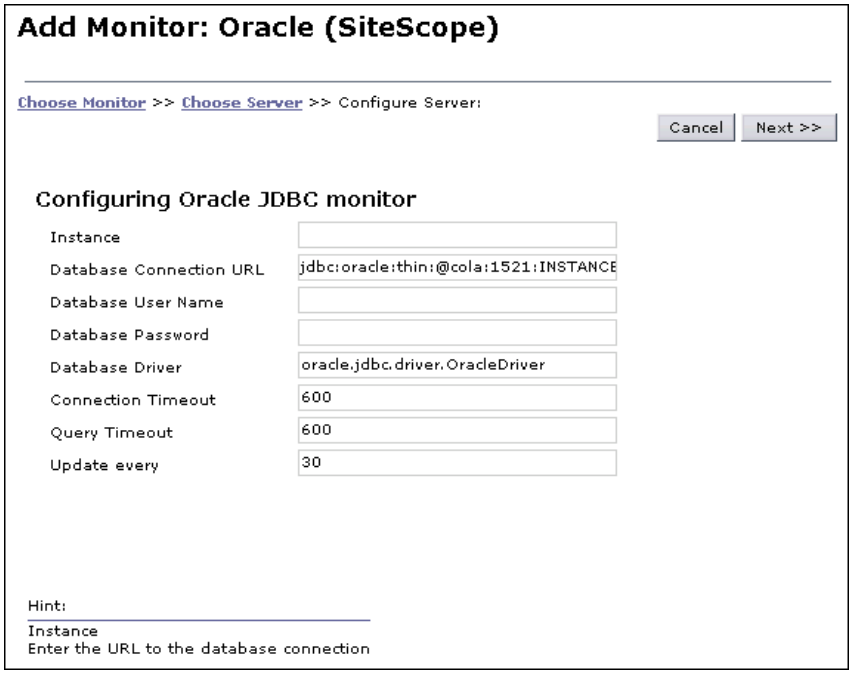
2 Under **Database Server Resource Graphs**, select **Oracle** or **Oracle (SiteScope)** and click **Next**. The Choose Server page opens.

3 Type the name or IP address of the server whose resources you want to monitor. In addition:

- For Oracle, type the user login name and password on the Oracle server.
- For Oracle (SiteScope), type the SiteScope machine name and port (default: 8888), and specify if you are using a Secure HTTP connection or a SiteScope user account. If you are using a SiteScope account, fill in the account name or number, and the user name and password defined to log in to the SiteScope account.

Click **Next**.

4 For the **Oracle (SiteScope)** monitor, the Configuring Oracle JDBC monitor page opens.



Enter the following Oracle JDBC monitor information, and click **Next**:

- **Instance.** Type the database SID. For example, ORCL.
- **Database Connection UR**. Type the connection URL to the database you want to monitor. For example, jdbc:oracle:thin:@206.168.191.19:1521:ORCL
- **Database User Name**. Type the user name that SiteScope should use to connect to the database.
- **Database Password**. Type the password for the user name that SiteScope should use to connect to the database.
- **Database Driver**. Type the driver used to connect to the database. For example, oracle.jdbc.driver.OracleDriver.

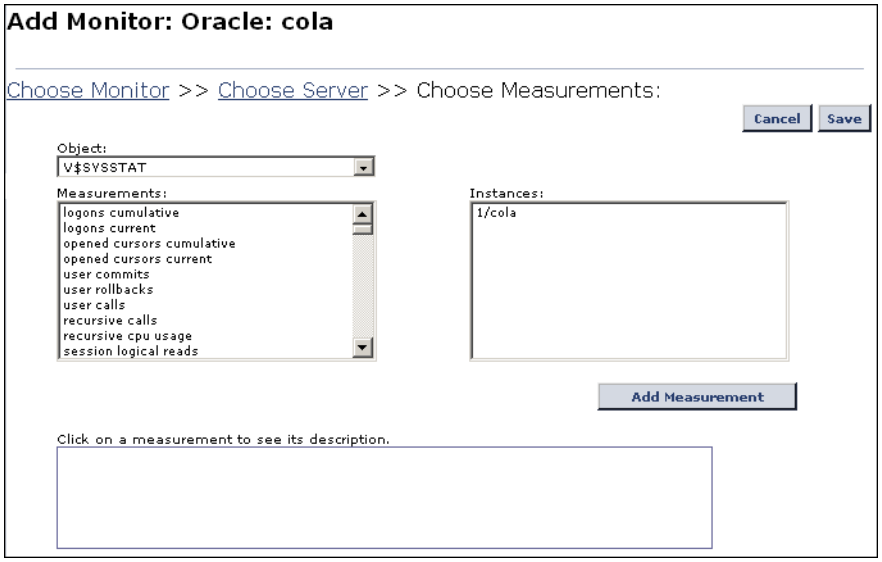
- **Connection Timeout**. Type an optional time out value, in seconds, that SiteScope should to wait for a database connection to respond. If the database connection can not be completed within the period specified, SiteScope will report an error.
- **Query Timeout**. Type an optional the time out value, in seconds, that SiteScope should to wait for a response from the database query. If the database does not respond within the period specified, SiteScope will report an error.
- **Update every**. Select how often the monitor should read the server statistics. The default interval is to run or update the monitor once every 30 seconds.

Note:

- The sum of the **Connection Timeout** value and **Query Timeout** value should always be less than the **Update every** value for the monitor.
- Some commonly used databases and database drivers do not support the **Query Timeout** feature. In these cases the **Query Timeout** value should be set to zero.

---

5 The Choose Measurements page opens, displaying the available measurements and server properties.



6 For each measurement that you want to monitor, select an object, measurements, and instances. The instances are relevant only if multiple instances of the highlighted instance are running.

7 Click **Save** to add the selected resource measurements to the monitor profile.

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.

Oracle Performance Counters

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
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

Measurement	Description
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) to work against the database.

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 the V$SYSSTAT and V$SESSTAT table counters that are currently provided by the Oracle monitor, as well as other tables that contain useful performance information.

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, type the name of the new counter.
5. In the Description line, type the description of the counter that you want the help message to contain.
6. In the Query line, type 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.

Note: Custom queries should not exceed 512 characters.

---

7. In the IsRate line, type 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, type 1.

Note: Custom queries cannot return negative values.

37

Sybase Database Server Monitoring

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 (through the Adaptive Server Enterprise Monitor Server) and retrieves metrics from the server using standard, Sybase-provided libraries.

To monitor the Sybase database server machine, you must first set up the Sybase monitoring environment. You then enable the Oracle online monitor and select the default measurements you want to display, before running the load test.

Although SiteScope is installed on the Utility Server, we recommend installing it on the same machine as the Controller, or on a dedicated server in order to reduce load on the Utility Server.

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**.

This chapter includes:

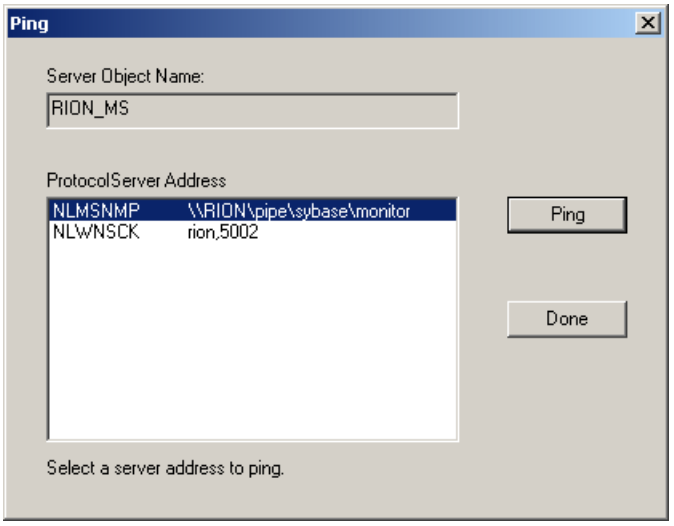
- Setting Up the Monitoring Environment on page 310
- Configuring the Sybase Monitor on page 311
- Sybase Performance Counters on page 314

Setting Up the Monitoring Environment

You need to set up the Sybase monitor environment before you can configure the monitor.

To set up the Sybase monitor environment:

1. Install the Sybase client files and libraries on the Controller machine.
2. Verify a connection between the client and server on the Controller machine. To do so, use the Sybase client's **dsedit** tool to ping the Adaptive Server Enterprise Monitor Server.



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.

Configuring the Sybase Monitor

After you have set up the Sybase monitor environment, you can select the objects you want the monitor to measure.

To configure the Sybase monitor:

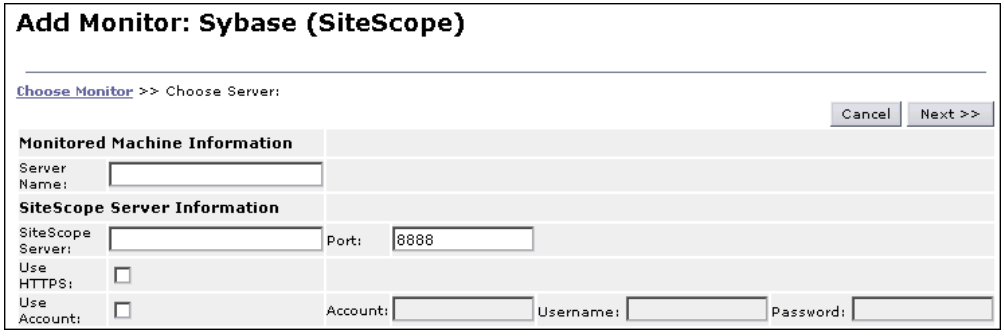
1 Open the Add Monitor window.

- **Before running a load test:** In the Load Test configuration page's **Monitors** tab, select the type of monitor profile to create:
  - To create a monitor profile for the current load test, click **Add Local Profile**.
  - To create a monitor profile that can be used for other load tests, click **Add Profile**, type a name and description of the profile and click **OK**.

  For details, see the section about configuring monitors in the *HP Performance Center User Guide*.
- **During a load test run:** On the Load Test run page:
  - Click the **Design** button. The Load Test Design dialog box opens.
  - Click **Monitors**.

  For details, see the section about creating monitor profiles while running a load test in the *HP Performance Center User Guide*.

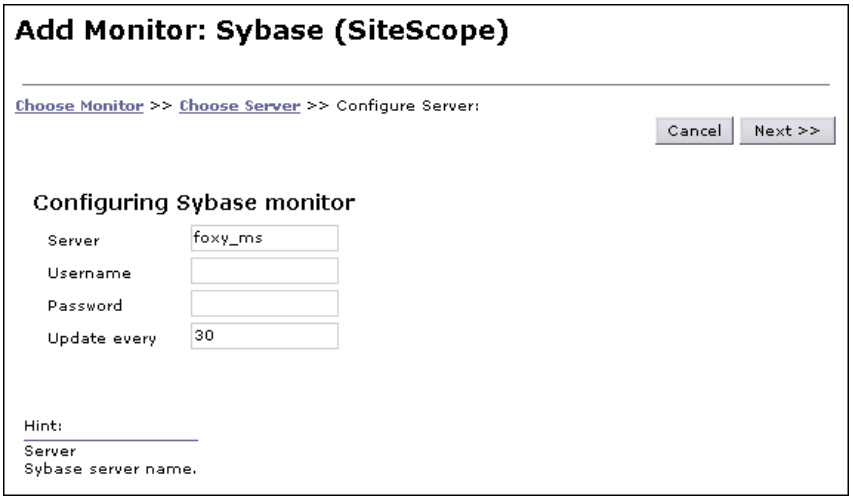
2 Under **Database Server Resource Graphs**, select **Sybase (SiteScope)** and click **Next**. The Choose Server page opens.



3 Under **Monitored Machine Information**, type the name or IP address of the server whose resources you want to monitor.

Under **SiteScope Server Information**, type the SiteScope machine name and port (default: 8888), and specify if you are using a Secure HTTP connection or a SiteScope user account. If you are using a SiteScope account, fill in the account name or number, and the user name and password defined to log in to the SiteScope account.

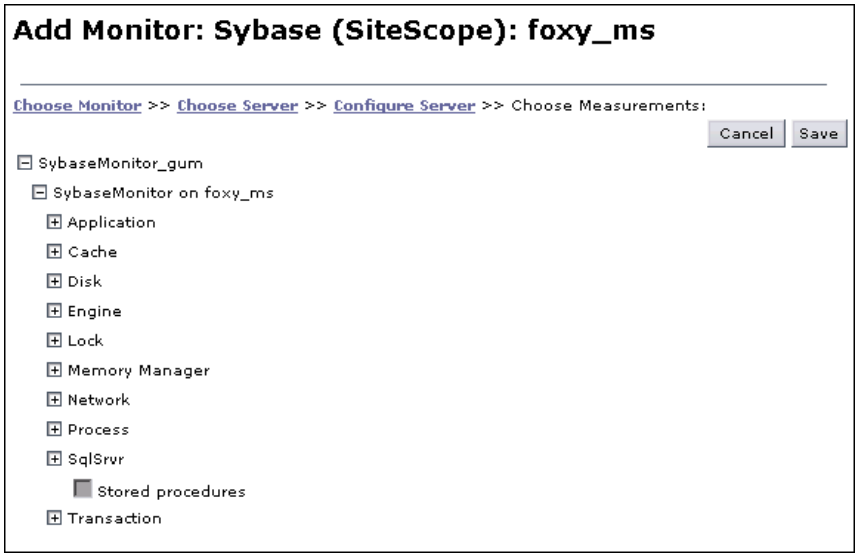
4 Click **Next**. The Configuring Sybase monitor page opens.



Enter the following server information:

- **Server**. Type the name of the server (usually the same name as the Sybase server but with the suffix _ms).
- **Username**. Type the login name for the Sybase server.
- **Password**. Type the password for the Sybase server.
- **Update every**. The number entered here indicates the amount of time, in seconds, between one monitor check and the next. By default the monitor updates every 30 seconds.

5 Click **Next**. The Choose Measurements page opens, displaying the available measurements and server properties.



6 Select the measurements you want to monitor.

For a list of the available performance counters, see "Sybase Performance Counters" on page 314.

7 Click **Save** to add the selected resource measurements to the monitor profile.

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 the section that describes useful counters for stress testing in the *HP LoadRunner Analysis User Guide*.

Sybase Performance Counters

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

Object	Measurement	Description
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
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

Object	Measurement	Description
Locks	**% Requests**	Reports the percentage of successful requests for locks
	Locks count	Reports the number of locks. This is an accumulated value.
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

Object	Measurement	Description
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
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

Object	Measurement	Description
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

Part X

J2EE Performance Monitoring

38

J2EE Performance Monitoring

The J2EE performance monitor provides complete insight into the J2EE components on the application server (Servlets, JSPs, EJBs, JNDI, JDBC, and DB SQL calls).

