HP Performance Agent

For the HP-UX 11i v1 and Later

Software Version: 4.70

Installation and Configuration Guide

hp ®

i n v e n t

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.

Copyright Notices

Trademark Notices

UNIX® is a registered trademark of The Open Group.

Adobe® and Acrobat® are trademarks of Adobe Systems Incorporated.

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

Microsoft® is a U.S. registered trademark of Microsoft Corporation.

All other product names are the property of their respective trademark or service mark holders and are hereby acknowledged.

Support

You can visit the HP Software Support web site at:

www.hp.com/go/hpsoftwaresupport

HP Software online support provides an efficient way to access interactive technical support tools. As a valued support customer, you can benefit by using the support 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

Contents

1 Installing or Upgrading HP Performance Agent

Introduction

HP Performance Agent captures performance, resource, and transaction data from your HP 9000 server, Integrity server, or workstation. Using minimal system resources, Performance Agent continuously collects, logs, summarizes, and time stamps data, and detects alarm conditions on current and historical data across your system. You can analyze the data using spreadsheet programs, or analysis products such as HP Performance Manager, or third-party analysis products.

Also, Performance Agent provides data access to other HP products like HP Performance Manager, HP Network Node Manager, and HP Operations Manager.

Performance Agent supports monitoring of HP Integrity Virtual Machines, for more information, refer to the section, Performance Agent on a Virtualized Environment on page 17.

▶ Performance Manager in this document refers only to versions 4.0 and later. The name Performance Manager 3.x is used throughout this document to refer to the product that was formerly known as PerfView.

Performance Agent uses data source integration (DSI) technology to receive, alarm on, and log data from external data sources such as applications, databases, networks, and other systems. The comprehensive data logged and stored by Performance Agent allows you to:

- Characterize the workloads in the environment.

- Analyze resource usage and load balance.

- Perform trend analyses on historical data to isolate and identify bottlenecks.

- Perform service-level management based on transaction response time.

- Perform capacity planning.

- Respond to alarm conditions.

- Solve system management problems before they arise.

For a comprehensive description of the product, see the *HP Performance Agent for UNIX User's Manual*.

If you are running Performance Agent on a factory integrated system, skip this chapter and go directly to, Chapter 2, Starting and Running HP Performance Agent.

Installation Requirements

Before installing or upgrading Performance Agent, make sure that your system meets the requirements described in this section.

▶ If you are planning to run a large number of `dsilog` processes, you may need to increase the values of the HP-UX kernel parameters `shmmni` and `nflocks`. `shmmni` specifies the maximum number of shared memory segments. `nflocks` specifies the maximum number of file locks on a system. The default value for each is 200.

Each active DSI log file set uses a shared memory segment and one or more file locks. You can change the settings for `shmmni` and `nflocks` through the System Administration and Maintenance utility (SAM).

Hardware

Performance Agent runs on HP 9000 and Integrity Servers, and HP Workstations supported on the HP-UX 11i Version 1, Version 2 and Version 3 release and later.

Software

This version of Performance Agent requires releases 11i v1 or later of the HP-UX operating system.

Communication Protocols

Performance Agent supports the following communication protocols:

- HTTP(S) 1.1
- DCE

▶ If you are installing Performance Agent 4.70 on a system which has HP Software products such as HP Operations Agent, HP Operations Manager Unix Management Server, HP Performance Manager, HP Performance Insight, and OV Internet Service, it is recommended to restart them after Performance Agent 4.70 installation is completed.

1 If you are adding new hardware or making any configuration changes, it is recommended to stop scopeux and restart it to make the changes to take effect.

2 All the default OS daemons and services should be enabled and running for the HP-UX system.

Disk Space

Performance Agent installs in the /opt/perf/ and /opt/OV/ directories, and creates its log and status files in the /var/opt/perf/ and /var/opt/OV/ directories.

• For first time installation of Performance Agent, 100 MB of disk space is required in the /opt/perf/ and /opt/OV/ directories.

• For Performance Agent databases and status files, allow for 125 MB of disk space in the /var/opt/perf/ and /var/opt/OV/ directories.

For a description of how to use the parm file to configure log file data storage, see the "Parm File" section in Chapter 2 of your *HP Performance Agent for UNIX User's Manual*.

Install or Upgrade Procedures

Performance Agent is available on the DVD-ROM installation media in several file sets. The size of the product is approximately 260 MB.

If you have previously installed version of Performance Agent or GlancePlus on the system, stop any performance tools or processes that may be running. For instructions, refer to the section Stopping Active Performance Tools or Processes. For installation instructions, refer to the section, Installing or Upgrading Performance Agent on page 14.

Stopping Active Performance Tools or Processes

1 Log in as **root** user.

2 To check for active performance tools, run perfstat by typing:

   /opt/perf/bin/perfstat

   If perfstat reports any active performance tools such as GlancePlus, you must stop them before installing Performance Agent.

▶ Before stopping any of the performance tools, make sure that none of the users are currently using these tools.

3 If a previously-installed version of Performance Agent is running, you must stop it by typing:

   /opt/perf/bin/mwa stop

▶ Customized configuration files such as the parm, alarmdef, ttd.conf, and perflbd.rc files as well as any customized log files will not be overwritten by the new installation. The new configuration files are installed in the /opt/perf/newconfig/ directory.

4 Make sure the midaemon (measurement interface daemon) is not active by typing:

   ps -ef | grep midaemon

   If the midaemon is still active, type:

   /opt/perf/bin/midaemon -T

5   Make sure the `ttd` (transaction tracking daemon) is not active by typing:

ps -ef | grep ttd

If the `ttd` is still active, type:

/opt/perf/bin/ttd -k

▶   If you stop the `ttd`, any ARM-instrumented applications that are running *must* also be stopped before you can restart `ttd` and Performance Agent process.

6   Run `perfstat` again to make sure that no performance tools or processes are active. When all tools or processes have been stopped, proceed with the installation.

Installing or Upgrading Performance Agent

1   Log in as **root** user.

2   Run the `swinstall` program to install Performance Agent the same way you install any other software package on an HP-UX system.

3   The Performance Agent product is included as part of the GlancePlus Pak.

   a   If you want to install GlancePlus Pak, modify the `swinstall` source to point to your installation media and select the GlancePlus Pak bundle,
   (or)

   b   If you have not purchased the GlancePlus Pak, modify the `swinstall` source to point to your installation media and select the Performance Agent bundle

4   The install process appends information to the end of the log files `swinstall.log` and `swagent.log`, available in the location `/var/adm/sw/`. These files contain system messages about the installation and records of any problems the system encountered. Check these files for error messages during installation and take appropriate action as needed.

To read the last 100 lines of the `swinstall.log` file, type the following command:

tail -100 /var/adm/sw/swinstall.log

If you run the program swinstall in an interactive mode, you can view the swinstall.log file while swinstall runs. Else, you can check the files swinstall.log and swagent.log available in /var/adm/sw/ for any errors or problems.

Installation of Performance Agent automatically updates files used by /etc/profile/ and allows you to access the product's executable in /opt/perf/bin/. You can also access the man pages without explicitly updating the shell environment variables. This will be effective from the next time you log in.

Performance Agent installation is now complete. For information on starting and running Performance Agent, refer to Chapter 2, Starting and Running HP Performance Agent.

► If you are also running the GlancePlus on your system, make sure to update GlancePlus to the same release version as Performance Agent. Both Performance Agent and GlancePlus must always be the same version.

► During a cluster maintenance operation, you must stop Performance Agent and restart after the maintenance operation is completed.

Deploying Performance Agent Using Operations Manager

If you are using Operations Manager for UNIX 7.x and 8.x, you can install Performance Agent from the management server to an HP-UX managed node.

⚠ You cannot install this version of Performance Agent on Itanium systems that have Operations Manager 8.x management server installations.

For installation instructions on Operations Manager for UNIX versions 7.x, see Chapter 6, "About HP Performance Agent for UNIX," in the *HP Operations Manager for UNIX Administrator's Reference Volume II*. For installation instructions on Operations Manager for UNIX versions 8.x, see Chapter 4, "HP Performance Agent," in the *HP Operations Manager Administrator's Reference*.

Installing Files on a Remote Drive

You can link the `/var/opt/perf/` directory to a different file system, if the disk space is a concern. For example, if your `/var/opt/perf/` directory can be redirected to a non-local file system, such as a remote NFS mount point. Make sure you take care of the following considerations:

- Performance Agent does not support remote NFS logging in a WAN environment. Performance Agent is designed to optimally use locally mounted log files.

- Remote NFS mounting of log files in a LAN environment could cause network overloading depending on data logging thresholds, which can negatively impact the performance of Performance Agent and other networked applications. Also, log file maintenance can take substantially longer due to the remote access, which can cause time-out errors in alarm processing.

➤ For more information and specific details about this version of Performance Agent, refer to the *Release Notes* available in the `/opt/perf/ReleaseNotes/ovpa` file.

