HP OpenView Network Diagnosis Add-On Module

User's Guide

Version: A.01.64

For HP-UX and Solaris
OpenView Operations Management Servers

Manufacturing Part Number: None
June 2004

Legal Notices

Warranty.

Hewlett-Packard makes no warranty of any kind with regard to this document, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. Hewlett-Packard shall not be held liable for errors contained herein or direct, indirect, special, incidental or consequential damages in connection with the furnishing, performance, or use of this material.

A copy of the specific warranty terms applicable to your Hewlett-Packard product can be obtained from your local Sales and Service Office.

Restricted Rights Legend.

Use, duplication or disclosure by the U.S. Government is subject to restrictions as set forth in subparagraph (c)(1)(ii) of the Rights in Technical Data and Computer Software clause in DFARS 252.227-7013.

Hewlett-Packard Company
United States of America

Rights for non-DOD U.S. Government Departments and Agencies are as set forth in FAR 52.227-19(c)(1,2).

Copyright Notices.

Trademark Notices.

Java ™ is a U.S. trademark of Sun Microsystems, Inc.

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

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

Motif is a trademark of the Open Software Foundation in the U.S. and other countries.

UNIX® is a registered trademark of The Open Group.

Support

Please visit the HP OpenView web site at:

`http://openview.hp.com/`

There you will find contact information and details about the products, services, and support that HP OpenView offers.

The support area of the HP OpenView web site includes:

- Downloadable documentation
- Troubleshooting information
- Patches and updates
- Problem reporting
- Training information
- Support program information

Contents

Contents

Contents

Contents

Contents

Contents

1 Overview

In this chapter you will find introductory information on the HP OpenView Network Diagnosis Add-On Module including:

- A description of the individual components

- Link Monitoring concept
- Network Performance monitoring
- Reporting

HP OpenView Network Diagnosis Add-On Module

The HP OpenView Network Diagnosis Add-On Module (NDAOM) provides you with detailed information on network performance and how this performance is affecting the services. New service views help to identify network failures in relation to services that are relying those network connections. Reporting on networking statistics, network status, and performance data is achieved in conjunction with HP OpenView Reporter (OVR).

Networking integration is based on level three device information with further details being derived from health utilities such as `TraceRoute`.

The following sections introduce the NDAOM tools that are integrated into HP OpenView Operations (OVO).

OVO Console Components

The HP OpenView Operations Console can be installed on either the management server system or on an alternative system. The Problem Diagnosis component is displayed in the HP OpenView Operations Console.

HP OpenView Problem Diagnosis

The HP OpenView Problem Diagnosis has a web interface used to display the network link information between two selected nodes. It shows all possible paths between those nodes and can be used to display more detail about a specific path. The Problem Diagnosis GUI is launched either in the context of event messages or service objects from the service view. In addition, there is an application in OVO and a tool in HP OpenView Operations that launches the Problem Diagnosis GUI with or without selected nodes.

At startup Problem Diagnosis requests the `probelist.xml` file from the Web Server on the managed node. This file defines which nodes can be selected as source node in the drop down menu of the Problem Diagnosis GUI. After selecting the source and target node this information is given to Problem Diagnosis on the source node. Problem Diagnosis probe sends the network health data of all detected paths to the Problem Diagnosis GUI which is then made available as a list. To retrieve detailed information of a particular path, the operator selects one of the discovered paths from the GUI.

OVO Management Server Components

There are two essential components installed on the OVO management server:

- ovnwlinkmon
- tuple database

ovnwlinkmon

The links to be monitored are stored in conjunction with the corresponding affected services in the tuple database by `ovnwlinkmon`.

`ovnwlinkmon` also triggers the download of configuration data and the NDAOM subagent to the managed nodes. The NDAOM subagent is NOT registered as a subagent on the management server. `ovnwlinkmon` is the only way to deploy the NDAOM subagent to the managed node and then the NDAOM subagent is automatically registered as a subagent.