This chapter includes:

- About J2EE Performance Monitoring on page 323
- Setting Up the Monitoring Enviroment on page 324
- Configuring the J2EE Monitor on page 329
- J2EE Performance Counters on page 331
- Examples of Modifying Application Server Configuration on page 332

About J2EE Performance Monitoring

The J2EE monitor provides the following information for each J2EE component:

- Average response time per method/query
- Number of method calls per second

With such coverage of the J2EE architecture, users can get an overview of the entire activity within the system. They can very easily correlate the end user response time with the Web server activity (Servlets and JSPs data), application server activity (JNDI and EJB's), and back-end activity of database requests (JDBC methods and SQL queries).

The J2EE Monitor allows Performance Center users to analyze J2EE component metrics during a load test run by using an agent which is installed on the application server to collect information on the J2EE components. These measurements are sent from the application server back to the Performance Center Host through a Web server contained in the J2EE monitor. The J2EE Monitor supports the leading applications servers, such as: IBM WebSphere, BEA WebLogic, Oracle 9iAS and JBoss. For information about the supported application servers, see "Support Matrix" on page 324.

Note: The J2EE Monitor requires MSXML 3.0 and later (this is included in Internet Explorer 6.0). You can install MSXML 3.0 from the Microsoft MSDN Web site (http://msdn2.microsoft.com/en-us/xml/bb190622.aspx).

Setting Up the Monitoring Enviroment

To monitor J2EE objects, you must first install and activate the J2EE monitor on the application server machine. You then configure the J2EE monitor on the client machine by selecting the counters you want the monitor to measure.

You can monitor Java 2 Platform, Enterprise Edition (J2EE) objects on a WebLogic, WebSphere, Oracle 9iAS, or JBoss application server during a load test run using the J2EE performance monitor.

Installing the J2EE Monitor on the Application Server

To monitor J2EE objects, you must first install and activate the J2EE monitor on the application server machine.

Support Matrix

Application Server	Version	Platform
WebLogic	4.x; 5.x; 6.x; 7.0; 8.1	Windows; Solaris; AIX
WebSphere	3.x; 4.x	Windows; Solaris; AIX

Application Server	Version	Platform
Oracle 9iAS	1.0.2.2	Windows; Solaris; AIX
JBoss	2.4.x; 3.04	Windows; Solaris; AIX

To install the J2EE monitor on the application server:

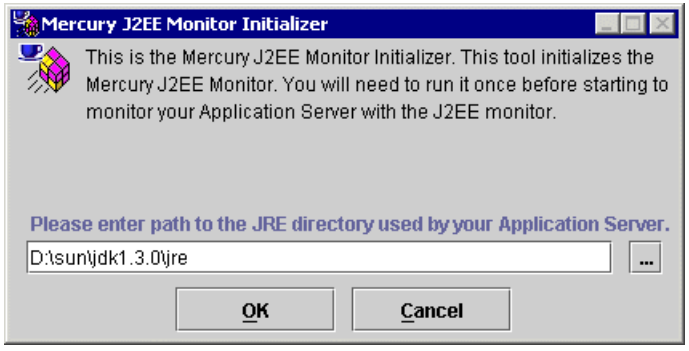
1 Create a home directory on the application server machine—for example, J2EEMonitor, and unzip the installation file **<Performance Center CD>\Additional Components\J2EE_Monitors\jmonitor_<platform>.jar** file into that directory.

If you do not have WinZip to unzip the installation file, use the following command line to extract the installation file:

```
<JDK>\bin\jar.exe -xf <installation file>
```

UNIX scripts extracted from the jar file may lose their execute permissions. To fix this, open the J2EEMonitor Home Directory, and change the permissions using the command line: chmod +x *.sh.

2 Open the J2EE Monitor Initializer by double-clicking **<J2EEMonitor Home Directory>\lib\sipatool.jar**.



On a UNIX platform, or if the **.jar** extension in your system is not associated with the Java run time environment, run **sipatool.jar** as follows:

From the **<J2EEMonitor Home Directory>\lib** directory, type

```
java -jar sipatool.jar
```

Note: If the J2EE Monitor Initializer is not displayed, run **sipatool.jar** as follows:

From the **<J2EEMonitor Home Directory>\lib** directory, type <Java path>\ -jar sipatool.jar -nogui. Use the full path of the Java process of the application server JDK.

---

3 In the J2EE Monitor Initializer dialog box, type the path to the application server Java home directory, and click **OK** to run the tool.

4 Add -Xbootclasspath/p:<J2EEMonitor Home Directory>\classes\boot to the application server command line arguments.

For more information about syntax for WebLogic, WebSphere, Oracle 9iAS, or JBoss application servers, see "Examples of Modifying Application Server Configuration" on page 332.

Initial J2EE Monitor Configuration Settings

During installation, the following was configured automatically on the J2EE monitor application server:

- **Hooking mechanism**. The J2EE monitor uses the J2EE Monitor Initializer and Java hooking library.
- **Operation mode**. The J2EE monitor uses the Auto Discovery operating mode. In this mode, the system automatically discovers the J2EE components (Servlet, JSP, JNDI, EJB and JDBC) that actually participate in the business process.
- **JDBC information retrieval**. The JDBC information retrieval setting determines which data to return from the JDBC call. By default, the J2EE monitor aggregates the measured data according to the JDBC operation, for example: SELECT,UPDATE,CREATE. To modify this configuration, see "Configuring JDBC Information Retrieval" on page 327.

- **EJB information retrieval**. The EJB information retrieval setting determines which data to return from the EJB call. By default, the J2EE monitor is not configured to measure container methods, (for example, ejbPassivate(), ejbCreate()). To modify this configuration, see "Configuring the EJB Information Retrieval" on page 328.

Note: For information about alternative configuration settings, contact the HP Software Support Web site (http://www.hp.com/go/hpsoftwaresupport).

Configuring JDBC Information Retrieval

To configure JDBC information retrieval:

1. Open **<J2EEMonitor Home Directory>\etc\dispatcher.properties**.
2. In the property sql.parsing.mode, type one of the following:
   - To measure the JDBC the method calls, like any other (non-JDBC) measured method calls, type 1.
   - To aggregate the measured data according to the JDBC operation, for example: SELECT, UPDATE, CREATE, type 2.
   - To aggregate the measured data according to specific SQL statement (including the operation, the tables it acted on, and other parameters of this statement), type 3.

Note: SQL Statements that exceed 3000 characters in length are not supported.

Configuring the EJB Information Retrieval

To configure EJB information retrieval to include container methods:

1 Open **<J2EEMonitor Home Directory>\dat\java_monitor.ini**.

2 In the EJB_CONFIG section of the file, change the

```
hook_files=auto_detect
```

setting to the following:

```
hook_files=auto_detect_container
```

Starting the Auto Discovery Process

In Auto Discovery mode (the J2EE monitor's default operating mode), the system discovers which methods of the components (Servlet, JSP, JNDI, EJB and JDBC) are participating in your business process and measures those objects only.

Before configuring the J2EE monitor you need to start the Auto Discovery process. To do this, start the application server, and run the Vuser script that you intend to use in your load test against the application server. This provides the Performance Center Host with a list of measurements that will be available for monitoring.

Note: The next time you run the same script, you do not need to run a Vuser before selecting the methods and counters you want to monitor.

Configuring the J2EE Monitor

To monitor J2EE performance, you must select the counters you want the J2EE monitor to measure. You select these counters using the J2EE Choose Measurements page.

To configure the J2EE monitor:

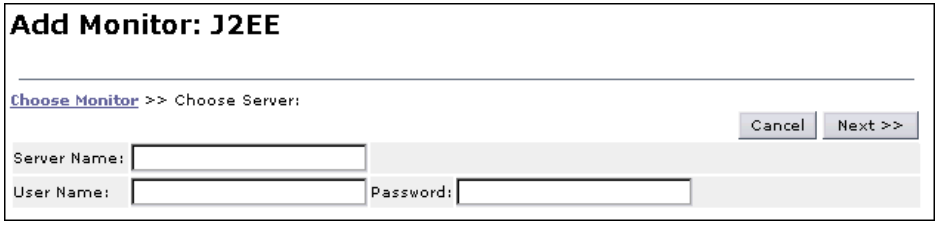
1 Open the Add Monitor window.

- **Before running a load test:** In the Load Test configuration page's **Monitors** tab, select the type of monitor profile to create:
  - To create a monitor profile for the current load test, click **Add Local Profile**.
  - To create a monitor profile that can be used for other load tests, click **Add Profile**, type a name and description of the profile and click **OK**.

  For details, see the section about configuring monitors in the *HP Performance Center User Guide*.
- **During a load test run:** On the Load Test run page:
  - Click the **Design** button. The Load Test Design dialog box opens.
  - Click **Monitors**.

  For details, see the section about creating monitor profiles while running a load test in the *HP Performance Center User Guide*.

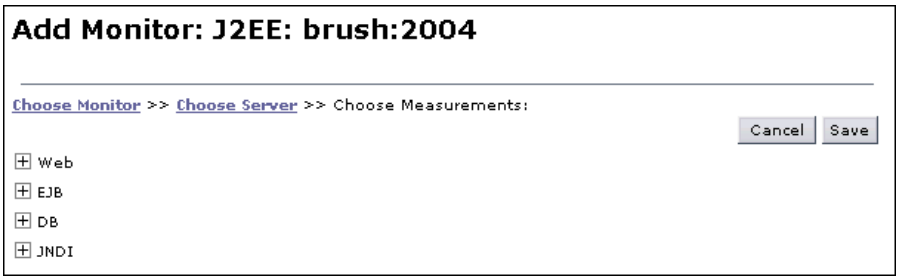
2 Under **Java Performance Graphs**, select **J2EE** and click **Next**. The Choose Server page opens.



3 Type the name or IP address of the server whose resources you want to monitor.

Type the user login name and password.

4 Click **Next**. The Choose Measurements page opens, displaying the available measurements and server properties.



5 Select the methods and counters that you want to monitor.

For a list of the available performance counters, see "J2EE Performance Counters" on page 331.

6 Click **Save** to add the selected resource measurements to the monitor profile.

Note: The size of a measurement name that can be displayed in the Analysis is limited to 255 characters. If a measurement name exceeds this limit, the counter name is truncated, and given a unique ID (UID). If you monitor different events or make cross result graphs on the same counter, the UID will remain the same.

The measurement name is truncated as follows:

standard prefix/**counter truncated name**<**UID**>/monitored event

For example:

/DB/JDBC/weblogic.jdbc.rmi.SerialPreparedStatement/int executeUpdate()/**INSERT INTO orders (orderid _ userid _ orderdate _ shipaddr1 _ shipaddr2 _ shipcity _ shipstate _ shipzip _ shipcountry _ billaddr1 _ billaddr2 _ b <1**> /Average Response Time

The full measurement name appears in the Measurement Description box.

J2EE Performance Counters

The following counters can be monitored for each method:

Measurement	Description
Average Response Time	The average response time, in milliseconds, of the J2EE object being monitored.
Method Calls per Second	The number of J2EE object method calls per second.

Examples of Modifying Application Server Configuration

When you installed HP's J2EE monitor files on your application server, you configured it to run with J2EE monitor support. This section provides examples for modifying the configuration of the following application servers:

- WebLogic - Version 4.x-5.x
- WebLogic - Version 6.x
- WebLogic - Version 7.x
- WebLogic - Version 8.1
- WebSphere Server - Version 3.x
- WebSphere Server - Version 4.x
- Oracle 9iAS Server
- JBoss 2.4.x-3.04 Server

Note: It is important to set the environment variables in the order in which they appear below.

WebLogic - Version 4.x-5.x

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 Just before the Java command line used to start the server, add the following variables:

For Windows platforms:

```
set MERC_MONITOR_HOME=<J2EEMonitor Home Directory>
set JAVA_CLASSPATH=%JAVA_CLASSPATH%;
%MERC_MONITOR_HOME%\dat;
%MERC_MONITOR_HOME%\classes\xerces.jar
```

For UNIX platforms (csh):

```
MERC_MONITOR_HOME <J2EEMonitor Home Directory>
JAVACLASSPATH=$JAVACLASSPATH:
$MERC_MONITOR_HOME/classes/xerces.jar
```

4 In the same section of the file, add the following parameter to the Java command line:

```
-Xbootclasspath/p:%MERC_MONITOR_HOME%\classes\boot
```

For example:

```
%JAVA_HOME%\bin\java -ms64m -mx64m -
-Xbootclasspath/p:%MERC_MONITOR_HOME%\classes\boot
-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.

WebLogic - Version 6.x

To configure the WebLogic 6.x server:

1. Copy the <**WebLogic Home**>**config**\<**domain name**>**startWeblogic.cmd** file into <**WebLogic Home**>**config**\<**domain name**>**startWeblogicMercury.cmd** so that the file is backed up.
2. Open the <**WebLogic Home**>**config**\<**domain name**>**startWeblogicMercury.cmd** file.
3. Just before the java command line used to start the server, add the following variables:

   For Windows platforms:

   ```
   set MERC_MONITOR_HOME=<J2EEMonitor Home Directory>
   set CLASSPATH=%CLASSPATH%;%MERC_MONITOR_HOME%\dat;
   %MERC_MONITOR_HOME%\classes\xerces.jar
   ```

   For UNIX platforms:

   ```
   MERC_MONITOR_HOME=<J2EEMonitor Home Directory>
   CLASSPATH=$CLASSPATH:$MERC_MONITOR_HOME/dat:
   $MERC_MONITOR_HOME/classes/xerces.jar
   ```

4. In the same section of the file add a parameter to the command line:

   ```
   -Xbootclasspath/p:%MERC_MONITOR_HOME%\classes\boot
   ```

   For example:

   ```
   "%JAVA_HOME%\bin\java" -hotspot -ms64m -mx64m
   -Xbootclasspath/p:%MERC_MONITOR_HOME%\classes\boot
   -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.

WebLogic - Version 7.x

To configure the WebLogic 7.x server:

1 Copy the <**WebLogic Home**>**server****bin****startwls.cmd** file into <**WebLogic Home**>**server****bin****startwlsMercury.cmd** so that the file is backed up.