Performance Agent on a Virtualized Environment

This version of Performance Agent supports monitoring of HP Integrity Virtual Machines. Performance Agent can be installed both on the guest and host operating systems. For installation procedures refer to the section, Install or Upgrade Procedures on page 13.

▶ HPVM 1.0 is not supported by Performance Agent and HP GlancePlus (GlancePlus).

Performance Agent on the Host Operating System

Host operating system is the system on which HPVM is installed. When installed on the host machine, Performance Agent captures the following:

- System wide performance, application, transaction and resource usage data.
- Individual virtual machine's performance and resource usage data.

    A new class of metrics, BYLS, is introduced to record performance and resource data for individual guest operating systems.

Performance Agent on the Guest Operating System

Guest operating systems are the systems that run on HPVM. When installed on the guest operating system, Performance Agent captures system wide performance, resource, and transaction data. All the resource utilization values captured represent the logical utilizations.

▶ For complete list of metrics and descriptions, refer to the *Dictionary of Performance Metrics*.

Removing Performance Agent

1 Make sure you are logged in as **root**.

2 Run the swremove program to uninstall Performance Agent the same way you uninstall any other software package on an HP-UX system.

3 The Performance Agent product is included as part of the GlancePlus Pak.

   a If you want to uninstall GlancePlus Pak, modify the swinstall source to point to your installation media and select the GlancePlus Pak bundle,
      (or)

   b If you have not purchased the GlancePlus Pak, modify the swinstall source to point to your installation media and select the Performance Agent bundle.

4 The uninstall process appends information to the end of the two log files named /var/adm/sw/swremove.log and /var/adm/sw/swagent.log. These files contain system messages and records of any problems the system encountered during uninstallation. Check these log files for any error messages that were logged during the uninstallation and take appropriate action as needed.

   To read the last 100 lines of the swremove.log file, type the following command:

   tail -100 /var/adm/sw/swremove.log

   If you run swremove in an interactive mode, you can view the swremove.log file while swremove is running. Else you can check the files swremove.log and swagent.log in /var/adm/sw/ for any errors or problems during uninstallation.

It is possible that some product packages may remain installed on the system, if those packages are shared across other HP Software products and are required by other tools. They will be removed only when the last tool requiring them is also removed.

2 Starting and Running HP Performance Agent

Introduction

This chapter contains instructions to start and run Performance Agent on a HP-UX system.

The following topics are discussed:

- procedures for Starting and Stopping Performance Agent
- Communicating Across Firewall
- setting variables in the `/etc/rc.config.d/ovpa` file
- running Performance Agent
- Parm File Configuration for HP-UX systems
- Configuring Data Sources
- configuring alarm definitions
- configuring coda

► If you want to log data from other sources using data source integration (DSI), refer to the *HP Performance Agent for UNIX Data Source Integration Guide.*

Performance Agent supports the HTTP data communication mechanism, along with the legacy DCE mode. If you are installing Performance Agent for the first time, the default data communication mode is HTTP. If you are using Performance Agent version 4.5, the previously used DCE data communication mode is used by default. If you want to move to the new HTTP mode of data communication, you can change the configuration after installation. Performance Agent supports HTTPS data communication only in an HP Operations Manager 8.x environment.

Starting and Stopping Performance Agent

When installation is complete, you can start Performance Agent. The Performance Agent scripts, ovpa and mwa, allows you to perform the following tasks:

- start all or some of the processes
- stop or restart processes that are currently running.

If you are installing Performance Agent for the first time, the default data communication mode is HTTP. If you are using Performance Agent version 4.5, the previously used DCE data communication mode is enabled by default. For information on changing the data communication protocol, see Changing Protocols on page 25.

Depending on the data communication protocol you want to enable, you can use the ovpa or mwa script to start or stop Performance Agent.

▶ It is recommended that you use the ovpa script to start Performance Agentand enable the HTTP data communication protocol. However, if you want to use the DCE data communication protocol, the mwa script is provided for backward compatibility.

The table below lists the different services that can be started for the different protocols.

Table 1   Performance Agent services started for different protocols.

Services started for HTTP protocol	Services started for DCE protocol
scopeux	scopeux
coda	coda
perfalarm	perfalarm
midaemon	midaemon
ttd	ttd
ovc	ovc
ovbbccb	ovbbccb

Table 1   Performance Agent services started for different protocols.

Services started for HTTP protocol	Services started for DCE protocol
	`perflbd`
	`rep_server`
	`alarmgen (if perfalarm is not present)`

▶ The `perflbd`, `rep_server`, and `alarmgen` processes are used for DCE communication

Before you start Performance Agent, check for any processes which are running by typing:

/opt/perf/bin/perfstat

Using the Performance Agent script

Listed below are the steps to start or stop Performance Agent using the `ovpa` script.

To Start Performance Agent

To start Performance Agent and its processes using Performance Agent:

1   Log in as user **root**.

2   Type: **/opt/perf/bin/ovpa start**

The `ovpa start` script starts Performance Agent and the following processes:

— `scopeux` (the data collector)

— `midaemon` (the measurement interface daemon)

— `ttd` (the transaction tracking daemon)

— `coda` daemon

— ovc

— ovbbccb

— the alarm generator.

As the script executes, the status of the processes that are started is displayed on the screen.

To Stop Performance Agent

You can stop Performance Agent processes while they are running and restart them using the ovpa script and appropriate options.

The command ovpa stop stops all Performance Agent processes except the following:

— ttd (the transaction tracking daemon)

— ovc

— ovbbccb

ttd should always be left running.

▶ If Operations Manager agent is running on the system, Performance Agent stop will not stop the coda daemon.

‼️ If you must stop ttd, any ARM-instrumented applications that are running must also be stopped before you restart ttd and Performance Agent processes.

Restart Performance Agent

The following are the command options available to restart Performance Agent:

• ovpa restart server

Running the command performs the following tasks:

— temporarily stop and restart coda

— disable the alarming and access for clients such as Performance Manager

— read the datasources file

— stop and restart the perfalarm processes

— read the `alarmdef` file.

- `ovpa restart`

  Running the command performs the following tasks:

  — temporarily stop and restart `scopeux` and the server processes

  — read the `parm` file

  — prompts the transaction daemon `ttd` to read the configuration file `ttd.conf`.

- `ovpa restart alarm`

  Running this command prompts the alarm generator process to reread the `alarmdef` file. If you have made changes to the `alarmdef` file, the new alarm definitions will take effect without restarting all Performance Agent processes. This command does not disrupt other processes.

Using the mwa script

Listed below are the steps to start or stop Performance Agent using the `mwa` script.

To Start Performance Agent

To start Performance Agent and its processes using `mwa`:

1 Log in as user **root**.

2 Start Performance Agent and its processes by typing:

   /opt/perf/bin/mwa start

   The `mwa start` script starts Performance Agent and all its processes, including `scopeux` (data collector), `midaemon` (measurement interface daemon), `ttd` (transaction tracking daemon), `coda`, `ovc`, `ovbbccb`, `perflbd`, `rep_server`, and the alarm generator. As the script executes, the names of some of the processes that are running and the files that are in use are displayed on the screen.

To Stop Performance Agent

You can stop Performance Agent processes while they are running and restart them using the mwa script and its appropriate options.

- Running the command `mwa stop`, stops all Performance Agent processes except `ttd` (the transaction tracking daemon), `ovc`, and `ovbbccb`.

▶
- `ttd` should always be running.
- `mwa stop` does not stop the `coda` daemon if other products are using `coda`.

To Restart Performance Agent

- To restart Performance Agent, use the command:

  `mwa restart`

  Changes to configuration files will *not* take effect on your system unless you restart the processes.

- `mwa restart server`

  Running the command performs the following tasks:

  — stop and restart the `coda` daemon and the repository servers

  — temporarily disable alarming and access for clients such as Performance Manager

  — reread the `perflbd.rc` file

  — stop and start the alarm generator processes

  — reread the `alarmdef` file

The HTTP based alarm generator, `perfalarm`, is enabled by default.

- `mwa restart`

  Running the command performs the following tasks:

  — temporarily stop and then start the server processes and all the daemons including `coda`, `scopeux` and the transaction daemon `ttd`

  — rereads the `parm` file and the `ttd.conf` transaction configuration file.

🛑 If you stop `ttd`, any ARM-instrumented applications that are running must also be stopped before you restart `ttd` and Performance Agent processes.

- `mwa restart alarm` cause the alarm generator processes to temporarily stop and then restart and reread the `alarmdef` file. This action does not disrupt any other processes.

Changing Protocols

During first time installation of Performance Agent, the protocol selected is written to the `/etc/rc.config.d/ovpa` file as an addition to the environment variable `MWA_PROTOCOL` (for example, `MWA_PROTOCOL=http`).

To switch to DCE mode

You cannot use the `ovpa` script to start or stop the DCE data communication components. You must set the `MWA_PROTOCOL` parameter in the `/etc/rc.config.d/ovpa` file to `dce`, and start Performance Agent using the `mwa` script.