Tuple Database

The tuple database lists all nodes to which the probe was deployed and to which the probe is to be deployed.

OVO Managed Node Components

There are four essential components installed on OVO managed nodes:

- OVO Agent
- NDAOM Templates and Policies
- Problem Diagnosis Probe
- NDAOM Monitor

OVO Agent

The standard OVO agent. This is configured by the OVO management server as normal.

NDAOM Templates and Policies

The NDAOM templates and policies are stored in the local template database. They must be deployed to the managed nodes using the OVO template deployment mechanism. The NDAOM subagent is NOT registered as a subagent on the management server and can only be deployed via `ovnwlinkmon`.

Problem Diagnosis Probe

The main purpose of the network probe is to provide network connection information for the connection to a specified node and information about the network path to this node.

The Problem Diagnosis probe sends events, for example, `Path to Node B is down`, and performance data, for example, `network delay times on the path to Node B`, to the appropriate subscribers.

Thus the network probe has to serve as interface between the monitoring program and the network commands, such as `traceroute`, `ping` or `tcp-echo`. The network probe provides an API to the monitoring program that allows:

- registering for events like `Network path to node B changed` and getting the events in a structured data format when the events occur

or

- triggering an action, such as `traceroute`, and getting the results of this action back in a structured data format.

Additionally, the network probe can return relevant health metrics and network performance metrics that it has collected for the specified network connections.

The Problem Diagnosis probe is a daemon process running permanently as a subagent of the OVO Agent. Its functionality is accessed via TCP socket connections to the Problem Diagnosis probe. It is distributed from the OVO management server via `ovnwlinkmon`.

At startup the Problem Diagnosis probe reads its configuration file, `npprobe.conf`. This file includes information on the destination nodes to which connections are to be monitored and the polling interval for each destination node.

After the startup phase the Problem Diagnosis probe repeatedly polls information about the network path to each destination node defined in the configuration file. The interval between two polls is defined for each destination node by the polling interval. The Problem Diagnosis probe stores the information gathered during these polls and uses it to generate statistical data.

NDAOM Monitor

The NDAOM Monitor is the instance that serves as the interface between the network probe and the OVO Agent. The NDAOM Monitor consists of two parts:

- NW Monitor

  The NW Monitor is generated by OVO as monitor template with predefined threshold values. It just waits for `opcmon` values that it gets from the `ovnwmonitor`, compares these values against the thresholds that it is given and sends an `opcmsg` to the OVO management server when the thresholds are exceeded.

  The user can customize the predefined threshold value. In this case the user must deploy the monitor template again to submit the new threshold values.

- ovnwmonitor

  The `ovnwmonitor` is an external monitor. It runs permanently as daemon process and does not need to be triggered by the NW Monitor.

  `ovnwmonitor` manages the following tasks:

  — Checks for probe availability.

  — Checks for local tuple DB content.

  — Subscribes to the Problem Diagnosis probe for events and performance data.

  — Processes the events or performance data that the Problem Diagnosis probe sends.

  The `ovnwmonitor` is the main process of the NDAOM on the managed node. It ensures that the Problem Diagnosis probe monitors all destination nodes specified in the local tuple database. The `ovnwmonitor` relies on a local tuple database and ensures that all destination nodes contained in the local tuple database are also specified in the Problem Diagnosis probe configuration file complete with the associated polling interval.

  Initialization is completed by subscribing to the Problem Diagnosis probe for events and performance data. This establishes two TCP connections to the local Problem Diagnosis probe.

  After the initialization the `ovnwmonitor` keeps waiting for input on one of the two TCP connections to the Problem Diagnosis probe. If the Problem Diagnosis probe sends a message then the `ovnwmonitor` checks if the destination node concerned by this message is contained in the local tuple database. If so, then this message gets processed.