2 Open the <**WebLogic Home**>**server****bin****startwlsMercury.cmd** file.

3 Just before the java command line used to start the server, add the following variables:

For Windows platforms:

```
set MERC_MONITOR_HOME=<J2EEMonitor Home Directory>
set CLASSPATH=%CLASSPATH%;%MERC_MONITOR_HOME%\dat;
%MERC_MONITOR_HOME%\classes\xerces.jar
```

For UNIX platforms:

```
MERC_MONITOR_HOME=<J2EEMonitor Home Directory>
CLASSPATH=$CLASSPATH:$MERC_MONITOR_HOME/dat:
$MERC_MONITOR_HOME/classes/xerces.jar
```

4 In the same section of the file add a parameter to the command line:

```
-Xbootclasspath/p:%MERC_MONITOR_HOME%\classes\boot
```

For example:

```
"%JAVA_HOME%\bin\java" -hotspot -ms64m -mx64m
-Xbootclasspath/p:%MERC_MONITOR_HOME%\classes\boot
-classpath %CLASSPATH% -Dweblogic.Domain=mydomain
-Dweblogic.Name=myserver "-Dbea.home=f:\bea" "
-Djava.security.policy==f:\bea\wlserver7.0/lib/weblogic.policy"
-Dweblogic.management.password=%WLS_PW% weblogic.Server
```

5 Copy the <**domain name**>**startWeblogic.cmd** file into <**domain name**>**startWeblogicMercury.cmd** so that the file is backed up.

6 Open the <**domain name**>**startWeblogicMercury.cmd** file.

7 Find the call to the Weblogic server. For example:

```
call D:\bea\weblogic700\server\bin\startWLS.cmd
```

8 Change the call from **startWLS.cmd** to **startWLSMercury.cmd**, and save the file.

WebLogic - Version 8.1

To configure the WebLogic 8.1 server:

1 Copy the <**WebLogic Home**>**samples****domains**\<**domain name**>**startWeblogic.cmd** file into <**WebLogic Home**>**samples****domains**\<**domain name**>**startWeblogicMercury.cmd** so that the file is backed up.

2 Open the <**WebLogic Home**>**samples****domains**\<**domain name**>**startWeblogicMercury.cmd** file.

3 Just before the java command line used to start the server, add the following variables:

For Windows platforms:

```
set MERC_MONITOR_HOME=<J2EEMonitor Home Directory>
set CLASSPATH=%CLASSPATH%;%MERC_MONITOR_HOME%\dat;
%MERC_MONITOR_HOME%\classes\xerces.jar
```

For UNIX platforms:

```
MERC_MONITOR_HOME=<J2EEMonitor Home Directory>
CLASSPATH=$CLASSPATH:$MERC_MONITOR_HOME/dat:
$MERC_MONITOR_HOME/classes/xerces.jar
```

In the same section of the file add a parameter to the command line:

```
-Xbootclasspath/p:%MERC_MONITOR_HOME%\classes\boot
```

For example:

```
"%JAVA_HOME%\bin\java" -hotspot -ms64m -mx64m
-Xbootclasspath/p:%MERC_MONITOR_HOME%\classes\boot
-classpath %CLASSPATH% -Dweblogic.Domain=mydomain
-Dweblogic.Name=myserver "-Dbea.home=f:\bea" "
-Djava.security.policy==f:\bea\wlserver8.1/lib/weblogic.policy"
-Dweblogic.management.password=%WLS_PW% weblogic.Server
```

4. Copy the **<domain name>\startWeblogic.cmd** file into **<domain name>\startWeblogicMercury.cmd** so that the file is backed up.
5. Open the **<domain name>\startWeblogicMercury.cmd** file.
6. Find the call to the Weblogic server. For example:

```
call D:\bea\weblogic8\server\bin\startWLS.cmd
```

7. Change the call from **startWeblogic.cmd** to **startWeblogicMercury**, and save the file.

Note: WebLogic 8.1 supports SUN JVM only. JRockit JVM is available with WebLogic 8.1 using the J2EE Diagnostics monitor only.

WebSphere Server - Version 3.x

By default, the WebSphere 3.x application server runs on Windows as an automatic service, upon machine startup. Since HP does not currently support Performance Center J2EE 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 Performance Center J2EE monitor support to the WebSphere 3.x server:

1. Make a backup copy of **<WebSphere Home>\AppServer\bin\debug\adminserver.bat**.
2. Open **<WebSphere Home>\AppServer\bin\debug\adminserver.bat**.
3. Add the following environment variables at the end of the 'SET_CP' section:

For Windows platforms:

```
set MERC_MONITOR_HOME=<J2EEMonitor Home Directory>
set CLASSPATH=%CLASSPATH%;%MERC_MONITOR_HOME%\dat;
%MERC_MONITOR_HOME%\classes\xerces.jar
```

For UNIX platforms:

```
MERC_MONITOR_HOME=<J2EEMonitor Home Directory>
CLASSPATH=$CLASSPATH:$MERC_MONITOR_HOME/dat:
$MERC_MONITOR_HOME/classes/xerces.jar
export MERC_MONITOR_HOME
```

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. Add

   ```
   -Xbootclasspath/p:%MERC_MONITOR_HOME%\classes\boot
   ```

   to the command line Arguments box, and click **Apply**.
9. If you are working with a WebSphere 3.0 Server with JDK1.1.7 IBM, double-click **Environment**.
   a. In the **Variable Name** box, type _CLASSLOAD_HOOK.
   b. In the **Value** box, type jdkhook.

   Click **Add**, then **OK**, and then **Apply**.
10. For Windows 2000/NT or Solaris, from the General tab, open the Environment Editor dialog box and add the following variables to the Environment box:

    For Windows 2000/NT:

    ```
    name=CLASSPATH
    value=<J2EEMonitor Home Directory>\dat
    ```

    For Solaris:

    ```
    name=CLASSPATH
    value=<J2EEMonitor Home Directory>/dat
    ```

    Click **OK** to close the Environment Editor dialog box.
11. Close the WebSphere Advanced Administrative Console.
12. Close and restart the **adminserver.bat** file.

WebSphere Server - Version 4.x

You can start the WebSphere 4.x server using the **startServerBasic.bat** file or the **startServer.bat** file.

To configure the WebSphere 4.x server:

1. Check 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 J2EE monitor).
3. Right-click the Default Server, select Properties from the menu, and click the **General** tab.
4. For Windows 2000/NT or Solaris, from the General tab, open the Environment Editor dialog box and add the following variables to the Environment box:

   For Windows 2000/NT:

   ```
   name=CLASSPATH
   value=<J2EEMonitor Home Directory>\dat;
   <J2EEMonitor Home Directory>\classes\xerces.jar
   ```

   For Solaris:

   ```
   name=CLASSPATH
   value=<J2EEMonitor Home Directory>/dat:
   <J2EEMonitor Home Directory>/classes/xerces.jar
   ```

   Click **OK** to close the Environment Editor dialog box.
5. Click the Advanced JVM Settings tab and select Advanced JVM settings. In the Command line arguments field, add the following value for Windows 2000/NT, Solaris, and AIX:

   ```
   -Xbootclasspath/p:%MERC_MONITOR_HOME%\classes\boot
   ```
6. Click **OK** and **Apply** to save the changes for the application server. You can now start and stop your WebSphere server using the Performance Center J2EE monitor.

Oracle 9iAS Server

1 Edit the file **env.cmd** (**env.sh** in UNIX platforms) as follows:

- The JAVA_HOME environment variable should point to the location of the Java Virtual machine used to run the application server.
- The DETECTOR_INS_DIR environment variable should point to the location of the monitor installation.
- The APP_SERVER_DRIVE environment variable should specify the drive hosting the application server installation (for example, D:). Do not modify this variable on UNIX platforms.
- The APP_SERVER_ROOT environment variable should specify the application server root directory.

2 Run the **oc4jMonitor.cmd** (**oc4jMonitor.sh** on UNIX platforms).

JBoss 2.4.x-3.04 Server

1 Make a backup copy of <**JBoss Home**>**run.bat** (**run.sh** on UNIX platforms) file into <**JBoss Home**>**runMercury.bat** (**runMercury.sh** for UNIX).

2 Open <**JBoss Home**>**runMercury.bat** (**runMercury.sh** on UNIX).

Just before the Java command line used to start the server, add the following variables:

For Windows platforms:

```
set MERC_MONITOR_HOME=<J2EEMonitor Home Directory>
set CLASSPATH=%CLASSPATH%;%MERC_MONITOR_HOME%\dat;
%MERC_MONITOR_HOME%\classes\xerces.jar
```

For UNIX platforms:

```
MERC_MONITOR_HOME=<J2EEMonitor Home Directory>
CLASSPATH=$CLASSPATH:$MERC_MONITOR_HOME/dat:
$MERC_MONITOR_HOME/classes/xerces.jar
```

3 In the same section of the file, add the following parameter to the command line:

```
-Xbootclasspath/p:%MERC_MONITOR_HOME%\classes\boot
```

For example:

```
%JAVA_HOME%\bin\java -ms64m -mx64m  -
Xbootclasspath/p:%MERC_MONITOR_HOME%\classes\boot
-Dweblogic.class.path=%WEBLOGIC_CLASSPATH% -Dweblogic.home=.
-Djava.security.manager
-Djava.security.policy==.\weblogic.policy weblogic.Server
```

4 Run the <**JBoss Home**>**runMercury.bat** file (**runMercury.sh** on UNIX platforms).

Part XI

Application Deployment Monitoring

39

Application Deployment Solutions

Using Performance Center's Application Deployment Solution monitor, you can isolate server performance bottlenecks by monitoring the Citrix MetaFrame XP or 1.8 server during a load test run.

This chapter includes:

- Setting Up the Monitoring Environment on page 346
- Configuring the Citrix MetaFrame Server Monitor on page 347
- Citrix MetaFrame Performance Counters on page 350

Setting Up the Monitoring Environment

The Citrix MetaFrame XP monitor displays statistics about the resource usage on the Citrix MetaFrame XP or 1.8 server machine during the load test run.

To monitor the Citrix server performance, you must first activate the Citrix MetaFrame XP monitor on the application server machine and enable the counters you want to monitor on the Citrix server.

Note: The port you use to monitor a Citrix MetaFrame server through a firewall depends on the configuration of your server.

Before configuring the monitor:

1. From the Performance Center Host machine, map a network drive to the Citrix server machine. This provides the required authentication to the Performance Center Host to access the resource counters.
2. Launch PerfMon from the Performance Center Host machine to enable the counters on the Citrix server. This allows you to monitor the same counters for the ICA Session object on the Citrix monitor.
3. To provide the Performance Center Host with a list of measurements that will be available for monitoring, you must first initialize Vusers before running the load test. After you have initialized the Vusers, you can configure the Citrix Monitor and add the ICA Session counters.
4. Although SiteScope is installed on the Utility Server, we recommend installing it on the same machine as the Controller, or on a dedicated server in order to reduce load on the Utility Server.

Note:

- Measurements that monitor instances are valid for the currently running Citrix session only. If you run this load test again, you will need to reconfigure the measurements that are instance-oriented.
- To monitor the different instances, check that the server login and logout procedures are recorded in the **Vuser_init** and **Vuser_end** sections respectively, and not in the Action section of the script. For more information, see the *HP Virtual User Generator User Guide*.

Configuring the Citrix MetaFrame Server Monitor

After you have set up the monitoring enviroment, you select the counters you want the Citrix MetaFrame XP monitor to measure. You select these counters using the Citrix MetaFrame XP Choose Measurements page.

To configure the Citrix MetaFrame Server monitor:

1 Open the Add Monitor window.

- **Before running a load test:** In the Load Test configuration page's **Monitors** tab, select the type of monitor profile to create:
  - To create a monitor profile for the current load test, click **Add Local Profile**.
  - To create a monitor profile that can be used for other load tests, click **Add Profile**, type a name and description of the profile and click **OK**.

  For details, see the section about configuring monitors in the *HP Performance Center User Guide*.
- **During a load test run:** On the Load Test run page:
  - Click the **Design** button. The Load Test Design dialog box opens.
  - Click **Monitors**.

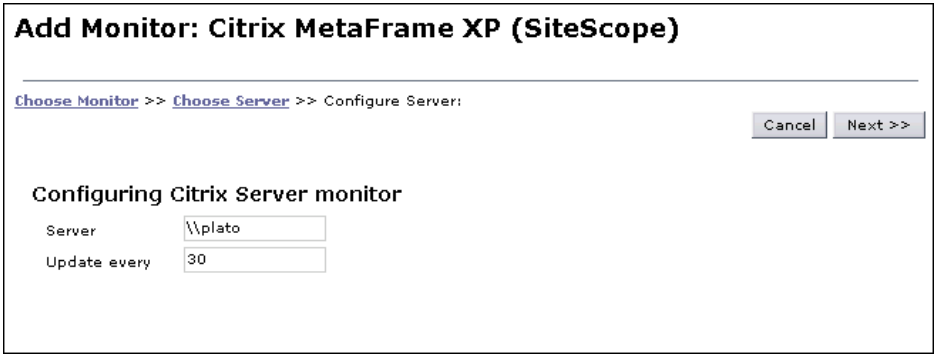
For details, see the section about creating monitor profiles while running a load test in the *HP Performance Center User Guide*.

2 Under **Application Deployment Solutions Graphs**, select **Citrix MetaFrame XP** or **Citrix MetaFrame XP (SiteScope)** and click **Next**. The Choose Server page opens.

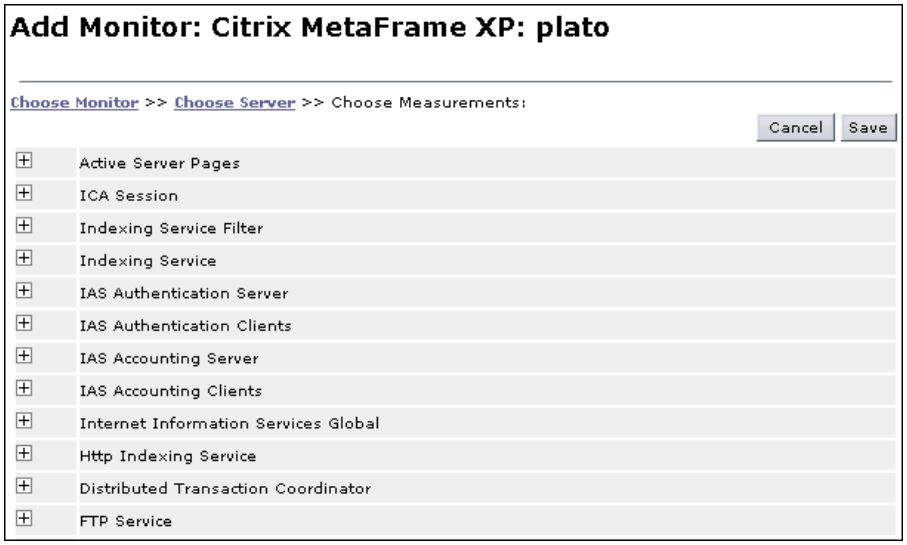
3 Type the name or IP address of the server whose resources you want to monitor. In addition:

- For Citrix MetaFrame XP Server, type the user login name and password on the Citrix server, and click **Next**.
- For Citrix MetaFrame XP Server (SiteScope), type the SiteScope machine name and port (default: 8888), and specify if you are using a Secure HTTP connection or a SiteScope user account. If you are using a SiteScope account, fill in the account name or number, and the user name and password defined to log in to the SiteScope account.