To switch to HTTP mode

If you want to switch to the HTTP protocol, you must set the `MWA_PROTOCOL` parameter in the `/etc/rc.config.d/ovpa` file to `http` and restart Performance Agent. To start Performance Agent using the HTTP data communication mode, you can either use the `ovpa` or `mwa` script. The `mwa` script starts the HTTP, DCE, or NCS data communication components depending on the value set for `MWA_PROTOCOL`. If `MWA_PROTOCOL` is set to `http`, `ovpa` and `mwa` scripts start the same components.

▶ The HTTP communication protocol is always enabled, irrespective of the protocol you have selected during installation. The daemons used for HTTP data communication are installed by default and will be active on your system.

Starting and Stopping Automatically

The process of automatically starting Performance Agent whenever the system restarts, and automatically stopping whenever the system shuts down is controlled by the Performance Agent startup and shutdown script, and the Performance Agent startup configuration file `/etc/rc.config.d/ovpa`.

The Performance Agent startup configuration file contains various environment variables that control the working of Performance Agent at system startup. This file is a source file for the Performance Agent system startup and shutdown script.

You can modify the following environment and shell variables to change the default behavior of Performance Agent:

- MWA_START controls the auto-start of Performance Agent whenever your system reboots. The variable can have one of the following values:

  0   Do not start Performance Agent at the system startup

  1   Start Performance Agent at the system startup

- The MWA_START_COMMAND is a variable that specifies the script options that are used to start Performance Agent whenever your system reboots. Normally, the variable is set to /opt/perf/bin/mwa start.

► The Performance Agent startup configuration file is not overwritten when Performance Agent is patched or updated, so that any customizations you made are preserved.

Status Checking

Several status files are created in the /var/opt/perf/ and /var/opt/OV/ directories when Performance Agent is started. You can check the status of Performance Agent processes using the perfstat command.

The following status files contain diagnostic information you can use to troubleshoot problems that may arise with the Performance Agent processes.

```
/var/opt/perf/status.alarmgen
/var/opt/perf/status.perflbd
/var/opt/perf/status.rep_server
/var/opt/perf/status.scope
/var/opt/perf/status.perfalarm
/var/opt/perf/status.ttd
/var/opt/perf/status.mi
/var/opt/OV/log/coda.txt
```

► Every time the Performance Agent process writes a message to its status file, it checks to see if the file is larger than one MB. If it is, the file is renamed to status.filename.old and a new status file is created.

Examples Directory

The `/opt/perf/examples/` directory contains examples of configuration files, syntax files, and sample program files that can be used to customize your HP Performance Tools. For example, the `/opt/perf/example/config/` subdirectory contains sample alarm definitions and examples of `parm` file application-specific parameters. For more information, see the `/opt/perf/examples/README` file.

Communicating Across Firewall

A firewall can be defined as a method for filtering the flow of data between one network and another. Performance Agent now supports HTTP 1.1 based communications interface for data access between clients such as Performance Manager and Reporter and server applications, in addition to the previously supported communication mechanism through a packet-filtering network firewall.

➤ Performance Agent supports certificate-based secure (HTTPS) data communication only in the Operations Manager 8.x environment. For more information, see Using Certificates on page 43.

The HTTP based interface is flexible, because it can use proxies, requires fewer ports and is firewall friendly. The DCE interface is still available for use.

The following sections explain how to configure HTTP and DCE communication across a firewall:

- Communicating in the HTTP Environment
- Communicating in the DCE Environment

**Figure 1   Communicating with Performance Agent in a Firewall
Environment**
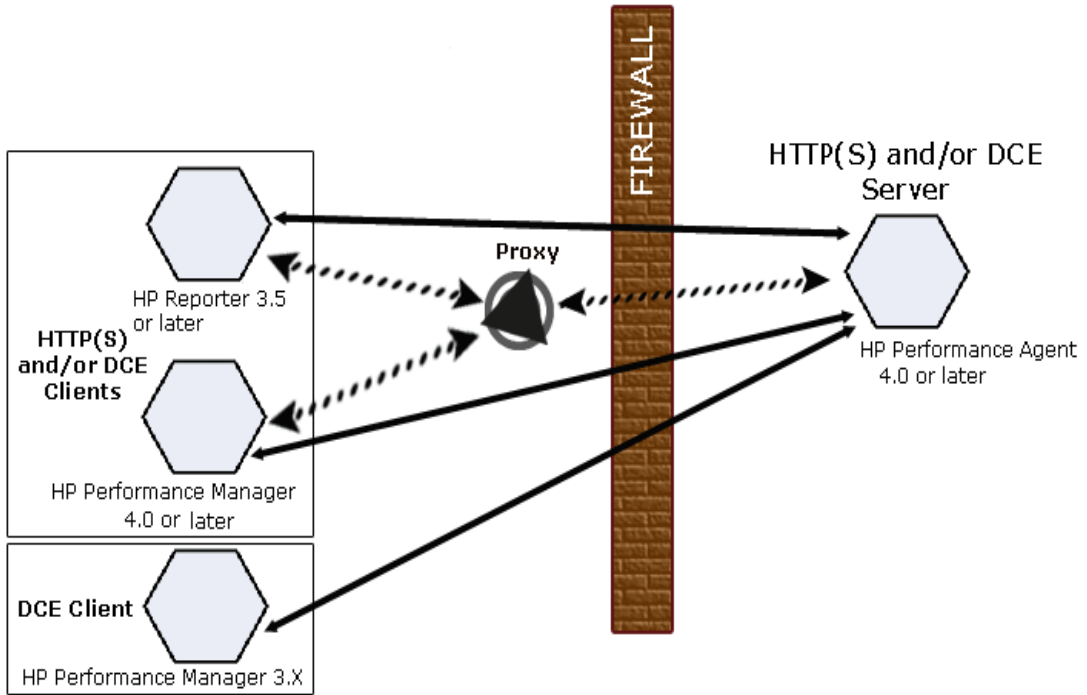
▶   The name Performance Manager 3.x is used throughout this document to
refer to the product that was formerly known as PerfView.

Communicating in the HTTP Environment

For firewall configuration it is important to know which system initiates the
communication (client) and which receives communication requests (server),
so that the firewall rules can be set up accordingly. In a typical remote
communication, a client, using the source port, connects to a server that is

listening on the destination port on a remote system. Understand your firewall environment including the client and server data flow. To configure communications with Performance Agent in a firewall environment, perform the following tasks:

1 Configure Performance Agent Ports

2 Configure HTTP Clients in a Firewall Environment

3 Verify Firewall Configuration

Figure 1 on page 29 shows how Performance Agent communicates with Reporter (version 3.5 or later) and Performance Manager (version 4.0 or later) through a firewall. Performance Agent is an HTTP or HTTPS server. Reporter and Performance Manager 4.x are HTTP clients. Performance Manager 5.0 can be an HTTP or HTTPS client. If an HTTP proxy is used, Reporter and Performance Manager communicate with Performance Agent through the proxy.

Performance Manager version 3.x uses the PerfView technology. PerfView does not use the HTTP datacomm components, but it connects to Performance Agent 4.5 when the DCE data communication mode is enabled.

Configure Performance Agent Ports

You can configure Performance Agent ports in a firewall environment in one of the two ways:

• Configure Two-Port Communication

• Configure Single-Port Communication

• Verify Port Settings

On the Performance Agent system using BBC5, by default, the BBC communication broker uses port 383 and `coda` uses a dynamically allocated port.

Configure Port Settings for the BBC Communication Broker

You can configure the port settings of the default port used by the BBC communication broker. Use the `ovconfchg` command to change the port settings on the Performance Agent system. You can use one of the following options:

— **ovconfchg -ns bbc.cb -set SERVER_PORT <*port*>**

(Or)

— **ovconfchg -ns bbc.cb.ports -set PORTS <*domain*>:<*port*>**

Example: **ovconfchg -ns bbc.cb.ports -set PORTS xyz.abc.com:50383**

The second option is the preferred way of changing ports.

Restart Performance Agent using the following command:

`ovpa restart server`

Configure Two-Port Communication

By default, `coda` daemon uses a dynamically chosen second port, in addition to port 383 used by the BBC communication broker. You can configure the port settings of `coda` to listen at a well known port of your choice using the `ovconfchg` command. Type the following commands:

`ovconfchg -ns coda.comm -set SERVER_PORT <`*`portnumber`*`>`

`ovpa restart server`

▶ Using a dynamic port when connecting to Performance Agent remotely through a firewall can be difficult, because you may not know which firewall ports to open.