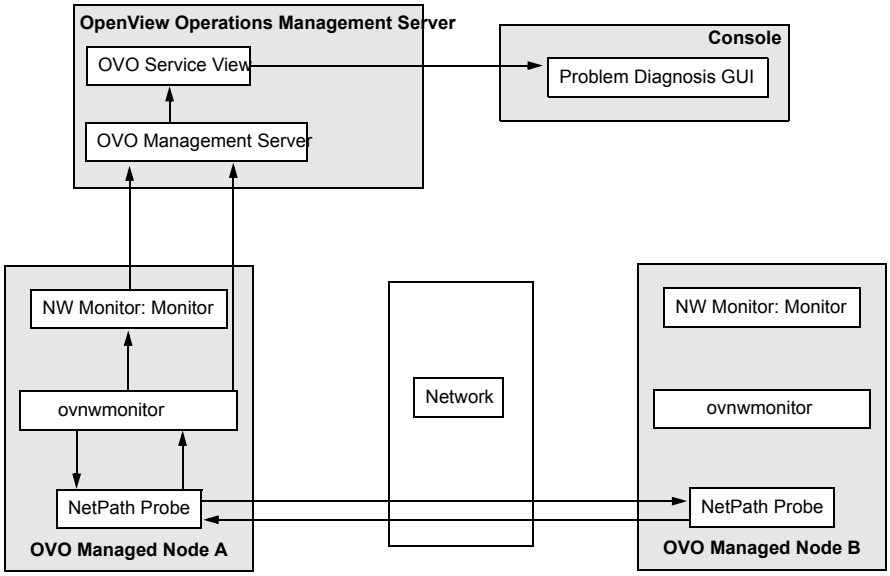
  If it is an event message, for example *Network path to node B changed", "Network path to node B down*, then the `ovnwmonitor` sends an `opcmsg` to Services concerned on the OVO management server.

  If it is a performance data message then the `ovnwmonitor` picks the relevant performance values of the message and generates `opcmon` messages from them to be processed by the NW Monitor. If the NW Monitor finds out that these values exceed one of its thresholds then it sends a corresponding `opcmsg` to the OVO management server.

Link Monitoring Concept

The link monitoring concept is the mechanism that is used to do network link health checking. The following figure illustrates how this concept works. Nodes A and Node B run OVO agents and have an `ovnwmonitor` that subscribes to the Problem Diagnosis probe to get new event or performance information about the connection between the nodes.

Node B needs to be a managed node only if both directions of the connection need to be monitored. Probes would then be required on both systems.



The following are the possible scenarios:

1. The Problem Diagnosis probe sends the event message *Network path has changed* for the path to the remote node B that is contained in the tuple DB. The `opcmsg`, *Network path to node B has changed*, is sent to the OVO management server.

2. The Problem Diagnosis probe sends the event message *Network path down* for the path to a remote node B that is contained in the tuple DB. The `opcmsg`, *Network path to node B is down*, is sent to the OVO management server. In this case, the network operator would launch the Problem Diagnosis GUI to get latest path information.
3. The Problem Diagnosis probe sends the event message *Network path up* for the path to a remote node B that is contained in the tuple DB. The `opcmsg`, *Network path to node B is up*, is sent to the OVO management server. In this case, the network operator would launch the Problem Diagnosis GUI to get latest path information.
4. The Problem Diagnosis probe sends performance data to `ovnwmonitor` for the path to the remote node B that is contained in the tuple DB. `ovnwmonitor` sends an `opcmon` call to the `NWMonitor`. If a specific round-trip time is exceeded, the `NWMonitor` sends the `opcmsg`, *Round-trip time exceeded for node C on path to node B* to the OVO management server.

Network Performance Monitoring

The NDAOM uses three sources of data to monitor network performance:

- OVO Performance Agent which supplies performance data of systems hosting a Problem Diagnosis probe.
- Problem Diagnosis probe which gathers data about the network link between two nodes. Data is retrieved from the performance port of the probe.
- Embedded Performance Component.

Reporting

The NDAOM can create reports from HP OpenView Operations and Network Node Manager data in addition to the performance data from Problem Diagnosis probe. The reports include:

- Customized reports for the NDAOM that are part of the latest release of HP OpenView Reporter.
- Network Node Manager reports.
- Network related reports available from the OVO Performance Agent.
- Performance data analysis of Problem Diagnosis probe (via OVP data gathering).