4 Click **Next**. The Add Monitor page opens. Verify the server and the update frequency, and click **Next**.



5 The Choose Measurements page opens, displaying the available measurements and server properties.



6 Select the measurements you want to monitor.

For a list of available performance counters, see “Citrix MetaFrame Performance Counters” on page 350.

7 Click **Save** to add the selected resource measurements to the monitor profile.

Note: If the dialog box freezes after clicking Add, you may need to rebuild the localhost cache on the Citrix server machine. For more information, see Document IDs CTX003648 and CTX759510 in the Citrix Knowledge Base (http://knowledgebase.citrix.com/cgi-bin/webcgi.exe?New,KB=CitrixKB).

Citrix MetaFrame Performance Counters

The following tables describe some of the counters that can be measured.

Non-Virtual Counters

Measurement	Description
% Disk Time	The percentage of elapsed time that the selected disk drive is busy servicing read or write requests.
% Processor Time	The percentage of time that the processor is executing a non-Idle thread. This counter is 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, then subtracting that value from 100%.
File data Operations/sec	The rate that the computer is issuing Read and Write operations to file system devices. It does not include File Control Operations.

Measurement	Description
Interrupts/sec	The average number of hardware interrupts the processor is receiving and servicing in each second. It does not include DPCs, which are counted separately. This value is an indirect indicator of the activity of devices that generate interrupts, such as the system clock, the mouse, disk drivers, data communication lines, network interface cards and other peripheral devices. These devices normally interrupt the processor when they have completed a task or require attention. Normal thread execution is suspended during interrupts. Most system clocks interrupt the processor every 10 milliseconds, creating a background of interrupt activity. This counter displays the difference between the values observed in the last two samples, divided by the duration of the sample interval.
Output Session Line Speed	This value represents the line speed from server to client for a session in bps.
Input Session Line Speed	This value represents the line speed from client to server for a session in bps.
Page Faults/sec	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 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 whom the page is shared.

Measurement	Description
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.
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, but instead remain in main memory as long as they are allocated.
Private Bytes	The current number of bytes this process has allocated that cannot be shared with other processes.
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.
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.
Latency – Session Average	The average client latency over the life of a session.

Measurement	Description
Latency – Last Recorded	Represents the last recorded latency measurement for this session.
Latency – Session Deviation	The difference between the minimum and maximum measured values for a session.
Input Session Bandwidth	The bandwidth from client to server traffic for a session in bps.
Input Session Compression	The compression ratio for client to server traffic for a session.
Output Session Bandwidth	The bandwidth from server to client traffic for a session (in bps).
Output Session Compression	The compression ratio for server to client traffic for a session.
Output Session Linespeed	The line speed from server to client for a session (in bps).

Virtual Channel Counters

Measurement	Description
Input Audio Bandwidth	The bandwidth from client to server traffic on the audio mapping channel (in bps).
Input Clipboard Bandwidth	The bandwidth from client to server traffic on the clipboard mapping channel (in bps)
Input COM1 Bandwidth	The bandwidth from client to server traffic on the COM1 channel (in bps).
Input COM2 Bandwidth	The bandwidth from client to server traffic on the COM2 channel (in bps).
Input COM Bandwidth	The bandwidth from client to server traffic on the COM channel (in bps).
Input Control Channel Bandwidth	The bandwidth from client to server traffic on the ICA control channel (in bps).

Measurement	Description
Input Drive Bandwidth	The bandwidth from client to server traffic on the client drive mapping channel (in bps).
Input Font Data Bandwidth	The bandwidth from client to server traffic on the local text echo font and keyboard layout channel (in bps).
Input Licensing Bandwidth	The bandwidth from server to client traffic on the licensing channel (in bps).
Input LPT1 Bandwidth	The bandwidth from client to server traffic on the LPT1 channel (in bps).
Input LPT2 Bandwidth	The bandwidth from client to server traffic on the LPT2 channel (in bps).
Input Management Bandwidth	The bandwidth from client to server traffic on the client management channel (in bps).
Input PN Bandwidth	The bandwidth from client to server traffic on the Program Neighborhood channel (in bps).
Input Printer Bandwidth	The bandwidth from client to server traffic on the printer spooler channel (in bps).
Input Seamless Bandwidth	The bandwidth from client to server traffic on the Seamless channel (in bps).
Input Text Echo Bandwidth	The bandwidth from client to server traffic on the local text echo data channel (in bps).
Input Thinwire Bandwidth	The bandwidth from client to server traffic on the Thinwire (graphics) channel (in bps).
Input VideoFrame Bandwidth	Tthe bandwidth from client to server traffic on the VideoFrame channel (in bps).
Output Audio Bandwidth	The bandwidth from server to client traffic on the audio mapping channel (in bps).
Output Clipboard Bandwidth	The bandwidth from server to client traffic on the clipboard mapping channel (in bps).
Output COM1 Bandwidth	The bandwidth from server to client traffic on the COM1 channel (in bps).

Measurement	Description
Output COM2 Bandwidth	The bandwidth from server to client traffic on the COM2 channel (in bps).
Output COM Bandwidth	The bandwidth from server to client traffic on the COM channel (in bps).
Output Control Channel Bandwidth	The bandwidth from server to client traffic on the ICA control channel (in bps).
Output Drive Bandwidth	The bandwidth from server to client traffic on the client drive channel (in bps).
Output Font Data Bandwidth	The bandwidth from server to client traffic on the local text echo font and keyboard layout channel (in bps).
Output Licensing Bandwidth	The bandwidth from server to client traffic on the licensing channel (in bps).
Output LPT1 Bandwidth	The bandwidth from server to client traffic on the LPT1 channel (in bps).
Output LPT2 Bandwidth	The bandwidth from server to client traffic on the LPT2 channel (in bps).
Output Management Bandwidth	The bandwidth from server to client traffic on the client management channel (in bps).
Output PN Bandwidth	The bandwidth from server to client traffic on the Program Neighborhood channel (in bps).
Output Printer Bandwidth	The bandwidth from server to client traffic on the printer spooler channel (in bps).
Output Seamless Bandwidth	The bandwidth from server to client traffic on the Seamless channel (in bps).
Output Text Echo Bandwidth	The bandwidth from server to client traffic on the local text echo data channel (in bps).
Output Thinwire Bandwidth	The bandwidth from server to client traffic on the Thinwire (graphics) channel (in bps).
Output VideoFrame Bandwidth	The bandwidth from server to client traffic on the VideoFrame channel (in bps).

Part XII

ERP/CRM Server Resource Monitoring

40

Introducing ERP/CRM Server Resource Monitoring

The ERP/CRM server resource monitors provide you with performance information for SAP, SAPGUI, SAP Portal, Siebel Server Manager, Siebel Web Server, and PeopleSoft (Tuxedo) servers. To monitor data for SAP CCMS (Computer Center Management System), use the SiteScope monitor. For more information, see "Configuring the SiteScope Monitor" on page 92.

To obtain this data, you need to activate the ERP/CRM server resource monitors before running the load test, and indicate which statistics and measurements you want to monitor.

Note: Performance Center no longer provides the SAP monitor. To monitor SAP server resources, use the SAPGUI monitor. Performance Center supports SAP monitor profiles for load tests saved in previous versions of Performance Center.

41

SAPGUI Monitoring

The SAPGUI monitor displays statistics about the resource usage of a SAP R/3 system during the load test run.

You can use the SAPGUI 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
- resource consumption for any application server
- application server workload for today or for a recent period

This chapter includes:

- Setting Up the Monitoring Environment on page 362
- Configuring the SAPGUI Monitor on page 362
- SAPGUI Performance Counters on page 365

Setting Up the Monitoring Environment

Before monitoring a SAP R/3 system server, you must enable the SAPGUI monitor.

To enable the SAPGUI monitor:

1. Install the SAPGUI for Windows 6.20 client on the Performance Center Host machine.
2. Install the latest patch for the SAPGUI for Windows 6.20 client. The lowest supported level is patch 36. (SAPGUI Patches can be downloaded from https://websmp104.sap-ag.de/patches.)
3. Check that you can access the st03 transaction and query for "last minute load" information with the user name and password defined in the Performance Center host.

Note: To monitor SAPGUI, you must change the DCOM configuration of Wlrun.Engine on the Controller machine to work as an Interactive User. To do so:
1. Make sure wlrun process is not running on the machine.
2. Select **Run** from the Windows Start menu and type dcomcnfg.
3. Press ENTER to display the list of DCOM applications.
4. In the displayed application list, double-click wlrun.LrEngine.
5. In the **Identity** tab of the dialog box, select **The interactive user.**
6. Click **OK** twice to close both dialog boxes.

Configuring the SAPGUI Monitor

To obtain data from the SAPGUI monitor, you need to indicate which statistics and measurements you want to monitor.

To configure the SAPGUI monitor:

1. Open the Add Monitor window.

   ➤ **Before running a load test:** In the Load Test configuration page's **Monitors** tab, select the type of monitor profile to create:

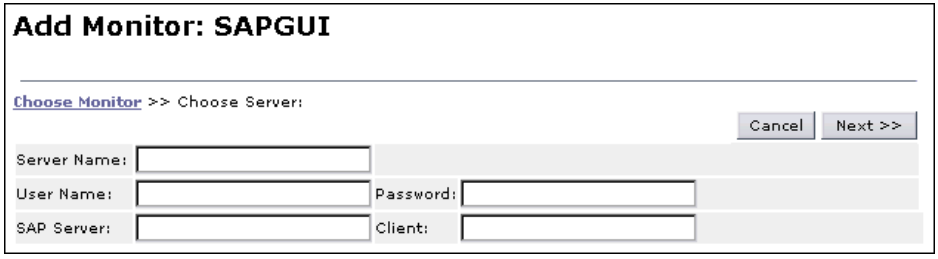
- To create a monitor profile for the current load test, click **Add Local Profile**.
- To create a monitor profile that can be used for other load tests, click **Add Profile**, type a name and description of the profile and click **OK**.

For details, see the section about configuring monitors in the *HP Performance Center User Guide*.

- **During a load test run:** On the Load Test run page:
  - Click the **Design** button. The Load Test Design dialog box opens.
  - Click **Monitors**.

  For details, see the section about creating monitor profiles while running a load test in the *HP Performance Center User Guide*.

2 Under **ERP/CRM Server Resource Graphs**, select **SAPGUI** and click **Next**. The Choose Server page opens.



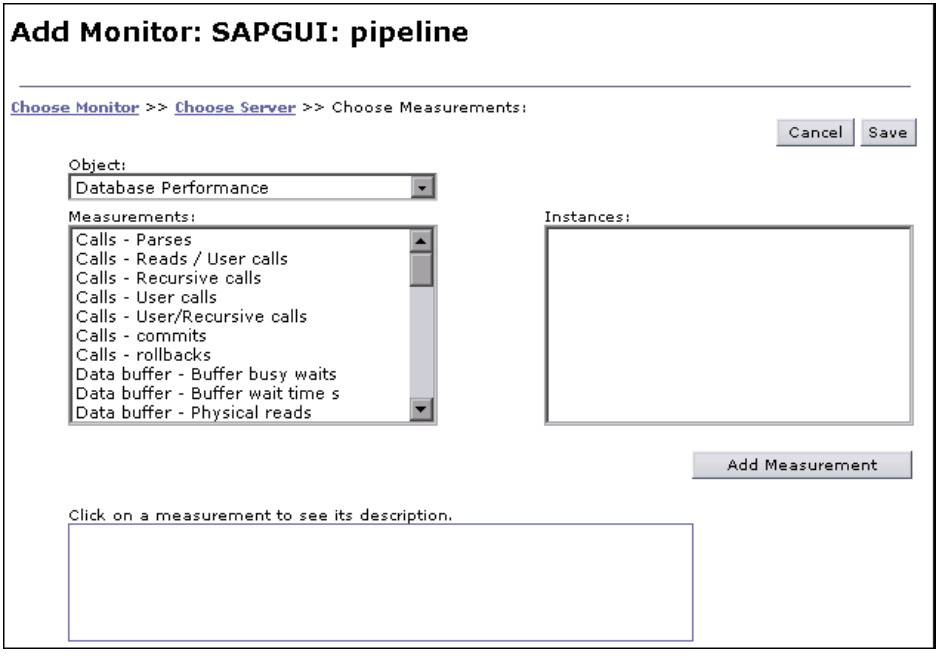
3 Type the following information:

- **Server Name**. Type the name or IP address of the server whose resources you want to monitor.
- **User Name**. Type the login name used to access the SAPGUI server.
- **Password**. Type the password for the login name.
- **SAP Server**. Type the name of the SAPGUI server.
- **Client**. Type the number used in the Client field of the SAP logon details.

You can type the server name in the following ways:

- Type the server description, as it is displayed in the Server Name box.
- Type a string, in the format server_network_name[:system_number], where server_network_name is the name (or IP address) of the application server as it is displayed in the Server Name field (for example: pipeline.mercury.com). If a SAP router string is also specified, the server_network_name should be the concatenation of the router string and the application server.

4 Click **Next**. The Choose Measurements page opens, displaying the available measurements and server properties.



5 For each measurement that you want to monitor, select an object, measurements, and instances. The instances are relevant only if multiple instances of the highlighted measurement are running.

Click **Add Measurement**.

For a list of the available performance counters, see "SAPGUI Performance Counters" on page 365.

6 Click **Save** to add the selected resource measurements to the monitor profile.

SAPGUI Performance Counters

The following table lists 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.

Measurement	Description
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.
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.

42

SAP Portal Monitoring

The SAP Portal monitor displays statistics about the resource usage of a SAP Portal system during the load test run.

This chapter includes:

➤ Configuring the SAP Portal Monitor on page 367

➤ SAP Portal Performance Counters on page 371

Configuring the SAP Portal Monitor

To obtain data on the SAP R/3 system server, you need to enable the SAP online monitor (from the Performance Center Host) before executing the load test, and indicate which statistics and measurements you want to monitor.

Although SiteScope is installed on the Utility Server, we recommend installing it on the same machine as the Controller, or on a dedicated server in order to reduce load on the Utility Server.

To configure the SAP Portal monitor through the User Site:

1 Open the Add Monitor window.

➤ **Before running a load test:** In the Load Test configuration page's **Monitors** tab, select the type of monitor profile to create:

➤ To create a monitor profile for the current load test, click **Add Local Profile**.

➤ To create a monitor profile that can be used for other load tests, click **Add Profile**, type a name and description of the profile and click **OK**.

For details, see the section about configuring monitors in the *HP Performance Center User Guide*.