Configure Single-Port Communication

On the Performance Agent system, the BBC communication broker uses port 383 and `coda` uses a port that is dynamically allocated. You can configure the port settings for `coda` to share the same port used by the communication broker using the `ovconfchg` command. Type the following commands:

`ovconfchg -ns coda.comm -set SERVER_BIND_ADDR localhost`

`ovpa restart server`

▶ To enable two-port communication from single-port communication, type the following command:

`ovconfchg -ns coda.comm -set SERVER_BIND_ADDR`

Verify Port Settings

To verify the port settings, type the following command:

`perfstat -d`

The output displays the following information:

— port number of the port used by `Coda`

— port number of the port used by `BBC communication broker`

— the port settings configured

— indicates if secure communication is enabled

— indicates if coda metric collection is enabled

For example:

```
Datacomm configuration:

-----------------------

 Coda Port                           49552 (Dynamic)

                                     Two port Communication

 BBC communication broker port          383

 SSL security                           NONE

 Coda Metric Collection (Prospector)    Disabled
```

Configure HTTP Clients in a Firewall Environment

There are two ways to configure HTTP clients in a firewall environment:

- Configuring HTTP Clients (Reporter/Performance Manager) with HTTP Proxy
- Configuring HTTP Clients (Reporter/Performance Manager) without HTTP Proxy

In both cases, to access data from Performance Agent nodes, only one port needs to be opened on the HTTP server (Performance Agent) side.

Configuring HTTP Clients (Reporter/Performance Manager) with HTTP Proxy

It is recommended that you use HTTP proxies when communicating through a firewall. This simplifies the configuration by using proxies that are often already in use in your environment. The firewall must be open for exactly one port if proxies are to be used in both directions. To access data collected by Performance Agent, ports for the HTTP server (Performance Agent) and the HTTP client (Reporter and Performance Manager) must be opened.

▶ It is recommended that you do not change the default 383 port.

When an HTTP proxy is used, Reporter and/or Performance Manager for Windows and UNIX need to be configured to specify the proxy to be used to contact Performance Agent.

To configure Performance Manager versions 5.0 and later, and Reporter 3.7 and later:

Type the following command,

`ovconfchg -ns bbc.http -set PROXY proxy:port+(a)-(b)`

The variables *a* and *b* are comma separated lists of hostnames, networks, and IP addresses that apply to the proxy. Multiple proxies may be defined for one PROXY key using the ";" or "," delimiter. "-" before the list indicates that those entities do not use this proxy, "+" before the list indicates that those entities do use this proxy. The first matching proxy is used.

To configure Reporter versions 3.6 and earlier, and Performance Manager 4.x:

Edit the `/var/opt/OV/conf/BBC/default.txt` configuration file.

In the `[DEFAULT]` section of the `default.txt` file, locate the lines that relate to the PROXY and set the PROXY parameter as follows.

`PROXY web-proxy.hp.com:8088-(localhost, *.hp.com) + (*)`

In this example, the proxy `web-proxy` will be used with port 8088 for every server (*) except requests for the local machine (localhost) and requests internal to HP (matching *.hp.com, for example **www.hp.com**.

Configuring HTTP Clients (Reporter/Performance Manager) without HTTP Proxy

If HTTP proxies are not available, additional configuration settings are required on the HTTP clients (Reporter and Performance Manager system).

If Reporter and Performance Manager for Windows are installed on the same system and both access Performance Agent in parallel, you can specify a port range as described in this section. If they are running on different systems, you can specify a single port for each. Depending on the versions of Performance Manager and Reporter you are using select from the following options:

Configure Performance Manager 5.0 and later, and Reporter 3.7 as follows:

Type the following command,

`ovconfchg -ns bbc.http -set CLIENT_PORT` *<port range>*

Where *<port range>* is the range of ports you want to use.

For example:

`ovconfchg -ns bbc.http -set CLIENT_PORT 14000-14003`

Configure Reporter versions 3.6 and earlier, and Performance Manager 4.x as follows:

Edit the `/var/opt/OV/conf/BBC/default.txt` file as follows.

1 Locate the lines that apply to `CLIENT_PORT` and uncomment the line;`CLIENT_PORT =`.

2 Specify the port range for the `CLIENT_PORT` parameter. For example:

   `CLIENT_PORT = <port range>`

   Where *<port range>* is the range of ports you want to use. For example:

   `CLIENT_PORT = 14000-14003`

Verify Firewall Configuration

To verify your configuration, use the command:

`ovcodautil -ping -n <system name>`

This output of this command indicates the status of your communication settings.

Example:

`/usr/lpp/OV/bin/ovcodautil -ping -n ovpaixt6`

The following message appears:

`Ping of 'OvBbcCb' at: 'http://ovpaixt6:383/Hewlett-Packard/OpenView/BBC/ping' successful`

`Ping of 'Coda' at: 'http://ovpaixt6:34425/Hewlett-Packard/OpenView/Coda/' successful`

Configuring Systems with Multiple IP Addresses

If your environment includes systems with multiple network interfaces and IP addresses and you want to use a dedicated interface for the HTTP-based communication, then you can use the parameters `CLIENT_BIND_ADDR` and `SERVER_BIND_ADDR` to specify the IP address that should be used.

• If you have multiple network interfaces and IP addresses on the Performance Agent (Server) system, specify the `SERVER_BIND_ADDR` parameter as follows:

```
ovconfchg -ns bbc.http -set SERVER_BIND_ADDR <IP Address>
```

- If you have multiple network interfaces and IP addresses on the Performance Manager 5.0 (client) system, specify the CLIENT_BIND_ADDR parameter as follows:

```
ovconfchg -ns bbc.http -set CLIENT_BIND_ADDR <IP Address>
```

- If you have multiple network interfaces and IP addresses on the Reporter/Performance Manager 4.x system, specify the CLIENT_BIND_ADDR parameter.

  Edit the /var/opt/OV/conf/BBC/default.txt file as follows:

  a   Locate the lines that apply to CLIENT_BIND_ADDR and uncomment the line

  ;CLIENT_BIND_ADDR =

  b   Specify the IP address for the CLIENT_BIND_ADDR parameter.

Communicating in the DCE Environment

In the DCE environment, Performance Agent uses dynamically allocated socket port numbers for interprocess communication. To communicate through a packet-filtering network firewall, you must configure the Performance Agent servers to use statically defined port numbers.

### Configuring Performance Manager and Performance Agent Communication

The method of configuring the socket port numbers for Performance Agent depends on the version of Performance Manager that will be communicating with Performance Agent. To configure the socket port numbers, follow the instructions in Configuring Performance Manager C.03.00 and Later with Performance Agent Communication.

Configuring Performance Manager C.03.00 and Later with Performance Agent Communication

The way to configure Performance Agent and Performance Manager firewall communication depends on which of these two programs is the source of the communication.

Configuring Performance Manager C.03.00 and Later (Source) with Performance Agent Communication

When Performance Manager is the source, it communicates with Performance Agent using the TCP protocol with the TCP socket port numbers shown in the following figure.
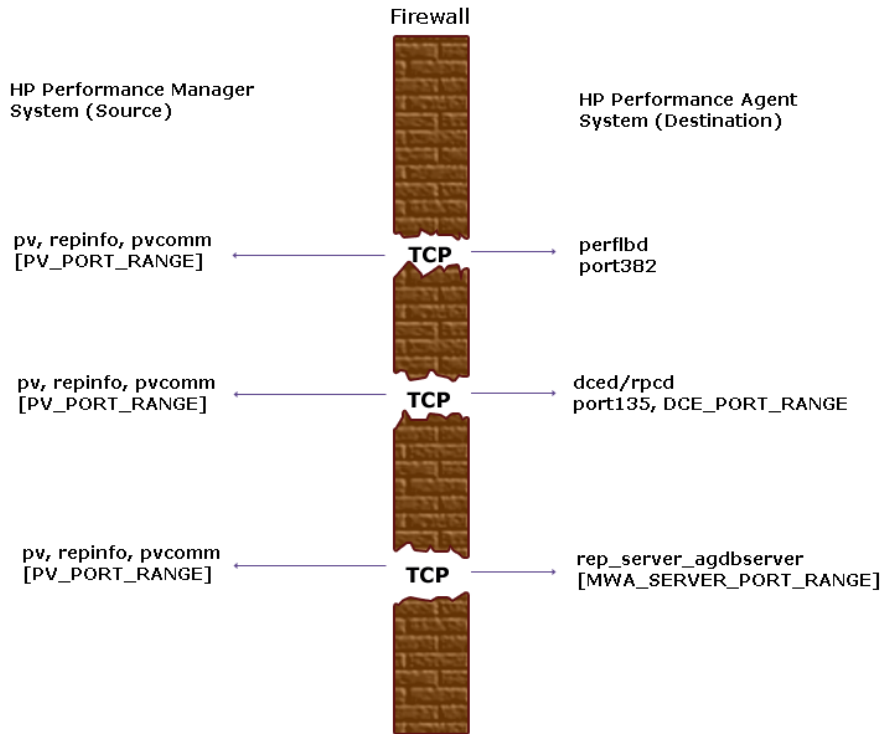
To configure the MWA_SERVER_PORT_RANGE as statically defined TCP socket port numbers, add the following entries to the /etc/services file:

agdbserver *xxxx***/tcp**
rep_server *yyyy***/tcp**

where *xxxx* and *yyyy* specify unused port numbers. agdbserver and rep_server register at the specified port numbers. If there are multiple data sources configured in the perflbd.rc file, the first rep_server uses the *yyyy* port number specified above. All other rep_servers add one to the last used port number.