2 Installing the HP OpenView Network Diagnosis Add-On Module

In this chapter you will find information on:

- Prerequisites for installing the NDAOM
- Installing the NDAOM on the system where the HP OpenView Operations software is installed
- Uninstallation steps

Installation Prerequisites

Before starting the NDAOM installation process, make sure that the following requirements are met:

OVO Console System

The OVO Console can be installed on the same system as the management server, or on another suitable system in the network. The prerequisites are as follows:

- HP OpenView Operations Java™ GUI, version A.07.x or A.08.00 is installed and configured on an appropriate Windows NT, Windows 2000 or UNIX® system.
- One of the following (or later) web browsers with a Java Plug-In:
  - Netscape Navigator 4.72
  - Microsoft® Internet Explorer 4.0.

Management Server System

- HP OpenView Operations, version A.07.x or A.08.00 is installed and configured on a UNIX system running one of the following operating systems: HP-UX 11.0 and 11.11, Solaris 2.6, 7 or 8.
- On OVO management servers running under Sun Solaris operating systems (Solaris 2.6, 7, 8), the Sun Workshop Compilers Bundled libC installation package `SUNWlibC` must be installed so that the library `libCstd.so.1` is in the standard library path.
- HP OpenView Problem Diagnosis, version A.01.00 or A.01.10 is installed on the HP OpenView Operations management server system. The installation of HP OpenView Problem Diagnosis must at least include the OVO Problem Diagnosis Probe (selected via Customized Installation during the installation of Problem Diagnosis).
- If HP OpenView Network Node Manager is being used, this must also be installed on the same system as the HP OpenView Operations management server. The supported versions of HP OpenView Network Node Manager are A.06.20 or higher.

- If HP OpenView Service Navigator is being used, this must also be installed on the same system as the HP OpenView Operations management server. The supported versions of Service Navigator are A.07.x or A.08.00.
- 25 MB disk space on the HP OpenView Operations management server system to install the integration components.
- Perl 5.004 or higher must be installed on the OVO management server system in order for the HP OpenView Performance Manager (OVPM) integration to correctly function. There must be a link from `/usr/bin/perl` to the Perl interpreter.
- DSI2DDF package, version A.01.10 or higher is installed on the OVO management server and deployed to the managed nodes.

Managed Node Systems

- HP OpenView Performance Agent is installed and configured.
- On OVO managed nodes running under Sun Solaris operating systems (Solaris 2.6, 7, 8), the Sun Workshop Compilers Bundled libC installation package `SUNWlibC` must be installed so that the library `libCstd.so.1` is in the standard library path.
- All managed nodes must meet the prerequisites of the Problem Diagnosis Probe. This includes an installed Java Runtime Environment version 1.2.2.06 or higher.
- 10 MB disk space on the managed node is required for the NDAOM software installation.

The Problem Diagnosis Probe and the Network Monitoring program is available for the following platforms:

- Windows NT
- Windows 2000
- HP-UX 11.0 and 11.11
- Sun Solaris 2.6, 7 and 8

NOTE All managed nodes that are to produce and collect performance data using NDAOM must have the HP OpenView Performance Agent installed before NDAOM deployment to the node is made.

Table 2-1 Supported OVO Managed Node Software

Platform	Operating System	Software Required on OVO Managed Nodes
HP-UX	11.00, 11.11	Performance Agent
Solaris	2.6, 7, 8	Performance Agent
Windows	NT 4.0 with Service Pack 5	Performance Agent
	2000	Performance Agent

Reporting

The NDAOM can generate reports through HP OpenView Reporter, version 3.0 based on Crystal Reports 8.5. Ensure that HP OpenView Reporter Reporter and the supplied InstallShield package is installed on a dedicated Windows system.