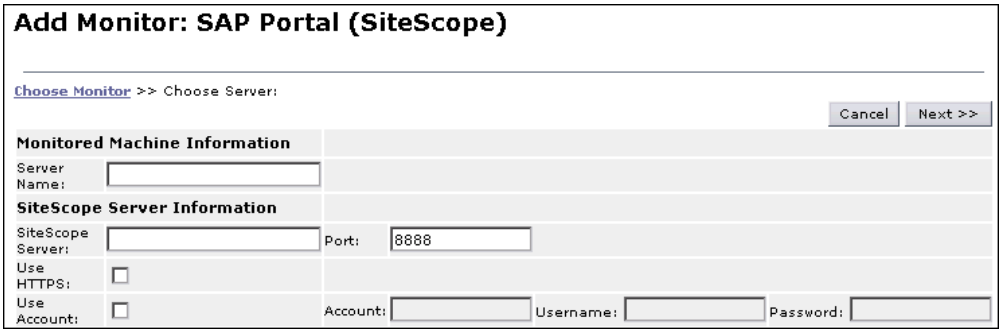
➤ **During a load test run:** On the Load Test run page:

➤ Click the **Design** button. The Load Test Design dialog box opens.

➤ Click **Monitors**.

For details, see the section about creating monitor profiles while running a load test in the *HP Performance Center User Guide*.

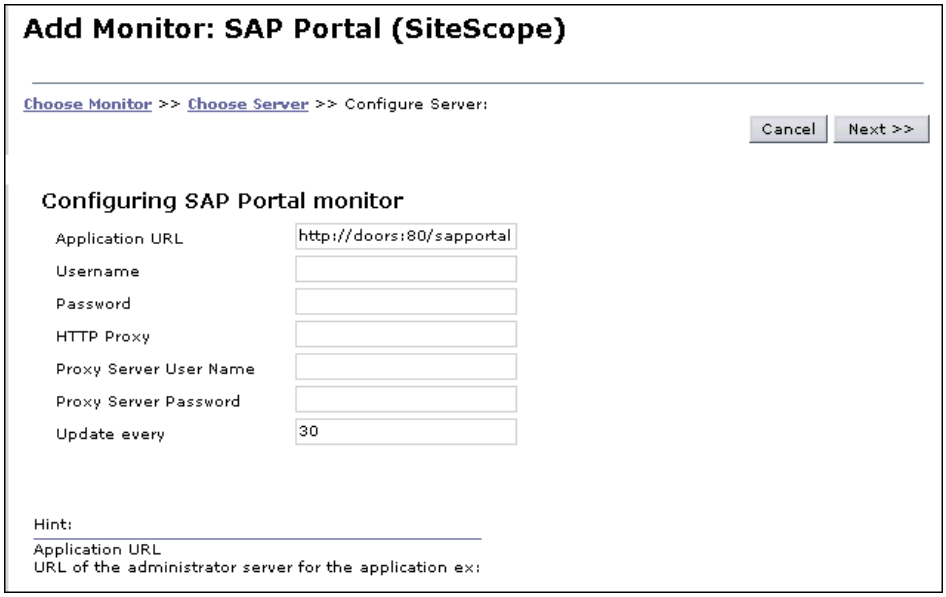
2 Under **ERP/CRM Server Resource Graphs**, select **SAP Portal (SiteScope)** and click **Next**. The Choose Server page opens.



3 Under **Monitored Machine Information**, type the name or IP address of the server whose resources you want to monitor.

Under **SiteScope Server Information**, type the SiteScope machine name and port (default: 8888), and specify if you are using a Secure HTTP connection or a SiteScope user account. If you are using a SiteScope account, fill in the account name or number, and the user name and password defined to log in to the SiteScope account.

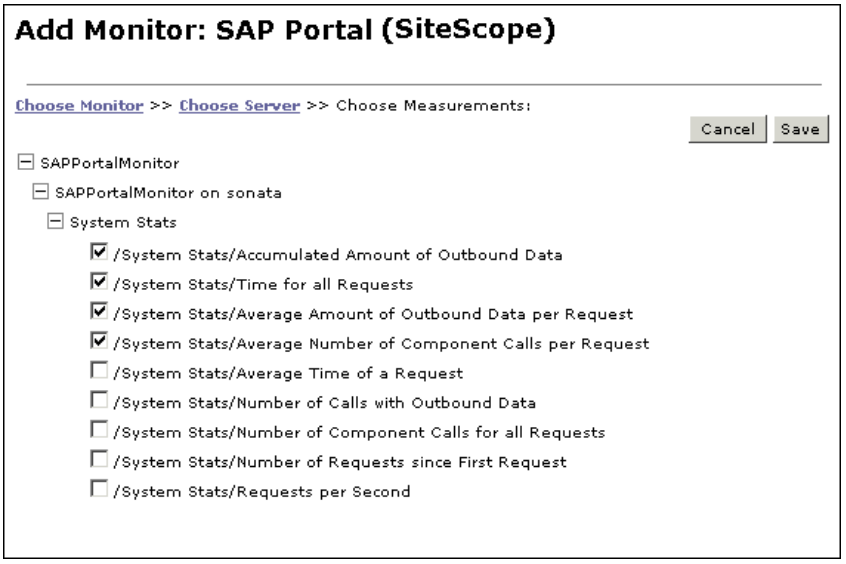
4 Click **Next**. The Configure Server page opens.



5 Type the following SAP Portal monitor information:

- **Application URL**. Type the URL of the administrator server for the application.
- **Username**. Type the user name for the server administrator page.
- **Password**. Type the password for the server administrator page.
- **HTTP Proxy**. Type a proxy server to use, including the port (optional).
- **Proxy Server User Name**. If the proxy server requires authorization, type the user name.
- **Proxy Server Password**. If the proxy server requires authorization, type the password.
- **Update every**. Type the amount of time lapse between the SiteScope check of the monitor. The default is 30 seconds.

6 Click **Next**. The Choose Measurements page opens, displaying the available measurements and server properties.



7 Select the measurements you want to monitor.

For a list of the available performance counters, see "SAP Portal Performance Counters" on page 371.

8 Click **Save** to add the selected resource measurements to the monitor profile.

SAP Portal Performance Counters

The following table shows the default counters that can be measured:

Measurement	Description
Accumulated Amount of Outbound Data (bytes)	The accumulated amount of outbound data, measured in bytes.
Time for all Requests (ms)	The total time, in milliseconds, taken for processing all requests.
Average Amount of Outbound Data per Request (bytes)	The average amount of outbound data per request, measured in bytes.
Average Number of Component Calls per Request (bytes)	The average number of component calls per request, measured in bytes.
Average Time of a Request (ms)	The average amount of time, in milliseconds, taken to process a request.
Number of Calls with Outbound Data	The total number of calls with outbound data.
Number of Component Calls for all Requests	The total number of component calls for all requests.
Number of Requests since First Request	The total number of requests since the first request was made.
Requests per Second	The number of requests made per second.
Time Stamp of First Request	The time stamp of the first request.

43

Siebel Web Server Monitoring

The Siebel Web Server monitor displays statistics about the resource usage of a Siebel Web Server during the load test run.

This chapter includes:

- Setting Up the Monitoring Environment on page 373
- Configuring the Siebel Web Server Monitor on page 374
- Siebel Web Server Performance Counters on page 378

Setting Up the Monitoring Environment

Before monitoring a Siebel Web Server, you must set up the server monitor environment.

To set up the Siebel Web Server monitor environment:

- Check that the Siebel web server plug-in is installed and configured to enable the display of the statistics you want to monitor. This may require that stats page sections be enabled by editing the **eapps.cfg** file for the Siebel server. For more information, see the Siebel documentation.
- Although SiteScope is installed on the Utility Server, we recommend installing it on the same machine as the Controller, or on a dedicated server in order to reduce load on the Utility Server.

➤ Verify that SiteScope is collecting the required data from the servers it is monitoring. From the SiteScope Panel, select the monitor group polling the Siebel Web Server machine, and check that the monitor displays a list of server measurements in the Status column. This may require that stats page sections be enabled by editing the eapps.cfg file for the Siebel server. Consult the Siebel documentation for more information.

Configuring the Siebel Web Server Monitor

To obtain data on the Siebel Web Server, you need to enable the Siebel Web Server online monitor (from the Performance Center Host) before running the load test, and indicate which statistics and measurements you want to monitor.

To configure the Siebel Web Server monitor through the User Site:

1 Open the Add Monitor window.

➤ **Before running a load test:** In the Load Test configuration page's **Monitors** tab, select the type of monitor profile to create:

➤ To create a monitor profile for the current load test, click **Add Local Profile**.

➤ To create a monitor profile that can be used for other load tests, click **Add Profile**, type a name and description of the profile and click **OK**.

For details, see the section about configuring monitors in the *HP Performance Center User Guide*.

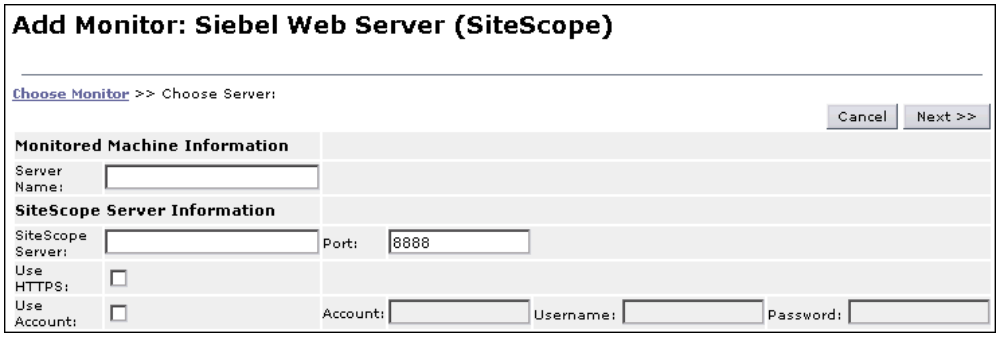
➤ **During a load test run:** On the Load Test run page:

➤ Click the **Design** button. The Load Test Design dialog box opens.

➤ Click **Monitors**.

For details, see the section about creating monitor profiles while running a load test in the *HP Performance Center User Guide*.

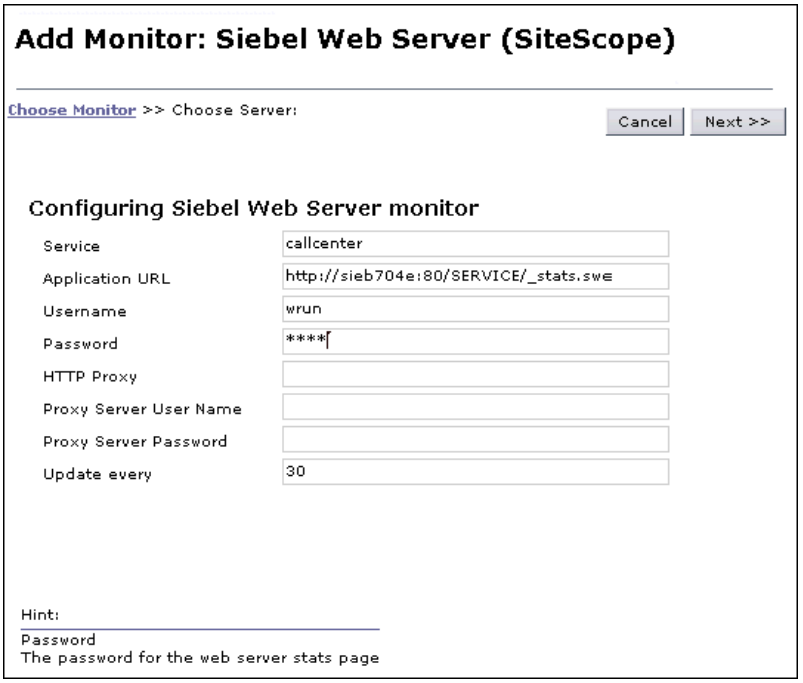
2 Under **ERP/CRM Server Resource Graphs**, select **Siebel Web Server (SiteScope)** and click **Next**. The Choose Server page opens.



3 Under **Monitored Machine Information**, type the name or IP address of the server whose resources you want to monitor.

Under **SiteScope Server Information**, type the SiteScope machine name and port (default: 8888), and specify if you are using a Secure HTTP connection or a SiteScope user account. If you are using a SiteScope account, fill in the account name or number, and the user name and password defined to log in to the SiteScope account.

4 Click **Next**. The Configure Server page opens.

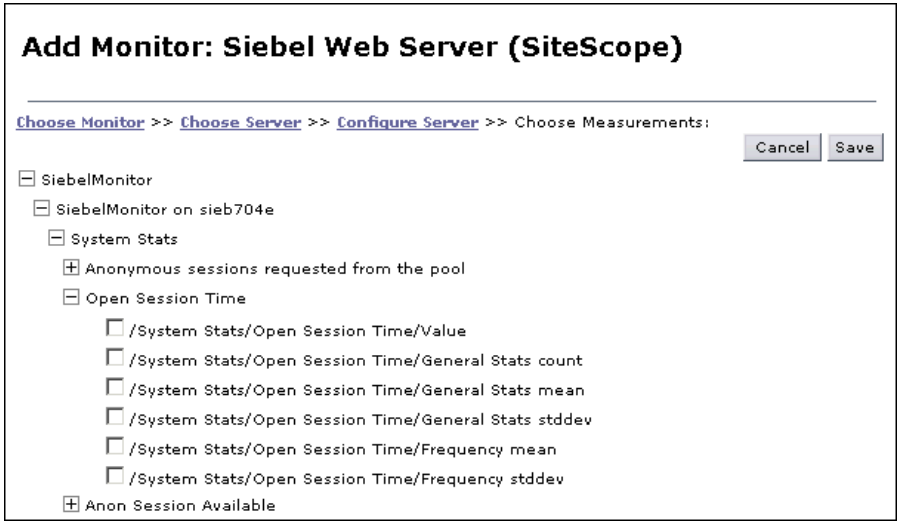


5 Type the following Siebel Web Server monitor information:

- **Service**. Type the name of a valid Siebel virtual directory (for example, callcenter or sales), one whose URL is an entry point defined in the Siebel SWSE configuration file (eapps.cfg).
- **Application URL**. Displays the default URL of the Web plug-in server stats page for the application you want to monitor. For example, **http://siebelsrv/<service>/_stats.swe**. If the Siebel web server is configured to support verbose mode, and you want to include information on "Locks" and "Current Operations Processing," you can add verbose=high, for example: **http://siebelsrv/<service>/_stats.swe?verbose=high**.
- **Username**. Type the user name to access the Web server stats page.
- **Password**. Type the password to access the Web server stats page.