For example, if you include the following lines in the /etc/services file:

```
agdbserver     20001/tcp
rep_server     20002/tcp
```

and there are three data sources configured in the perflbd.rc file, Performance Agent will use the following TCP port numbers:

```
agdbserver    20001
rep_server    20002
rep_server    20003
rep_server    20004
```

Based on the /etc/services example above, the MWA_RANGE will be 20001-20004.

Configuring Performance Agent (Source) with Performance Manager C.03.00 and Later Communication

When Performance Agent is the source, it communicates with Performance Manager using the TCP protocol with the TCP socket port numbers shown in the following figure.
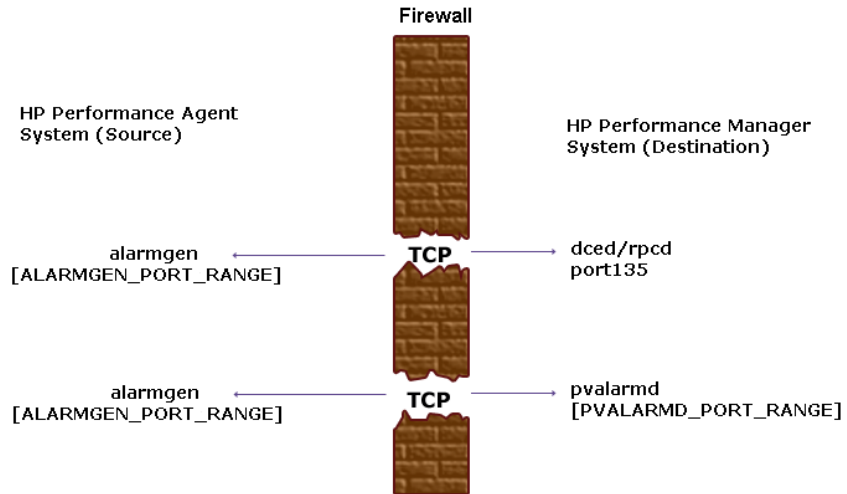
If you previously had Performance Agent communicating to Performance Manager through a firewall, port 135/UDP was open in the firewall. The firewall configuration must now be changed to open port 135/TCP. This is because the protocol used to connect to port 135 on the Performance Manager system was changed from UDP to TCP, regardless of the version of Performance Manager.

To configure the ALARMGEN_PORT_RANGE for the alarmgen process, edit the file /var/opt/perf/vppa.env and set the RPC_RESTRICTED_PORTS to the following:

RPC_RESTRICTED_PORTS=ncacn_ip_tcp[*xxxx-yyyy*]

where *xxxx-yyyy* is a range of unused port numbers. The formula for calculating the port range is:

```
2 *  (# of Performance Manager systems receiving alarms from the
Performance Agent system)
```

For example, if the Performance Agent system was sending alarms to two Performance Manager systems, set RPC_RESTRICTED_PORTS to the following range in the vppa.env file:

RPC_RESTRICTED_PORTS=ncacn_ip_tcp[30001-30004]

► This environment variable affects only the ports that are used for communication outside the localhost. Ports that are used internal to the local host, such as local communication between alarmgen and rep_server, are not affected by this variable.

Restart the Performance Agent servers using /opt/perf/bin/mwa restart server to make the port restriction take effect.

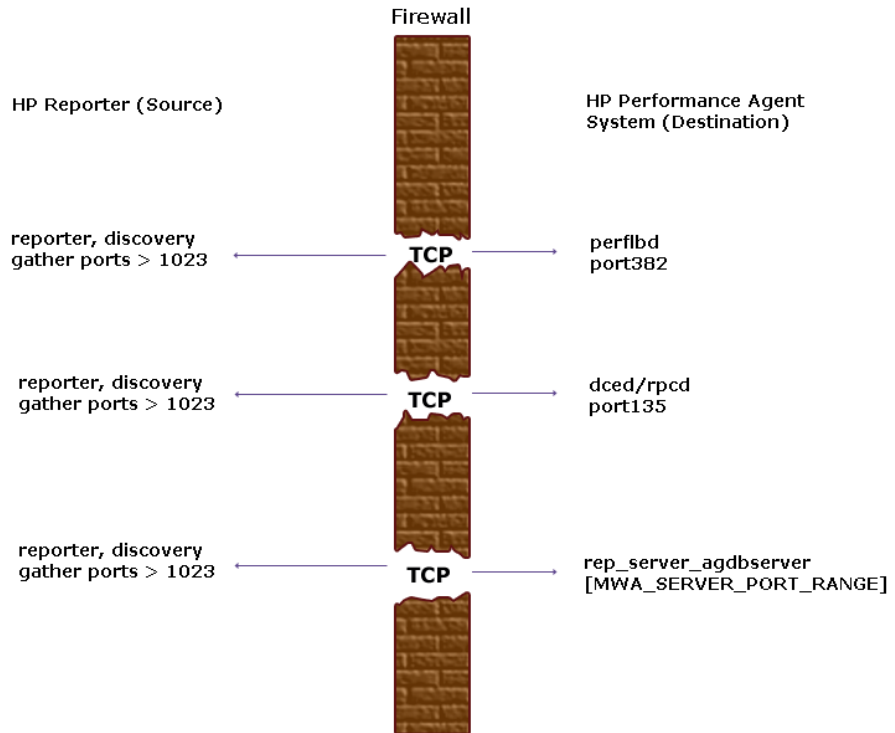Refer to the *HP Performance Manager Installation Guide* to determine the PV_PORT_RANGE.

► Sending alarms from Performance Agent to Performance Manager through firewalls that use Network Address Translation (NAT) IP addresses is not supported.

Configuring Reporter and Performance Agent Communication

The configuration of the socket port numbers for Performance Agent depends on the version of Reporter that will be communicating with Performance Agent. To configure the socket port numbers, follow the instructions in Configuring Reporter A.03.50 and Later with Performance Agent Communication.

Configuring Reporter A.03.50 and Later with Performance Agent Communication

Reporter communicates with Performance Agent using the TCP protocol with the TCP socket port numbers shown in the following figure.

Firewall

HP Reporter (Source)   HP Performance Agent
                              System (Destination)

reporter, discovery   perflbd
gather ports > 1023   TCP   port382

reporter, discovery   dced/rpcd
gather ports > 1023   TCP   port135

reporter, discovery   rep_server_agdbserver
gather ports > 1023   TCP   [MWA_SERVER_PORT_RANGE]

For instructions on configuring the MWA_SERVER_PORT_RANGE as statically defined TCP socket port numbers, see Configuring Performance Manager C.03.00 and Later with Performance Agent Communication on page 36.

Restricting RPC Addresses

The RPC_SUPPORTED_NETADDRS environment variable is used to force the DCE/RPC service to bind to specific LAN cards in a multi-homed environment. If this environment variable is set, only addresses in the list are advertised in the endpoint map; addresses not found on the list are excluded from the server's list of available addresses.

To restrict the servers to using only a specified set of IP addresses, set the RPC_SUPPORTED_NETADDRS environment variable before starting the servers.

The syntax is:

RPC_SUPPORTED_NETADDRS=protocol:ip_address[,protocol:ip_add ress]

To set the environment variable for the Performance Agent servers, add the lines:

RPC_SUPPORTED_NETADDRS=ncadg_udp:192.1.1.1
export RPC_SUPPORTED_NETADDRS

to the /var/opt/perf/vppa.env file and then restart the servers using mwa restart server.

If you still have problems connecting, the problem may be with the endpoint map (dced/rpcd). Try setting the environment variable *before* starting dced/rpcd. Then restart the system to make the IP address restriction take effect.

Configuring Secure Communication

Performance Agent supports certificate-based secure communication and client authentication based communication.

Using Certificates

Performance Agent supports certificate-based secure data communication only in the Operations Manager 8.x environment.

To configure secure communication on your Operations Manager setup, refer to the *HP Operations Manager for UNIX Firewall Concepts and Configuration Guide*. For more information on Operations Manager 8.x HTTPS agent, refer to the *HP Operations Manager HTTPS Agent Concepts and Configuration Guide*.

If you have already configured HTTPS communication in the Operations Manager 8.x environment, make the following changes to configure secure communication between Performance Agent and Performance Manager 5.0.

▶ Reporter and Performance Manager 4.x do not support certificate-based secure communication.

On the Performance Agent system, set SSL_SECURITY to REMOTE for coda. Type the following commands:

ovconfchg -ns coda -set SSL_SECURITY REMOTE

ovcodautil -config

Using Client Authentication

Performance Agent enables optional authentication of client connections from products such as Performance Manger or Reporter. The authentication capability allows you to specify, for a given Performance Agent instance, which hosts are allowed to make client connections to that instance.

The Client Authentication feature enables/disables connections from any version of the Performance Manager and Reporter clients. Your client software does not need to be updated for you to take advantage of this feature.

For authorized clients the authentication process is transparent – their client connection proceeds as it has with previous versions of Performance Agent. Unauthorized clients receive a message indicating denial of service.

Enabling Authentication with the authip File

Authentication is enabled by the presence of a file called `authip`. On systems where HTTP communication is enabled, the `authip` file exists in the `/var/opt/OV/conf/perf/` directory. On systems where DCE communication is enabled, the `authip` file exists in the `/var/opt/perf/` directory. The `authip` file lists hosts from which client connections are to be permitted.

- If the `authip` file exists in the default directory, then its contents determine which hosts are permitted client connections. Clients running on the same host as the Performance Agent instance are automatically authenticated, which means the clients do not need an entry. A zero-length `authip` file dictates that only clients running on the Performance Agent host can connect.

- If the `authip` file does not exist in the default directory, then no authentication is performed and any client will be allowed to connect, as was the case with prior Performance Agent versions.

The `authip` file is checked each time a client attempts to register for service with Performance Agent. Performance Agent does not need to be restarted for changes to the `authip` file to become effective.

Note, however, that an existing authorized client session can continue its current connection despite a subsequent change in the server's `authip` file, which would otherwise disqualify it, until the client takes an action that requires re-registration with Performance Agent. Thus, an authorized Performance Manager connection continues to be permitted, regardless of changes in the Performance Agent `authip` file, until the data source to the Performance Agent host has been closed. If there is then an attempt to reopen the data source, the `authip` file is reread and the connection is denied.

In the case of Performance Manager registration for alarms, a previously authorized client will continue to receive alarms until the data source has been removed (not just closed) by the client. If you want to force removal of a client from the server's alarm generator database from the Performance Agent side, use the command:

agsysdb -delpv <host>

The Performance Agent client authentication capability requires that your network be able to resolve the client entries in the `authip` file. Depending upon the nature of the entries, this may require name services such as those provided by DNS, NIS, or `/etc/hosts` files.

A good test is to ensure that you can successfully "ping" each `authip` entry from the Performance Agent host. Client authentication works through a firewall with the same proviso that the client entries in the `authip` file be pingable from the Performance Agent host.

Formatting the authip File

The `authip` file must conform to the following format:

- One client host may be listed per line.
- Client entries can be in any one of the following formats:
    - Fully qualified domain name
    - Alias
    - IP address (must be in IPv4 dotted quad format)
- Client entries can have no embedded spaces.
- A line containing a pound sign (#) in the first column is taken as a comment, and is ignored.
- Blank or zero-length lines are ignored.
- The IP address may not have a leading zero. For example, the IP address 23.10.10.10 cannot be represented as 023.10.10.10.

Thus, given an `/etc/hosts` entry as follows:

```
123.456.789.1  testbox  testbox.group1.thecompany.com
```

any one of the following entries in the `authip` file would enable clients from the "testbox" host to connect:

```
#=========== Examples of authip file entries =============
#
# Use of an IP address
123.456.789.1

#
```

```
# Use of an alias
testbox

#
# Use of a fully qualified domain name
testbox.group1.thecompany.com

#=========== End of examples of authip file entries ========
```

Configuring Data Sources

Performance Agent uses the coda daemon or a set of repository servers that provide previously collected data to the alarm generator and the Performance Manager analysis product. The coda daemon uses the HTTP data communication mechanism, and the repository servers use the DCE mechanism. If both HTTP and DCE data communication mechanisms are enabled, Performance Agent uses both the coda daemon and the set of repository servers. Each data source consists of a single log file set.

The data source list that coda accesses is maintained in the datasources configuration file that resides in the /var/opt/OV/conf/perf/ directory. The data source list that the repository servers access is maintained in the perflbd.rc file that resides in the /var/opt/perf/ directory. The perflbd.rc file is maintained as a symbolic link to the datasources file.

There is a repository server for each specific data source such as scopeux log files or DSI log files. When you first start up Performance Agent after installation, a default data source named SCOPE is already configured and provides a scopeux log file set.

If you want to add other data sources, you can configure them in the datasources file. If you no longer want to view the Performance Agent or DSI log file data from Performance Manager, or process alarms for the log file, you can modify the datasources file to remove the data source and the path to the log file set. When you restart the coda daemon or the repository server, it reads the datasources file and makes the data available over datacomm linkages to analysis tools for each data source it finds. Restart coda or the repository server as described in Datasources Configuration File Format on page 48.

You can also remove the log file set if you no longer need the data. If you remove the log file set but do not remove the data source from datasources, coda or the repository server will skip the data source.

You might also choose to stop logging DSI data to a log file set but keep the coda daemon or the repository server open so you can view the historical data in Performance Manager. In this case, stop the dsilog process but do not delete the data source from the datasources file.

Datasources Configuration File Format

Each entry you place into the `datasources` configuration file represents a data source consisting of a single log file set. The entry specifies the data source name and location. Fields are case-insensitive except for the log file path name. The syntax is:

datasource=datasource_name logfile=logfile_set

- **datasource** is a keyword. **datasource_name** is the name used to identify the data source. For example, the data source name used in alarm definitions or by analysis software. Data source names must be unique. They are translated into upper case. The maximum length for a data source name is 64 characters.

- **logfile** is a keyword. **logfile_set** is the fully-qualified name identifying the DSI log file (created by the `dsilog` process, ending in `.log`), and is case-sensitive.

Following are two examples of the `datasources` file's data source entries:

```
datasource=SCOPE logfile=/var/opt/perf/datafiles/logglob
datasource=ASTEX logfile=/tmp/dsidemo/log/astex/ASTEX_SDL
```

After updating `datasources`, run the following command to make the new data sources available through coda:

/usr/lpp/perf/bin/ovpa restart server

If you are also running repository servers, run the following command to make the new datasources available through repository servers (`rep_server`):

/opt/perf/bin/mwa restart server

Note that stopping repository server processes results in any connection to Performance Manager being lost. For example, if you are drawing a graph on a data source and try to draw another graph, you will need to reselect the data source in Performance Manager and re-establish the connection when the repository server is started again.

Examine the contents of the `/var/opt/OV/log/coda.txt` file to check if the `coda` daemon was activated or for error messages.

For specific examples of configuring DSI data sources, see "Configuring Data Sources" in Chapter 4 of the *HP Performance Agent for UNIX Data Source Integration Guide.*

Parm File Configuration

The `parm` file is a text file that specifies configuration of the `scopeux` data collector including log file maximum sizes, interesting process threshold definitions, and application definitions. Comments in the file provide an overview of the various settings.

The `parm` file is provided with Performance Agent in the `/opt/perf/newconfig/` directory and is copied into the `/var/opt/perf/` directory during installation, if there is not an existing `/var/opt/perf/parm` file. For a complete description of the `parm` file and its parameters, see the "Parm File" section in Chapter 2 of the *HP Performance Agent for UNIX User's Manual*.

Defining Alarms

If you plan to use alarms to monitor performance, you need to specify the conditions that generate alarms in a set of alarm definitions in the Performance Agent `alarmdef` file. When Performance Agent is first installed, the `alarmdef` file contains a set of default alarm definitions. You can use these default definitions or customize them to suit your needs.

For detailed information about defining alarms, see Chapter 7, "Performance Alarms," in the *HP Performance Agent for UNIX User's Manual*. This chapter describes the alarm definitions syntax, how alarms work, and how alarms can be used to monitor performance.

Native Language Support

A feature of the Performance Agent `extract` and `utility` programs running on HP-UX systems is native language support (NLS).

The `extract` and `utility` programs can support the date and time formats of other native languages if you use the NLS feature of the HP-UX operating system.

If a native language is installed and selected on your system, the `extract` and `utility` programs make the following adjustments:

- Dates and times are entered and printed in the language specified in the NLS package. If a date or time is entered in an unrecognizable format, the program prompts you with an example in the correct format.

- The text for the `help` command is obtained from `extract.help` and `utility.help`. These files are located in the `/opt/perf/help/ovpa/`*$LANG* directory where *$LANG* is the language specified for this system. The default language is "`C`." If a help catalog cannot be found in the *$LANG* directory, the default help files are used.

Viewing and Printing Documents

Performance Agent software includes the standard Performance Agent documentation set in viewable and printable file formats. You can view the Adobe Acrobat format (`*.pdf`) documents online and print as needed. The ASCII text (`*.txt`) documents are printable. However, you can view a text file on your screen using any UNIX text editor such as `vi`. The documents are listed in the following table along with their file names and online locations.

Table 2   Performance Agent Documentation Set

Document	File Name	UNIX Location
HP Performance Agent for HP-UX Installation & Configuration Guide	`ovpainst.pdf`	`/opt/perf/paperdocs/ovpa/C/`
HP Performance Agent for UNIX User's Manual	`ovpausers.pdf`	`/opt/perf/paperdocs/ovpa/C/`
HP Performance Agent for UNIX Data Source Integration Guide	`ovpadsi.pdf`	`/opt/perf/paperdocs/ovpa/C/`
HP Performance Agent for UNIX & GlancePlus: Tracking Your Transactions	`tyt.pdf`	`/opt/perf/paperdocs/arm/C/`
Application Response Measurement (ARM) API Guide	`arm2api.pdf`	`/opt/perf/paperdocs/arm/C/`
HP Performance Agent HP-UX Metric Definitions	`methpux.txt`	`/opt/perf/paperdocs/ovpa/C/`
HP Performance Agent metrics list by Data Class for all operating systems	`mettable.txt`	`/opt/perf/paperdocs/ovpa/C/`

Viewing Documents on the Web

The listed documents can also be viewed on the HP Software Manuals web site at:

http://ovweb.external.hp.com/lpe/doc_serv

Select **Performance Agent** from the product list box, select the release version, select the OS, and select the manual title. Click **[Open]** to view the document online, or click **[Download]** to place the file on your computer.