- **HTTP Proxy**. If you are using a proxy to access the Siebel server, type the proxy server to use including the port (for example, proxy.sitescope.com:8080).
- **Proxy Server User Name**. If the proxy server requires authorization, type the proxy user name.
- **Proxy Server Password**. If the proxy server requires authorization, type the proxy password.
- **Update every**. Type how frequently the monitor should check the Database server.

6 Click **Next**. The Choose Measurements page opens, displaying the available measurements and server properties.



7 Select the measurements you want to monitor.

For a list of the available performance counters, see "Siebel Web Server Performance Counters" on page 378.

8 Click **Save** to add the selected resource measurements to the monitor profile.

Siebel Web Server Performance Counters

The following table shows the default counters that can be measured:

Measurement	Description
Anonymous sessions requested from the pool	The number of anonymous sessions requested from the pool.
Open Session Time	The time users experience logging on to the system.
Anon Session Removed	The number of anonymous sessions removed from the pool.
Anon Session Available	The number of anonymous sessions available in the pool.
Anonymous sessions returns to the pool	The number of anonymous sessions returned to the pool.
Response Time	The time taken to respond to a user request.
Close Session Time	The time users experience logging off the system.
Request Time	The time taken to process the user request.

44

Siebel Server Manager Monitoring

The Siebel Server Manager monitor displays statistics about the resource usage of a Siebel Server Manager during the load test run.

This chapter includes:

- Setting up the Monitoring Environment on page 379
- Configuring the Siebel Server Manager Monitor on page 381
- Siebel Server Manager Performance Counters on page 382

Setting up the Monitoring Environment

Before you set up the monitor, perform the following:

- Check that SiteScope (the application that is used to monitor the Siebel Server Manager) has been installed on a server. You can install SiteScope on the same machine as the Controller, or on a dedicated server.
- On the machine where SiteScope is installed, configure SiteScope to monitor the required Siebel Server Manager machine. For more information, see "Configuring the Siebel Server Manager Client on the SiteScope Machine" on page 380.
- Verify that SiteScope is collecting the required data from the servers it is monitoring. From the SiteScope Panel, select the monitor group polling the Siebel Server Manager machine, and check that the monitor displays a list of server measurements in the Status column.

Configuring the Siebel Server Manager Client on the SiteScope Machine

To monitor the Siebel Server Manager performance, you install the Siebel Server Manager client on the SiteScope machine, and configure SiteScope to monitor the required Siebel Server Manager machine.

To configure the Siebel Server Manager monitor on the SiteScope machine:

1. Restart SiteScope after installing the monitor add-in, and verify connectivity to the Siebel SWSE page by opening the following URL from the machine where sitescope is installed:

```
http://<your_siebel_server>/callcenter/_stats.swe
```

2. In the SiteScope main panel, select an existing group or create a new group.
3. Under **Add to Group**, click **Monitor** and select **Siebel Server Manager** from the list of monitors.
4. Click **Choose server**, and type the name of the Siebel Server in the **Application server** field, the Enterprise Server in the **Enterprise server** field, and the Gateway Server in the **Gateway server** field.
5. If necessary, type the user name and password for the Siebel Server Manager client.
6. Type the path to the Siebel Server Manager in the **Path to Script** field.
7. Click **Browse**, select your desired counters, and click **Choose Counters**.
8. In the **Title** field, type a name for the monitor.

Note: When you assign a name to a monitor, include the server name in the monitor name. This avoids any confusion as to which host the monitor belongs. For example, SiebelManager on sieb07.

Do not use "\" in the Title field.

---

9. Click **Add Monitor**.

Configuring the Siebel Server Manager Monitor

After installing and configuring the Siebel Server Manager client on the SiteScope machine, you must select the counters to measure.

To configure the Siebel Server Manager monitor:

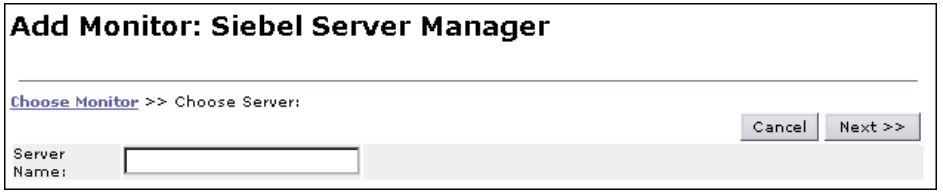
1 Open the Add Monitor window.

- **Before running a load test:** In the Load Test configuration page's **Monitors** tab, select the type of monitor profile to create:
  - To create a monitor profile for the current load test, click **Add Local Profile**.
  - To create a monitor profile that can be used for other load tests, click **Add Profile**, type a name and description of the profile and click **OK**.

  For details, see the section about configuring monitors in the *HP Performance Center User Guide*.
- **During a load test run:** On the Load Test run page:
  - Click the **Design** button. The Load Test Design dialog box opens.
  - Click **Monitors**.

  For details, see the section about creating monitor profiles while running a load test in the *HP Performance Center User Guide*.

2 Under **ERP/CRM Server Resource Graphs**, select **Siebel Server Manager** and click **Next**. The Choose Server page opens.



3 Type the name or IP address of the server whose resources you want to monitor.

4 Click **Next**. The Choose Measurements page opens, displaying the available measurements and server properties.

5 Select the measurements you want to monitor.

For a list of the available performance counters, see "Siebel Server Manager Performance Counters" on page 382.

6 Click **Save** to add the selected resource measurements to the monitor profile.

Limitations

The SiteScope monitor has the following limitations when it retrieves information from a Siebel Server Manager:

- The component counters (for example, Average SQL Time for <component>) are updated with the aggregated Siebel data only at the end of a user session.
- The SiteScope monitor consumes very high CPU resources (approximately 40%).

Siebel Server Manager Performance Counters

The following table shows the default counters that can be measured:

Measurement	Description
Average Connect Time	The average connection time.
Average Reply Size	The average size of a user reply.
Average Request Size	The average size of a user request.
Average Requests Per Session	The average number of user requests per session.
Average Response Time	The average amount of time that it takes the server to respond to a request.
Average Think Time	The average amount of think time taken to respond to a request.
Avg SQL Execute Time	The average SQL execute time.
Avg SQL Fetch Time	The average SQL fetch time.

Measurement	Description
Avg SQL Parse Time	The average SQL parse time.
CPU Time	The CPU time used in the work process.
Elapsed Time	The total amount of elapsed time.
Num of DBConn Retries	The number of database connection retries.
Num of DLRbk Retries	The number of DLRbk retries.
Num of Exhausted Retries	The total number of retries that expired.
Number of SQL Executes	The total number of SQL executes.
Number of SQL Fetches	The total number of SQL fetches.
Number of SQL Parses	The total number of SQL parses.
Number of Sleeps	The number of sleeps.
Object Manager Errors	The total number of object manager errors.
Reply Messages	The total number of reply messages.
Request Messages	The total number of request messages.
SQL Execute Time	The total SQL execute time.
SQL Fetch Time	The total SQL fetch time.
SQL Parse Time	The total SQL parse time.
Sleep Time	The total sleep time.
Tests Attempted	The number of tests attempted.
Tests Failed	The number of tests that failed.
Tests Successful	The number of tests that were successful.
Total Reply Size	The total reply size, measured in bytes.
Total Request Size	The total request size, measured in bytes.
Total Response Time	The total response time.

Measurement	Description
Total Tasks	The total number of tasks.
Total Think Time	The total think time.

45

PeopleSoft (Tuxedo) Monitoring

The PeopleSoft (Tuxedo) monitor displays statistics about the resource usage of a PeopleSoft (Tuxedo) server during the load test run.

This chapter includes:

- Setting Up the PeopleSoft (Tuxedo) Monitor on page 385
- Configuring the PeopleSoft (Tuxedo) Monitor on page 386
- PeopleSoft (Tuxedo) Performance Counters on page 389

Setting Up the PeopleSoft (Tuxedo) Monitor

If Tuxedo 7.1 or later is installed, you can monitor more than one PeopleSoft (Tuxedo) application server at a time. If Tuxedo 6.5 or earlier is installed, you can monitor only one PeopleSoft (Tuxedo) application server at a time.

Before you set up the monitor, perform the following:

- Ensure that a Tuxedo workstation client (not a native client), version 6.3 or later, is installed on the Controller machine. Use a Tuxedo 6.x client if a Tuxedo 6.x server is used, and Tuxedo 7.1 or later client if a Tuxedo 7.1 or later server is used. If you use a Tuxedo 6.5 or earlier server, you can still use a Tuxedo 7.1 or later client in order to monitor it, provided that you set the WSINTOPPRE71 environment variable to “yes”.

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.

- Define the Tuxedo environment variables on the Performance Center Host machine—set the TUXDIR variable to the Tuxedo installation directory, (for example, V:\environ\32\Tuxedo8.0), and add the Tuxedo **bin** directory to the PATH variable.
- Ensure that the workstation listener (WSL) process is running. This enables the application server to accept requests from workstation clients. The address and port number used to connect to the application server must match those dedicated to the WSL process.

Note: For information on configuring the WSL, see the BEA Tuxedo Web site (http://edocs.beasys.com/tuxedo/tux81/rf5/rf5101.htm#1534543).

Configuring the PeopleSoft (Tuxedo) Monitor

To obtain data for this graph, you need to enable the PeopleSoft (Tuxedo) online monitor (from the Performance Center Host) and select the default measurements you want to display, before running the load test.

To configure the PeopleSoft (Tuxedo) monitor through the User Site:

1 Open the Add Monitor window.

- **Before running a load test:** In the Load Test configuration page's **Monitors** tab, select the type of monitor profile to create:
  - To create a monitor profile for the current load test, click **Add Local Profile**.

➤ To create a monitor profile that can be used for other load tests, click **Add Profile**, type a name and description of the profile and click **OK**.

For details, see the section about configuring monitors in the *HP Performance Center User Guide*.

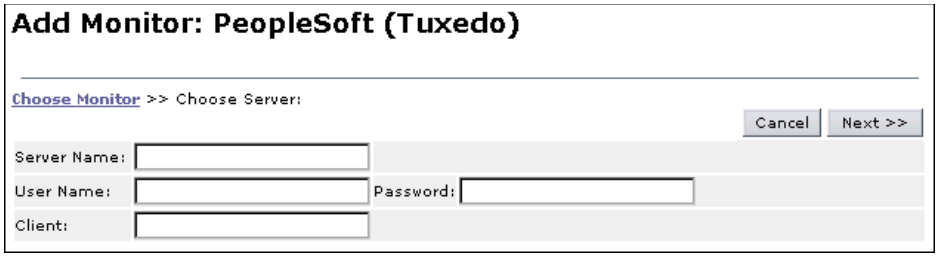
➤ **During a load test run:** On the Load Test run page:

➤ Click the **Design** button. The Load Test Design dialog box opens.

➤ Click **Monitors**.

For details, see the section about creating monitor profiles while running a load test in the *HP Performance Center User Guide*.

2 Under **ERP/CRM Server Resource Graphs**, select **PeopleSoft (Tuxedo)** and click **Next**. The Choose Server page opens.



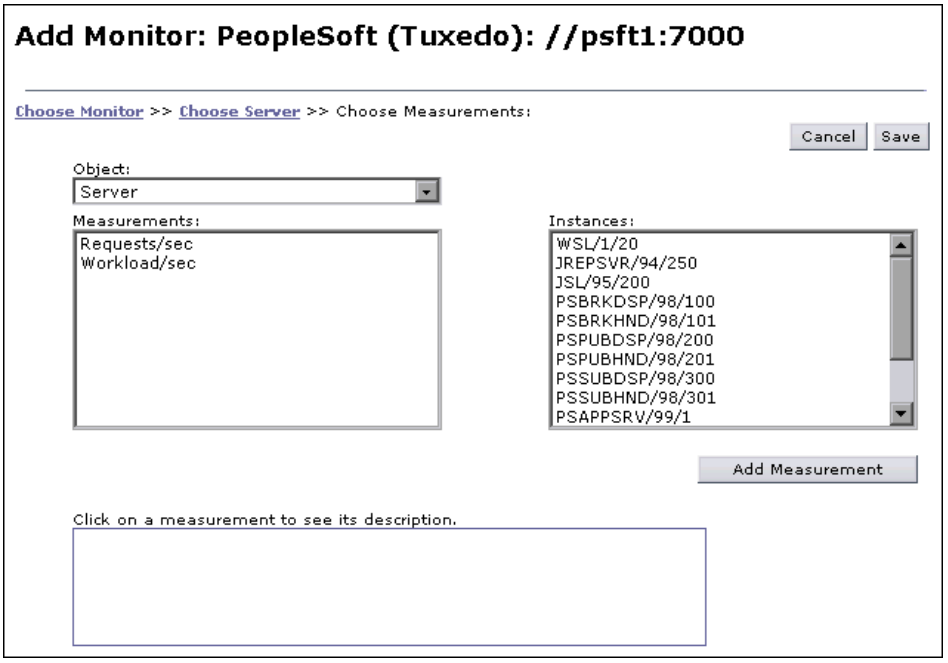
3 Type the name or IP address of the server whose resources you want to monitor.

Note: If you are using multiple instances of the Tuxedo monitor on the same machine, then type the port number of each Tuxedo monitor in order to distinguish one instance from another. The entry should have the following format: <machine name>:<port number>.

Type the user name and password of the monitored server, and the name of the client machine.

4 Click **Next**. The Choose Measurements page opens, displaying the available measurements and server properties.

5 Select the measurements you want to monitor.



6 For each measurement that you want to monitor, select an object, measurements, and instances. The instances are relevant only if multiple instances of the highlighted measurement are running.

Click **Add Measurement**.

7 Click **Save** to add the selected resource measurements to the monitor profile.

For a list of the available performance counters, see “PeopleSoft (Tuxedo) Performance Counters” on page 389.

8 Click **Save** to add the selected resource measurements to the monitor profile.

PeopleSoft (Tuxedo) Performance Counters

The following table describes the default counters that can be measured. Pay particular attention to the following measurements: % Busy Clients, Active Clients, Busy Clients, Idle Clients, and all the queue counters for the APPQ/PSAPPSRV queue.

Monitor	Measurements
Machine	**% Busy Clients**. The per cent of active clients currently logged in to the Tuxedo application server which are waiting for a response from the application server.
	Active Clients. The total number of active clients currently logged in to the Tuxedo application server.
	Busy Clients. The total number of active clients currently logged in to the Tuxedo application server that are waiting for a response from the application server.
	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 Transactions. Number of in use transaction table entries on this machine.
	Idle Clients. The total number of active clients currently logged in to the Tuxedo application server that are not waiting for a response from the application server.
	Workload Completed/second. The total workload on all the servers for the machine that was completed, per unit time.
	Workload Initiated/second. The total workload on all the servers for the machine that was initiated, per unit time.