Adobe Acrobat Files

The Adobe Acrobat files were created with Acrobat 7.0 and are viewed with the Adobe Acrobat Reader versions 4.0 and later. If the Acrobat Reader is not in your Web browser, you can download it from Adobe's web site:

http://www.adobe.com

While viewing a document in the Acrobat Reader, you can print a single page, a group of pages, or the entire document.

From HP-UX, you can read a .PDF file using the acroread command, if you have installed Adobe Acrobat Reader on your system. Enter the following command where /path/ is the location of the acroread command:

/path/acroread *filename.pdf*

ASCII Text Files

To print a .txt file, type:

lp -dprintername *filename*

For example,

lp -dros1234 Metrics.txt

Configuring Coda

Introduction

Coda is a light weight data collection agent for HP Operations Agent. It is a subset of Performance Agent and acts as a communication conduit for Performance Agent. All the configuration parameters are configured under the namespaces `coda` and `coda.comm` using the command `ovconfchg`.

Coda namespace

The following table lists the configuration parameters for the namespace `coda`:

Parameters in coda namespace	Description	Default Value
DISABLE_PROSPECTOR	Used to disable data collection from coda datasource in a coexistence environment of HP Operations Agent and Performance Agent	false
ENABLE_PROSPECTOR	Used to enable data collection from coda datasource in a standalone environment of Performance Agent	false
SSL_SECURITY	Used to enable secure communication from coda	NONE
RESPONSE_SIZE_LIMIT	Used to specify the maximum amount of memory allocated by coda	104857600 (100 megabytes)

DISABLE_PROSPECTOR

Use this option to specify the data collection preferences through coda, when both the HP Operations Agent and Performance Agent are installed. The default value is false. The format is as follows:

ovconfchg -namespace coda -set DISABLE_PROSPECTOR *<value>*

— true: coda will not collect data for the Coda datasource

— false: coda will collect the data for the Coda datasource

ENABLE_PROSPECTOR

Use this option to specify the data collection preferences through coda, when only Performance Agent is installed. The default value is `false`. This parameter will be ignored in a coexistence scenario. The format is as follows:

ovconfchg -namespace coda -set ENABLE_PROSPECTOR *<value>*

— `true:` coda will collect data for the Coda datasource

— `false:` coda will not collect data for the Coda datasource

To verify if coda prospector is enabled, type the following command:

ovcodautil -dumpds coda

SSL_SECURITY

Use this option to enable secure communication through coda. The default value is `NONE`. The format is as follows:

ovconfchg -namespace coda -set SSL_SECURITY *<value>*

— `NONE:` coda does not require SSL connections for either the local or remote clients

— `REMOTE:` coda requires all remote connections to use SSL

— `ALL:` coda requires all connections (both local and remote) to use SSL

⚠ Use this parameter only when certificates are present on the system. Certificates will be installed only if HP Operations Agent 8.x is present on the system.

RESPONSE_SIZE_LIMIT

Use this command to specify the maximum amount of memory allocated by the coda daemon for a query response. The default value is `104857600` (`100megabytes`). The format is as follows:

ovconfchg -namespace coda -set RESPONSE_SIZE_LIMIT *<value>*

> If the specified limit is exceeded the following error message appears:
>
> ```
> Response exceeds memory limits, use several smaller requests
> ```

Coda Communication namespace (coda.comm)

The following table lists the configuration parameters for the namespace `coda.comm`:

Parameters in coda.comm namespace	Description	Default Value
SERVER_PORT	Used to configure port settings	0
SERVER_BIND_ADDR	Used to specify bind address for the server port	INADDR_ANY
LOG_SERVER_ACCESS	If set to 'true', coda logs every access to the server providing the information about sender's IP address, requested HTTP address, requested HTTP method, and response status.	false

SERVER_PORT

You can configure the port settings of the default port used by the coda. The default value for this port is 0. If the port is set to 0, the operating system will assign the first available port number. Use the ovconfchg tool to change the port settings on the Performance Agent system. Type the command:

ovconfchg -namespace coda.comm -set SERVER_PORT *<port no>*

SERVER_BIND_ADDR

Use this option to specify the bind address for the server port. When the value is set to localhost, all the communication to coda server happen through ovbbccb. The format is as follows:

ovconfchg -namespace coda.comm -set SERVER_BIND_ADDR *<Bind address>*

LOG_SERVER_ACCESS

You can enable or disable the access to server using this option. If this option is set to true, coda records every access to the server, providing information about the senders IP address, requested HTTP address, requested HTTP method, and response status. This value typically will not be changed.

ovconfchg -namespace coda.comm -set LOG_SERVER_ACCESS *<value>*

Single Port Communication

This is the default communication method with Performance Agent 4.70. Use the following options to enable single port communication in coda:

ovconfchg -ns coda -set SSL_SECURITY REMOTE/ALL

ovconfchg -ns coda.comm -set SERVER_BIND_ADDR localhost

Multi Port Communication

Use the following option to enable multi port communication in coda:

ovconfchg -ns coda.comm -set SERVER_BIND_ADDR ""

There are two methods to find out single port and multi port communication in coda:

1   You can use this option to verify if the port is used for single port or multi port communication from the local host machine:

bbcutil -reg

For single port communication,

```
BasePath=/Hewlett-Packard/OpenView/Coda/
Protocol=HTTPS
BindAddress=localhost
Port=59814
Authentication=NONE
```

If the value returned is local host or 127.0.0.1, then it is single port communication.

For multi port communication,

```
BasePath=/Hewlett-Packard/OpenView/Coda/
Protocol=HTTPS
BindAddress=ANY
Port=381
Authentication=NONE
```

If the value returned is any other value other than local host, then it is multi port communication.

▶   **bbcutil** will be present in the *<Install Dir>*/bin directory.

2   You can use this option to verify if the port used is for singe port or multi port communication from a system other than local host:

ovcodautil -n *<hostname>* **-ping**

For example,

ovcodautil -n ovphpt4 -ping

For single port communication,

Ping of '**OvBbcCb**' at: 'http://ovphpt4:383/Hewlett-Packard/ OpenView/BBC/ping'   successful

Ping of '**Coda**' at: 'http://ovphpt4:383/Hewlett-Packard/OpenView/ Coda/' successful

For multi port communication,

ovcodautil -n ovphpt4 -ping

Ping of '**OvBbcCb**' at: 'http://ovphpt4:383/Hewlett-Packard/ OpenView/BBC/ping' successful

Ping of '**Coda**' at: 'http://ovphpt4:62581/Hewlett-Packard/OpenView/ Coda/' successful

The port numbers are different in the two outputs.

▶   For the Coda clients, all the parameters in the coda.comm namespace override the parameters defined in the bbc.http namespace.

Communication Broker namespace (bbc.cb)

The following table lists the configuration parameters for the namespace `bbc.cb`:

Parameters in bbc.cb namespace	Description	Default Value
SERVER_PORT	Used to configure port settings	383
SERVER_BIND_ADDR	Used to specify bind address for the server port	INADDR_ANY

SERVER_PORT

You can configure the port settings of the default port used by the communication broker. The default value for this port is 383. Use the `ovconfchg` tool to change the port settings on the Performance Agent system. The format is as follows:

`ovconfchg -namespace bbc.cb -set SERVER_PORT` *<port no>*

▶ If a port is already defined in the communication broker port namespace (bbc.cb.ports), the operating system assigns it as the default port and overrides the SERVER_PORT value.

SERVER_BIND_ADDR

Use this option to specify the bind address for the server port. The format is as follows:

`ovconfchg -namespace bbc.cb -set SERVER_BIND_ADDR` *<Bind address>*

Communication Broker Port namespace (bbc.cb.ports)

The following table lists the configuration parameter for the namespace `bbc.cb.ports`:

Parameter in bbc.cb.ports namespace	Description	Default Value
PORTS	Used to define the list of ports for all the communication brokers in the network that may be contacted by the applications on this host. The client applications use this as target port to communicate with the communication broker in the network. If the host name matches with one of the entries in this port settings, then the local communication broker port will be set to the port specified here.	The value of ports is not set by default.

PORTS

This configuration parameter must be same on all the nodes. To change the port number of a communication broker on a particular host, the hostname must be added to the parameter.

For example,

```
name.hp.com:8000
```

You can use an asterisk as a wild card to denote the entire network

```
*.hp.com:8001
```

You can use a comma or a semicolon to seperate entries in a list of hostnames.

For example,

```
name.hp.com:8000, *.hp.com:8001
```

In this example, all the hostnames ending with hp.com will configure their BBC communication broker to use port 8001 except the host "name", which will use port 8000. All the other ports uses the default port 383.

You can also use the IP addresses and asterisk (*) to specify the hosts.

For example,

```
15.0.0.1:8002, 15.*.*.*:8003
```

Run the following comamnd to set ports

ovconfchg -namespace bbc.cb.ports -set PORTS *<port no>*

HTTP namespace (bbc.http)

The following table lists the configuration parameter for the namespace `bbc.http`:

Parameters in bbc.http namespace	Description	Default Value
RESPONSE_TIMEOUT	Used to specify the maximum number of seconds to wait for a response	300
CLIENT_PORT	Used to specify bind port for the client requests	0
PROXY	Used to specify the proxy and port to be used for the specified hostname	The value of proxy is not set by default.

RESPONSE_TIMEOUT

Use this option to specify the maximum number of seconds to wait for a response. The default value is `300`. The format is as follows:

ovconfchg -namespace bbc.http -set RESPONSE_TIMEOUT *<value>*

CLIENT_PORT

Use this option to specify the bind port for the client requests. The default value is `port 0`. The operating system assigns the first available port. This parameter is ignored for the requests to the localhost.

▶ On the Windows system, this parameter should be defined on a large value because Windows system does not immediately release ports for reuse.

PROXY

Used to specify the proxy and port to be used for the specified hostname.

The format is as follows:

`proxy:port+(a)-(b);proxy2:port2+(a)-(b); ...;`

In this instance, the variables a and b are comma or semicolon separated lists of hostnames that apply to the proxy.

`a: for which the proxy shall be used`

`b: for which the proxy shall not be used`

The first matching proxy is chosen.

You can also use the IP addresses instead of hostnames. For example, 15.*.*.* is also valid, provided correct number of dots and colons are specified.

Glossary

This glossary contains an alphabetized list of some of the terms associated with Performance Agent.

alarm

An indication of a period of time in which performance meets or exceeds user-specified alarm criteria. Alarm information can be sent to an analysis system (such as Performance Manager) and to Operations Manager. Alarms can be identified in historical data log files using the `utility` program.

alarm generator

Handles the communication of alarm information. It consists of `perfalarm` and the `agdb` database. The `agdb` database contains a list of Performance Manager analysis nodes (if any) to which alarms are communicated, and various on/off flags that you set to define when and where the alarm information is sent.

alarmdef file

The file containing the alarm definitions in which alarm conditions are specified.

application

A user-defined group of related processes or program files. Applications are defined so that performance software can collect performance metrics for and report on the combined activities of the processes and programs.

application log file

See **logappl**.

coda daemon

A daemon that provides collected data to the alarm generator and analysis product data sources including `scopeux` log files or DSI log files. `coda` reads the data from the data sources listed in the `datasources` configuration file.

data source

Consists of one or more classes of data in a single `scopeux` or DSI log file set. For example, the Performance Agent SCOPE data source is a `scopeux` log file set consisting of global data. See also **datasources file**.

datasources file

A configuration file residing in the `/var/opt/OV/conf/perf/` directory. Each entry in the file represents a `scopeux` or DSI data source consisting of a single log file set. See also **perflbd.rc**, **coda** and **data source**.

data source integration (DSI)

The technology that enables Performance Agent to receive, log, and detect alarms on data from external sources such as applications, databases, networks, and other operating systems.

default.txt

A communications configuration file used to customize communication parameters for HP Software applications.

device

A device is an input and/or output device connected to a system. Common devices include disk drives, tape drives, DVD-ROM drives, printers, and user terminals.

device log file

See **logdev**.

DSI

See **data source integration**.

DSI log files

Log files containing self-describing data that are created by Performance Agent's DSI programs.

extract

A Performance Agent program that allows you to extract (copy) data from raw or previously extracted log files and write it to extracted log files. It also lets you export data for use by analysis programs.

extracted log file

A log file created by the `extract` program. It contains user-selected data ranges and types of data. An extracted log file is formatted for optimal access by the workstation analysis tool, Performance Manager. This file format is suitable for input to the `extract` and `utility` programs and is the preferred method for archiving performance data.

GlancePlus

GlancePlus (or Glance) is an online diagnostic tool that displays current performance data directly to a user terminal or workstation. It is designed to assist you in identifying and troubleshooting system performance problems as they occur.

global

A qualifier that implies the whole system.

global log file

See **logglob**.

interesting process

A process becomes interesting when it is first created, when it ends, and when it exceeds user-defined thresholds for CPU use, disk use, response time, and so on.

log file set

A collection of files that contain data collected from one source.

logappl

The raw log file that contains measurements of the processes in each user-defined application.

logdev

The raw log file that contains measurements of individual device (such as disk) performance.

logglob

The raw log file that contains measurements of the system-wide, or global, workload.

logindx

The raw log file that contains additional information required for accessing data in the other log files.

logproc

The raw log file that contains measurements of selected "interesting" processes. A process becomes interesting when it is first created, when it ends, and when it exceeds user-defined thresholds for CPU use, disk use, response time, and so on.

logtran

The raw log file that contains measurements of transaction data.

midaemon

A Performance Agent program that collects and counts trace data coming from the operating system kernel and stores the counters in the shared memory segment. This data is the main source of performance data for scopeux.

ovbbccb

The Operations Manager Communication Broker for HTTP(S) based communication controlled by ovcd. See also **coda** and **ovc**.

ovc

The Operations Manager controlling and monitoring process. In a standalone Performance Agent installation, ovcd monitors and controls coda and ovbbccb. If Performance Agent is installed on a system with Operations Manager for UNIX 8.x agent installed, ovcd also monitors and controls Operations Manager for UNIX 8.x processes. See also **coda** and **ovbbccb**.

ovpa script

The Performance Agent script that has options for starting, stopping and restarting Performance Agent processes such as the `scopeux` data collector, alarm generator, `ttd`, `midaemon`, and `coda`. See also the `ovpa` man page.

HP Performance Manager

Provides integrated performance management for multi-vendor distributed networks. It uses a single workstation to monitor environment performance on networks that range in size from tens to thousands of nodes.

parm file

A Performance Agent file containing the parameters used by `scopeux` to customize data collection.

perflbd.rc

A configuration file residing in the `/var/opt/perf/` directory. This file is maintained as a symbolic link to the `datasources` file. See also **datasources file.**

perfstat

A program that displays the status of all performance processes in your system.

PerfView

See **HP Performance Manager**.

process

Execution of a program file. It can represent an interactive user (processes running at normal, nice, or real-time priorities) or an operating system processes.

process log file

See **logproc**.

raw log file

Summarized measurements of system data collected by `scopeux`. See **logappl, logproc, logdev, logtran,** and **logindx**.

real time

The actual time in which an event takes place.

repository server

A server that provides data to the alarm generator and the Performance Manager analysis product. There is one repository server for each data source configured in the `perflbd.rc` configuration file. See also **data source**.

resize

Changing the overall size of a log file using the `utility` program's `resize` command.

run file

Created by the `scopeux` collector to indicate that the `scopeux` process is running. Removing the run file causes `scopeux` to terminate.

scopeux

The Performance Agent data collector program that collects performance data and writes (logs) it to raw log files for later analysis or archiving. *See also* **raw log file**.

scopeux log files

See **raw log file**.

status.scope file

Created by the **scopeux** collector to record status, data inconsistencies, and errors.

system ID

The string of characters that identifies your system. The default is the host name as returned by `uname -n`.

transaction log file

See **logtran**.

transaction tracking

The technology used in Performance Agent that allows information technology
(IT) resource managers to measure end-to-end response time of business
application transactions.

ttd.conf

The transaction configuration file where you define each transaction and the
information to be tracked, such as transaction name, performance distribution
range, and service level objective.

utility

A Performance Agent program that allows you to open, resize, scan, and
generate reports on raw and extracted log files. You can also use it to check
`parm` file and `alarmdef` file syntax, and obtain alarm information from
historical log file data.

Index

A

alarmdef file, 50

alarms, 50

authip file, 44
    examples, 45
    formatting, 45

C

changing protocols, 25

client authentication, 44

coda.log file, 48

communicating across a firewall, 36

configuring
    data sources, 47
    parm file, 49
    Performance Manager and Performance
        Agent firewall communication, 36
    Performance Manager C.03.00 and later
        with Performance Agent firewall
        communication, 36, 37
    Reporter and Performance Agent
        firewall communication, 40

D

data sources
    configuring, 47
    deleting, 47
    DSI, 47
    SCOPE, 47
    scopeux, 47

defining alarms, 50

deleting data sources, 47

disk space requirements, 12

documents
    Performance Agent documentation set,
        52
    viewing on the web, 53
    viewing PDF files from UNIX, 53

DSI data sources, 47

E

environment variables, 26

examples directory
    README, 27

extract program, 48