Monitor	Measurements
Queue	**% Busy Servers**. The per cent of active servers currently handling Tuxedo requests.
	Active Servers. The total number of active servers either handling or waiting to handle Tuxedo requests.
	Busy Servers. The total number of active servers currently busy handling Tuxedo requests.
	Idle Servers. The total number of active servers currently waiting to handle Tuxedo requests.
	Number Queued. The total number of messages which have been placed on the queue.
Server	**Requests/second**. How many server requests were handled per second.
	Workload/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.
Workstation Handler (WSH)	**Bytes Received/sec**. The total number of bytes received by the workstation handler, per second.
	Bytes Sent/sec. The total number of bytes sent back to the clients by the workstation handler, per second.
	Messages Received/sec. The number of messages received by the workstation handler, per second.
	Messages Sent/sec. The number of messages sent back to the clients by the workstation handler, per second.
	Number of Queue Blocks/sec. The number of times the queue for the workstation handler blocked, per second. This gives an idea of how often the workstation handler was overloaded.

Part XIII

Application Component Monitoring

46

Introducing Application Component Monitoring

Using Performance Center's Application Component monitors, you can monitor the Microsoft COM+ server.

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 load test.

The procedures for selecting monitor measurements and configuring the monitors vary according to server type.

47

Microsoft COM+ Monitoring

The Microsoft COM+ monitor provides information about the resource usage of the Microsoft COM+ server during load test runs.

This chapter includes:

- Setting Up the Monitoring Environment on page 395
- Configuring the Microsoft COM+ Server Monitor on page 396
- Microsoft COM+ Performance Counters on page 398

Setting Up the Monitoring Environment

To monitor the Microsoft COM+ server performance, you must first set up the Microsoft COM+ server on the server machine.

- Before configuring the monitor, on the Microsoft COM+ server machine, run the **<ComPlus_server_side.exe>** file to configure the server for monitoring.
- Before running the Microsoft COM+ monitor over a firewall:
  - Make sure that the MI Listener is installed on any machine outside of the firewall. For installation instructions, see the *HP Performance Center System Configuration and Installation Guide*.
  - Configure the Performance Center agent on the server machine. For more information, see the *HP Performance Center System Configuration and Installation Guide*.
  - Configure the monitor as described in "Configuring the Microsoft COM+ Server Monitor" below.

Configuring the Microsoft COM+ Server Monitor

After you have set up the Microsoft COM+ server on the server machine, you can specify which measurements and resources you want the Microsoft COM+ monitor to measure. You select these counters using the Performance Center Host's monitor configuration page.

To configure the COM+ monitor:

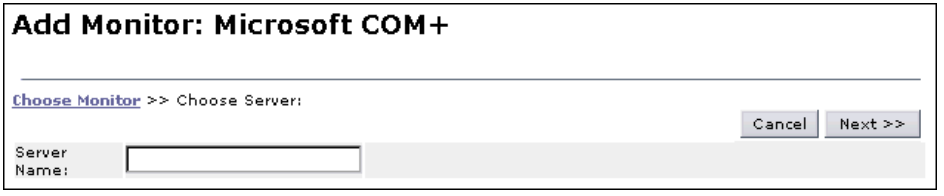
1 Open the Add Monitor window.

- **Before running a load test:** In the Load Test configuration page's **Monitors** tab, select the type of monitor profile to create:
  - To create a monitor profile for the current load test, click **Add Local Profile**.
  - To create a monitor profile that can be used for other load tests, click **Add Profile**, type a name and description of the profile and click **OK**.

  For details, see the section about configuring monitors in the *HP Performance Center User Guide*.
- **During a load test run:** On the Load Test run page:
  - Click the **Design** button. The Load Test Design dialog box opens.
  - Click **Monitors**.

  For details, see the section about creating monitor profiles while running a load test in the *HP Performance Center User Guide*.

2 Under **Application Component Graphs**, select **Microsoft COM +** and click **Next**. The Choose Server page opens.



3 Type the monitor information:

- For regular monitoring without a firewall, type the name or IP address of the server whose resources you want to monitor.
- To connect to the monitor over a firewall, type the server name or IP address of the machine you want to monitor, according to the following format:

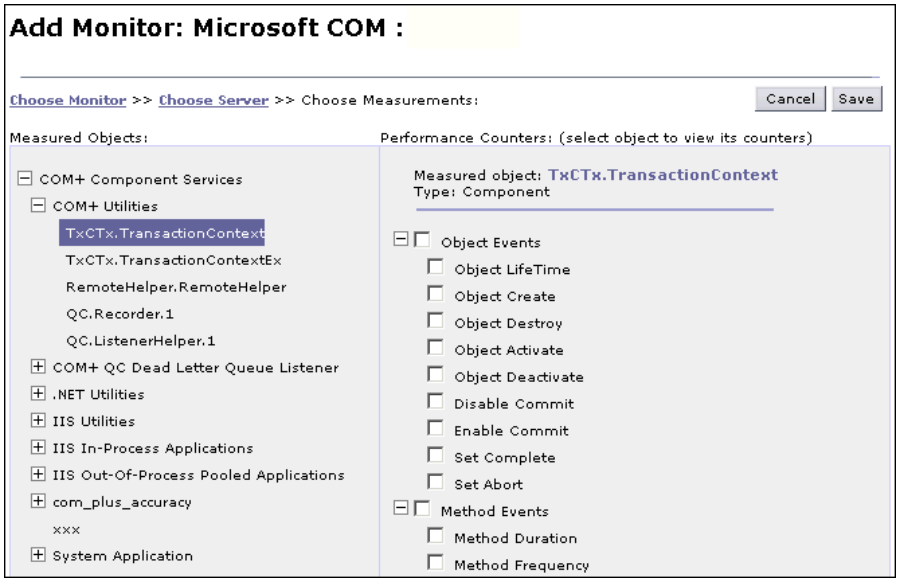
```
<MI Listener machine>:<server machine key>
```

where server machine key is the unique key that you chose when configuring the firewall Agent on the server machine.

For example:

```
12.12.12.3:serverid
```

4 Click **Next**. The Choose Measurements page opens, displaying the available measurements and server properties.



5 Select the objects, counters, and instances that you want to monitor. The instances are relevant only if multiple instances of the highlighted counter are running.

For a list of available performance counters, see "Microsoft COM+ Performance Counters" on page 398.

6 Click **Save** to add the selected resource measurements to the monitor profile.

Microsoft COM+ Performance Counters

The following tables describe the default counters that can be measured:

Authentication Metrics

Measurement	Description
Authenticate	Frequency of successful method call level authentication. When you set an authentication level for an application, you determine what degree of authentication is performed when clients call into the application.
Authenticate Failed	Frequency of failed method call level authentication.

Application Events

Measurement	Description
Activation	Frequency of application activation or startup.
Shutdown	Frequency of application shutdown or termination.

Thread Events

Measurement	Description
Thread Start	Rate at which single-threaded apartment (STA) thread for application have been started.
Thread Terminate	Rate at which single-threaded apartment (STA) thread for application have been terminated.
Work Enque	Event sent if a work is queued in single thread apartment object (STA). Note: These events are not signaled/sent in Windows Server 2003 and later.
Work Reject	Event sent if a work is rejected from single thread apartment object (STA). Note: These events are not signaled/sent in Windows Server 2003 and later.

Transaction Events

Measurement	Description
Transaction Duration	Duration of COM+ transactions for selected application.
Transaction Start	Rate at which transactions have started.
Transaction Prepared	Rate at which transactions have completed the prepare phase of the two-phase protocol.
Transaction Aborted	Rate at which transactions have been aborted.
Transaction Commit	Rate at which transactions have completed the commit protocol.

Object Events

Measurement	Description
Object Life Time	Duration of object existence (from instantiation to destruction).
Object Create	Rate at which new instances of this object are created.
Object Destroy	Rate at which instances of the object are destroyed.
Object Activate	Rate of retrieving instances of a new JIT-activated object.
Object Deactivation	Rate of freeing JIT-activated object via SetComplete or SetAbort.
Disable Commit	Rate of client calls to DisableCommit on a context. DisableCommit declares that the object's transactional updates are inconsistent and can't be committed in their present state.
Enable Commit	Rate of client calls to EnableCommit on a context. EnableCommit declares that the current object's work is not necessarily finished, but that its transactional updates are consistent and could be committed in their present form.
Set Complete	Rate of client calls to SetComplete on a context. SetComplete declares that the transaction in which the object is executing can be committed, and that the object should be deactivated on returning from the currently executing method call.
Set Abort	Rate of client calls to SetAbort on a context. SetAbort declares that the transaction in which the object is executing must be aborted, and that the object should be deactivated on returning from the currently executing method call.

Method Events

Measurement	Description
Method Duration	Average duration of method.
Method Frequency	Frequency of method invocation.
Method Failed	Frequency of failed methods (i.e. methods that return error HRESULT codes).
Method Exceptions	Frequency of exceptions thrown by selected method.

Part XIV

Middleware Performance Monitoring

48

Middleware Performance Monitoring

Using Performance Center's Middleware Performance Tuxedo monitor, you can monitor Tuxedo servers during a load test run and isolate server performance bottlenecks. You can also monitor the performance of IBM WebSphere MQ Series servers using the SiteScope monitor.

This chapter includes:

- Setting Up the Monitoring Environment on page 406
- Configuring the Tuxedo Monitor on page 407
- Tuxedo Monitor Measurements on page 410

Note: For information about monitoring the IBM WebSphere MQ Series servers using SiteScope, see "Configuring the SiteScope Monitor" on page 92.

Setting Up the Monitoring Environment

The Tuxedo monitor allows you to measure and view the performance of your Tuxedo application server.

If Tuxedo 7.1 or later is installed, you can monitor more than one Tuxedo application server at a time. However, if Tuxedo 6.5 or earlier is installed on the machine, you can monitor only one Tuxedo application server at a time.

Note: The Tuxedo workstation client should be installed on local drive. You cannot install the client on the mapped network drive.

Before you configure the monitor, perform the following:

➤ Check that a Tuxedo workstation client (not a native client) is installed on the Performance Center Host and on the Utility Server. Use a Tuxedo 6.x client if a Tuxedo 6.x server is used, and Tuxedo 7.1 or later client if a Tuxedo 7.1 or later server is used. If you use a Tuxedo 6.5 or earlier server, you can still use a Tuxedo 7.1 or later client to monitor it, provided that you set the WSINTOPPRE71 environment variable to **yes**.

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.

➤ Define the Tuxedo environment variables on both the Performance Center Host and Utility Server—set the TUXDIR variable to the Tuxedo installation directory, (for example, V:\environ\32\Tuxedo8.0), and add the Tuxedo **bin** directory to the PATH variable.

➤ Check that the workstation listener (WSL) process is running. This enables the application server to accept requests from workstation clients. The address and port number used to connect to the application server must match those dedicated to the WSL process.

Note: For information on configuring the WSL, see the BEA Tuxedo Web site (http://edocs.beasys.com/tuxedo/tux81/rf5/rf5101.htm#1534543).

Configuring the Tuxedo Monitor

After you install the client add-in on the Performance Center Host, you specify which measurements and resources you want the Tuxedo monitor to measure.

To configure the Tuxedo monitor:

1 Open the Add Monitor window.

➤ **Before running a load test:** In the Load Test configuration page's **Monitors** tab, select the type of monitor profile to create:

➤ To create a monitor profile for the current load test, click **Add Local Profile**.

➤ To create a monitor profile that can be used for other load tests, click **Add Profile**, type a name and description of the profile and click **OK**.

For details, see the section about configuring monitors in the *HP Performance Center User Guide*.

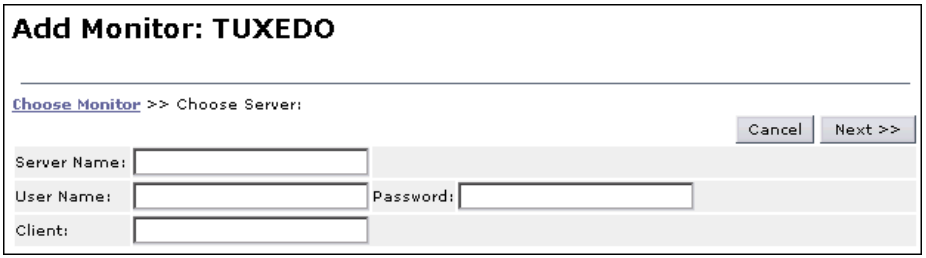
➤ **During a load test run:** On the Load Test run page:

➤ Click the **Design** button. The Load Test Design dialog box opens.

➤ Click **Monitors**.

For details, see the section about creating monitor profiles while running a load test in the *HP Performance Center User Guide*.

2 Under **Middleware Performance Graphs**, select **TUXEDO** and click **Next**. The Choose Server page opens.



3 Type the details of the server whose resources you want to monitor:

a Type the name or IP address of the server whose resources you want to monitor.

Note: If you are using multiple instances of the Tuxedo monitor on the same machine, then type the port number of each Tuxedo monitor in order to distinguish one instance from another. The entry should have the following format: <machine name>:<port number>.

---

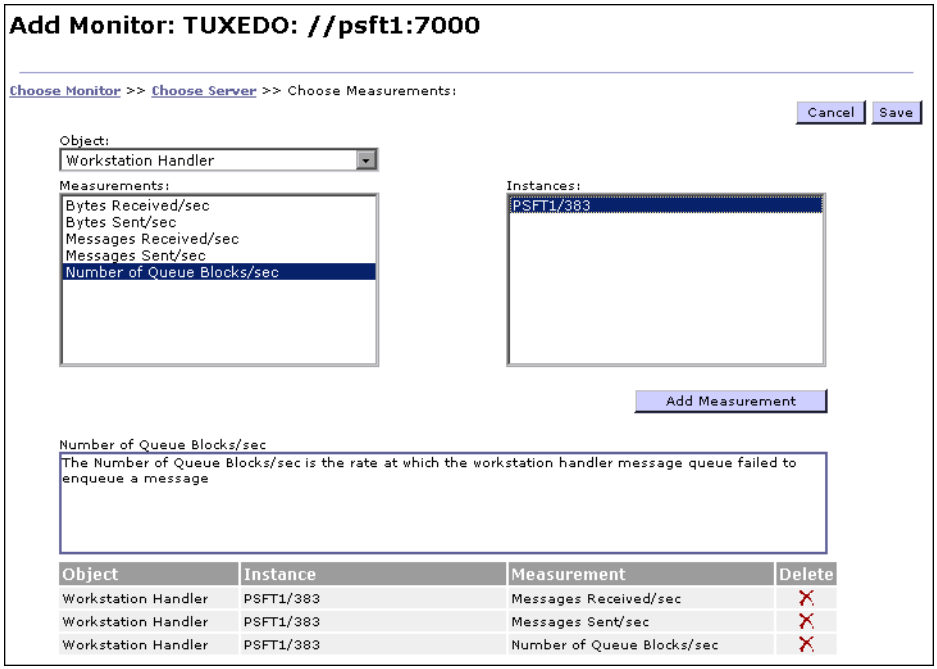
b Type the Tuxedo User Name, Password, and Client name.

Logon information is located in the Logon section of the **tpinit.ini** file in the recorded script's directory. Rather than type the values manually, use the **Browse** button and select the **tpinit.ini** file from a recorded script. You can also determine the client name from the **lrt_tpinitialize** statement in the recorded script.

In the following example of a **tpinit.ini** file, the Tuxedo monitor was configured for a server named psft1 using port 7000, and a client named bankapp. The logon user name was PS and the password was PS.

```
[Logon]
LogonServername=//psft1:7000
LogonUsrName=PS
LogonCltName=bankapp
LogonGrpName=
LogonPasswd=PS
LogonData=
```

4 Click **Next**. The Choose Measurements page opens, displaying the available measurements and server properties.



5 Select the measurements you want to monitor.

6 Click **Save** to add the selected resource measurements to the monitor profile.

Tuxedo Monitor Measurements

The following table describes the default counters that can be measured. Pay particular attention to the following measurements: **% Busy Clients**, **Active Clients**, **Busy Clients**, **Idle Clients**, and all the queue counters for relevant queues.

Monitor	Measurements	Measurements
Machine	**% Busy Clients**	The per cent of active clients currently logged in to the Tuxedo application server which are waiting for a response from the application server.
	Active Clients	The total number of active clients currently logged in to the Tuxedo application server.
	Busy Clients	The total number of active clients currently logged in to the Tuxedo application server that are waiting for a response from the application server.
	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 Transactions	Number of in use transaction table entries on this machine.
	Idle Clients	The total number of active clients currently logged in to the Tuxedo application server that are not waiting for a response from the application server.
	Workload Completed/ second	The total workload on all the servers for the machine that was completed, per unit time.
	Workload Initiated/ second	The total workload on all the servers for the machine that was initiated, per unit time.

Monitor	Measurements	Measurements
Queue	**% Busy Servers**	The per cent of active servers currently handling Tuxedo requests.
	Active Servers	The total number of active servers either handling or waiting to handle Tuxedo requests.
	Busy Servers	The total number of active servers currently busy handling Tuxedo requests.
	Idle Servers	The total number of active servers currently waiting to handle Tuxedo requests.
	Number Queued	The total number of messages which have been placed on the queue.
Server	**Requests/second**	How many server requests were handled per second.
	Workload/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.
Workstation Handler (WSH)	**Bytes Received/sec**	The total number of bytes received by the workstation handler, per second.
	Bytes Sent/sec	The total number of bytes sent back to the clients by the workstation handler, per second.
	Messages Received/sec	The number of messages received by the workstation handler, per second.
	Messages Sent/sec	The number of messages sent back to the clients by the workstation handler, per second.
	Number of Queue Blocks/sec	The number of times the queue for the workstation handler blocked, per second. This gives an idea of how often the workstation handler was overloaded.

Part XV

Troubleshooting Monitors

49

Troubleshooting Monitors

This chapter provides information on troubleshooting problems related to Performance Center monitors.

This chapter includes:

- Cannot Create a Monitor on page 415
- Cannot Save Measurements on page 417
- Port Required for J2EE Monitor Is Already Taken on page 418
- J2EE Monitor Initialization Errors on page 419
- Cannot Save SiteScope Monitors with Many Counters on page 419

Cannot Create a Monitor

Problem description: You are unable to create a specific monitor.

Troubleshooting

1 Make sure that the user trying to access the machine to monitor exists on this machine and has proper privileges. Usually **IUSR_METRO** will be the user attempting to monitor.

For example, if you add the **Windows Resources** monitor, the user **IUSR_METRO** must exist on the machine to be monitored and must have proper access rights.

2. Some monitors require certain clients to reside on the Utility Server and the Controller.
   - For BroadVision, JDK must be installed.
   - For WebLogic, JDK and the **weblogic.jar** file must be present.
   - For Oracle, the Oracle Client must be installed.
   - For DB2, the DB2 administrative client must be installed. (OS390 is not supported.)
3. Make sure the agent required for a particular monitor is running on the machine that you want to monitor.
4. For COM+, make sure that the COM+ probe is running on the machine that you want to monitor.
5. Some monitors require access to a particular URL, from both the Performance Center utility server and the Controller, for monitoring purposes.
   - For Apache, this URL is: http://<server>:<port>/server-status:auto
   - For Websphere 4.x & 5.x, this URL is: http://<server>:<port>/wasPerftool/ervlet/perfservlet

Cannot Save Measurements

Problem description: During configuration of a monitor, after you select the measurements that you want to monitor and you click **Save**, instead of saving the monitor, Performance Center goes back to the Choose Monitor page.

Troubleshooting

This problem usually occurs when you select a large number of measurements. To resolve this issue, you need to increase the value of the **AspMaxRequestEntityAllowed** property in the IIS metabase. This property specifies the maximum number of bytes allowed in the entity body of an ASP request.

Note: For more information about the **AspMaxRequestEntityAllowed** property, see the Microsoft Terminal Services documentation: http://www.microsoft.com/technet/prodtechnol/WindowsServer2003/Library/IIS/a6401b5e-c902-4035-90aa-ee46c270d357.mspx?mfr=true, or http://support.microsoft.com/kb/327659

To increase the value of the AspMaxRequestEntityAllowed property:

1 Run the following command from the command line:

```
cd <IIS_installation_drive>:\inetpub\adminscripts.
```

where **<IIS_installation_drive>** is the hard disk where IIS is installed.

2 To view the current value of the **AspMaxRequestEntityAllowed** property, run the following command:

```
cscript adsutil.vbs get w3svc/ASPMaxRequestEntityAllowed.
```

3 To increase the value of the **AspMaxRequestEntityAllowed** property, run the following command:

```
cscript adsutil.vbs set w3svc/ASPMaxRequestEntityAllowed <maximum value>.
```

where **<maximum value>** is the maximum value, in bytes, that you want to set for the **AspMaxRequestEntityAllowed** property.

4 Run the following command:

```
iisreset.
```

5 To verify that the value of the **AspMaxRequestEntityAllowed** property has changed, run the following command again:

```
cscript adsutil.vbs get w3svc/ASPMaxRequestEntityAllowed.
```

Note: If you are still not able to save the measurements in the monitor profile, increase the value of the **AspMaxRequestEntityAllowed** property further.

For more information about this issue, see the Microsoft support Web site: http://support.microsoft.com/kb/327659

Port Required for J2EE Monitor Is Already Taken

Problem description: The J2EE monitor communicates with Performance Center, by default, using port 2004 and this port is already taken.

Troubleshooting

Select another port as follows:

1 On the application server machine, open the **<J2EE Monitor installation directory>\dat\monitor.properties** file and change the port number specified for the **webserver.monitor.port** property.

2 On the Performance Center host machine, open the **<Performance Center installation directory>\dat\monitors\xmlmonitorshared.ini** file and change the port number specified in **mon_j2ee** section under the **DefaultPort** key.

J2EE Monitor Initialization Errors

Problem description: You are receiving application server initialization errors, such as **UnsupportedClassVersionError**, **NoSuchMethodError**, or **NoClassDefFoundError**.

Troubleshooting

There might be a conflict between the JDK version specified using the J2EE Monitor Initializer, and the actual JDK version used in the application server launch.

Make sure that you selected the correct JDK that is currently being used by the application server. Note that if you switched the application server to work with a different JDK, you must run the J2EE Monitor Initializer again.

Cannot Save SiteScope Monitors with Many Counters

Problem description: You get an error when trying to save a SiteScope monitor profile where the monitor has many counters.

Troubleshooting

This error occurs when there are more than 160 counters on the SiteScope monitor. The error occurs because the value of the IIS parameter, **ASPMaxRequestEntityAllowed**, which defines the maximum size of the Asp page body section, is not high enough. To solve this error, increase the value of the parameter.

To increase the value of the IIS parameter:

1 Log in to the User Site server machine.

2 On the Command line, change the directory to the IIS installation directory, **C:\Inetpub>\AdminScripts**.

3 Run the following commands from the user server and from the utility

a Check the current parameter value.

```
cscript adsutil.vbs get w3svc/ASPMaxRequestEntityAllowed
```

b Set a higher value for the parameter:

```
cscript adsutil.vbs set w3svc/ASPMaxRequestEntityAllowed 8000000
```

c Check that the parameter value was updated.

```
cscript adsutil.vbs get w3svc/ASPMaxRequestEntityAllowed
```

Index

A

Adobe Reader 16
Antara FlameThrower monitor 61
    performance counters 63
Apache monitor 135
    performance counters 137
Application Component monitors 391, 393
    Microsoft COM+ monitor 396
Application Deployment Solutions monitors 345
    Citrix MetaFrame XP 347
Ariba monitor 161
    performance counters 164
ASP monitor 209
    performance counters 212
ATG Dynamo monitor 167
    performance monitors 170
Auto Discovery mode 328

B

BEA WebLogic monitor 231
    performance counters 235
BroadVision monitor 174
    performance counters 179
    servers and services 177

C

Check Point FireWall-1 monitor 125
    performance counters 127
Citrix MetaFrame XP monitor 347
    performance counters 350
ColdFusion monitor 188
    performance counters 190
Connections graph 55
Connections per Second graph 55
custom queries, Oracle monitor 307

D

Data Points graph (online) 41
Database Server Resource monitors 271, 297
    DB2 monitor 281
    SQL Server monitor 273
    Sybase monitor 311
DB2 monitor 281
    performance counters 282
documentation updates 19
documentation, online 16

E

ERP/CRM Server Resource monitors 359
    PeopleSoft (Tuxedo) monitor 386
    SAP Portal monitor 367
    SAPGUI monitor 362
    Siebel monitor 374
    Siebel Server Manager monitor 381
Error Statistics graph 40

F

Firewall Server monitors 125
firewalls
    network monitoring 121

H

Hits per Second graph 50
HP Software Support Web site 18
HP Software Web site 18
HTTP Response per Second graph 51

I

IIS monitor 139
iPlanet (NAS) monitor 196
  performance counters 198
iPlanet (SNMP) monitor 149
  performance counters 151
iPlanet/Netscape monitor 144
  performance counters 147

J

J2EE
  EJB information retrieval 328
  examples of modifying application server configurations 332
  JDBC information retrieval 327
J2EE monitor 323, 329
  activating on client machine 328
  initial configuration settings 326
  installing on application server 324
  performance counters 331
Java Performance monitors
  J2EE monitor 323

K

Knowledge Base 18

M

Microsoft ASP monitor 209
  performance counters 212
Microsoft COM+ monitor 396
  performance counters 398
Microsoft IIS monitor 139
  performance counters 142
Middleware Performance monitors
  Tuxedo monitor 407
monitoring using SiteScope 59
monitors
  Application Component 391, 393
  Application Deployment Solutions 345
  Database Server Resources 271, 297
  ERP/CRM Server Resources 359
  Firewall Server 125
  Java Performance 323
  Run-Time 39
  System Resources 59, 61, 75, 83, 91, 97, 103
  Transaction 43
  Web Application Server Resources 159, 161, 209
  Web Resources 49
  Web server resources 131

N

Network Delay monitor 113
Network monitor 111
  configuring 117
  determining bottlenecks 111
  monitoring over a firewall 121
  on UNIX 113

O

online documentation 16
online graphs
  Connections 55
  Connections per Second 55
  Error Statistics 40
  Hits per Second 50
  HTTP Response per Second 51
  Pages Downloaded per Second 53
  Retries per Second 54
  Throughput 50
  Total Transactions per Second (Passed) 44
  Transaction 43
  User-Defined Data Points 41
online monitors
  Network 111
  Network Delay 113
online resources 18
online transaction monitoring
  adding transactions 45
  graphs 43
  setup 44
online Web server resource monitoring, using a proxy server 132

Oracle Database Resource monitor 302
custom queries 307
performance counters 306
Oracle9iAS 219
Oracle9iAS HTTP monitor 213
performance counters 219

P

packets 111
Pages Downloaded per Second graph 53
PeopleSoft (Tuxedo) monitor 386
performance counters 389
proxy server 132

R

Retries per Second graph 54
rsh, connection for UNIX Network monitor 115
rstatd process
activating 104
resource monitors 104
Running Vusers graph
Error state 40
Finished state 40
Ready state 40
Running state 40
Run-Time graphs 39

S

SAP CCMS resources, monitoring using SiteScope 91
SAP Portal monitor 367
performance counters 371
SAPGUI monitor 362
performance counters 365
Server Resources monitor 98
performance counters 100
Services.UserDataPoint(Value,Name) 41
Siebel monitor 374
performance counters 378
Siebel Server Manager monitor 381
performance counters 382
SilverStream monitor 223
performance counters 226
SiteScope monitor 92
SiteScope Oracle9iAS monitor
performance counters 217
SNMP Resources monitors 59, 83
configuring SiteScope SNMP by MIB monitor 87
monitoring using SiteScope 91
SQL Server monitor 273
performance counters 276
SSL Connections per Second graph 56
Sybase monitor 311
performance counters 314
System Resource monitors 59, 61, 75, 83, 91, 97, 103
Antara FlameThrower monitor 61
Server Resources monitor 98
SiteScope monitor 92
SNMP Resources monitor 83
UNIX Resources monitor 106
Windows Resources monitor 76

T

Throughput graph 50
Total Transactions per Second (Passed) graph 44
Transaction monitor 43
transactions
Transaction monitor 43
Transaction Response Time graph 43
Transactions per Second (Failed, Stopped) graph 44
Transactions per Second (Passed) graph 44
Troubleshooting and Knowledge Base 18
Tuxedo monitor 407
performance counters 410

U

UNIX Resources monitor 106
performance counters 108
UNIX, activating rstatd 104
updates, documentation 19
User-Defined Data Points graph 41

V

Vuser status
    Run-Time graphs 40
Vusers with Error Statistics graph 40

W

Web Application Server Resource monitors
        159, 161, 209
    Ariba monitor 161
    ATG Dynamo monitor 167
    BroadVision monitor 174
    ColdFusion monitor 188
    iPlanet (NAS) 196
    Microsoft ASP 209
    Oracle9iAS HTTP monitor 213
    SilverStream monitor 223
    WebLogic (JMX) monitor 231
    WebSphere (EPM) monitor 267
    WebSphere monitor 238
Web Resource monitors 49
Web Server Resource monitors 131
    Apache monitor 135
    iPlanet (SNMP) monitor 149
    iPlanet/Netscape monitor 144
    Microsoft IIS monitor 139
WebLogic (JMX) monitor 231
WebSphere monitor 238
    EPM 267
    performance counters 243
Windows Resources monitors 76
    performance counters 79