Data from the following databases can be used to generate reports:

- SOLID Embedded Engine
- SQL format data such as:
  - — Microsoft SQL Server
  - — Oracle®
- Performance Agent DSI
- Embedded Performance Component

Table 2-2 Product Support Matrix

Product Name	Product Version	NDAOM A.01.62	NDAOM A.01.64
Embedded Performance Component	A.01.00	✔	✔
MeasureWare (MWA)	All releases	✔	✔
Problem Diagnosis	A.01.00	✔	✔
Problem Diagnosis	A.01.10	✔	✔
OV Performance Manager	A.02.00	✔	✔
PerfView	All releases	✔	✔
OV Reporter (only in combination with MWA)	A.02.00	✘	✘
OV Reporter (in combination with the Embedded Performance Component or MWA)	A.03.00	✔	✔

Installation on the HP OpenView Operations Management Server System

Installing the HP OpenView Network Diagnosis Add-On Module on the HP OpenView Operations management server system is handled via HP OpenView Software Distributor. The steps are detailed in “How Is the NDAOM Installed on the OVO Management Server?” on page 34.

Please follow the instructions given in this chapter to install the NDAOM on a system where HP OpenView Operations is already installed.

What Does NDAOM Give You?

Installation of NDAOM adds the following features to HP OpenView Operations:

- A virtual node, `NDAOM_Infrastructure`.
- A node group, `NDAOM`.
- A message group, `NDAOM`, that contains all messages generated by `NDAOM`.
- An application group `NDAOM-Admin` is created that contains applications added by the NDAOM used to administer the add-on module itself.
- An application group, `NDAOM-Reports`, is created to get HP OpenView Reporter reports about network statistics.
- An application group, `NDAOM`, is created that contains applications added by the NDAOM.
- A user profile `NDAOM-Admin-Profile`.
- A user profile `NDAOM-Operation-Profile`.
- A template group `NDAOM` is created that contains the following policies:

— **NDAOM Templates**

The NDAOM templates should be deployed to each node that is to supervise a network connection relevant to a service to forward events to the OVO message browser.

What Files Are Installed?

The following files are installed on the HP OpenView Operations management server and the managed nodes.

OVO Management Server

The NDAOM files are placed in the following directories:

`/opt/OV/ndaom/bin`	Binary files
`/opt/OV/lib/nls/<locale>`	Message Catalog file (`ndaom.cat`) for NDAOM executable files
`/opt/OV/ndaom/doc`	Documentation
`/opt/OV/ndaom/contrib`	Service Reports for NDAOM
`/etc/opt/OV/ndaom/conf`	Configuration files
`/var/opt/OV/ndaom/tmp`	Temporary files
`/var/opt/OV/ndaom/log`	Log and trace files
`/var/opt/OV/share/ndaom`	Global tuple database
`/var/opt/OV/share/databases/subagent/ndaom`	Subagent files to be deployed

HP-UX and Solaris Managed Nodes

The NDAOM files are placed in the following directories:

`/opt/OV/subagent/ndaom`	Deployed subagent files

/opt/OV/ndaom/bin	Binary files
/opt/OV/lib/nls/<locale>	Message catalog file (ndaom.cat) for NDAOM executable files
/etc/opt/OV/ndaom/ddf	Configuration and specification files for Dynamic Data Feed
/etc/opt/OV/ndaom/conf	Configuration files
/var/opt/OV/ndaom	Managed node tuple database
/var/opt/OV/ndaom/tmp	Temporary files
/var/opt/OV/ndaom/log	Log and trace files
/var/opt/OV/ndaom/ddf	Log files for OVPM and OVR integration via Embedded Performance Component

Directories used by the NDAOM via the OVO agent mechanisms are:

- /var/opt/OV/bin/OpC/monitor
- /var/opt/OV/bin/OpC/cmds
- /var/opt/OV/bin/OpC/actions

Microsoft Windows NT and Windows 2000 Managed Nodes

Default directories used by the Add-On Module are within the directory:

OVO for Win managed node:
\Program Files\Hewlett-Packard\OVEnterprise

OVO managed node:
\usr\OV

For example:

C:\usr\OV\:

subagent\ndaom	Deployed subagent files
ndaom\bin	Binary files

nls\<locale>\<locale>	Message catalog file (ndaom.cat)
ndaom\conf	Configuration files
ndaom	Managed node tuple database
ndaom\tmp	Temporary files
ndaom\log	Log and trace files
ndaom\ddf	Files for OVPM and OVR integration

Sub-directories used by the NDAOM via the OVO agent mechanisms are found under either:

OVO for Win managed node:
C:\Winnt\Hewlett-Packard\OVEnterprise\Agent\<GUID>:

OVO managed node:
C:\usr\OV\bin\OpC

\monitor	Monitor scripts
\cmds	Scripts triggered by applications
\actions	Scripts triggered by actions

This is the default if the agent is distributed from the OVO management server.

How Is the NDAOM Installed on the OVO Management Server?

Installation of the HP OpenView Network Diagnosis Add-On Module for HP OpenView Operations is divided into three parts:

- Install the Problem Diagnosis software.
- Install the NDAOM software on the management server.
- Install the NDAOM components on the managed node.

Installing the Problem Diagnosis Software on the HP OpenView Operations Management Server System

The first step is for you to install the Problem Diagnosis server on a system of your choice. The easiest option is to use the OVO management server system.

NOTE

You may choose not to install the Problem Diagnosis server at all. NDAOM can work without the Problem Diagnosis server, provided that the OVO Problem Diagnosis Probe is installed. However, without the Problem Diagnosis server the graphical user interface of the HP OpenView Problem Diagnosis with which NDAOM integrates will not be available to you.

1. Make sure that the prerequisites for the Problem Diagnosis product have been fulfilled. Please refer to the installation documentation provided with the Problem Diagnosis product.
2. Start the installation of the Problem Diagnosis software.
3. When requested for the type of installation to make, select **Customize**.
4. Mark the following software components for installation:
   - **Problem Diagnosis Server**
   - **OVO Problem Diagnosis Probe**

   Do not mark the Problem Diagnosis Probe for installation.
5. Complete the installation.
6. Make sure that the Problem Diagnosis Server is running. If it is not, start it with the appropriate command on the host system:

   UNIX `/opt/OV/pd/app_server/bin/ovpdstart`

   Windows **Start →Programs →HP OpenView →Problem Diagnosis →PD Server-Start**

   If you have installed the Problem Diagnosis Server on a system other than the OVO management server system or if you have chosen not to install the HP OpenView Problem Diagnosis server at all, perform the following additional installation tasks on the OVO management server system:

7. Start the installation of the Problem Diagnosis software on the OVO management server system.
8. When requested for the type of installation to make, select **Customize**.
9. Only mark the following software component for installation:

   `OVO Problem Diagnosis Probe`
10. Complete the installation.

Mounting the CD-ROM on HP-UX

To mount the CD-ROM on HP-UX, complete the following steps:

1. Login as user root.
2. Set the umask of user root by entering:

   `umask 027`
3. Create a directory to mount the CD-ROM:

   `mkdir /<mount_point>`

   For example: **`mkdir /cdrom`**
4. Insert the HP OpenView Smart Plug-ins for OVO/HP-UX CD-ROM into the disk drive and mount it as user root by entering:

   `mount -r -F cdfs /dev/<cdrom_drive_name> /<mount_point>`

   For example, for a local CD-ROM, you might enter:

   `mount -r -F cdfs /dev/dsk/c0t2d0 /cdrom`

   You can also run SAM and mount the CD-ROM to a specific path in the `Disks and File Systems` window.

Mounting the CD-ROM on Solaris

Insert the HP OpenView Smart Plug-ins for OVO/Solaris CD-ROM into the CD-ROM drive. The CD-ROM is automatically mounted (and unmounted) on Sun Solaris systems.

Installing the NDAOM on the HP OpenView Operations Management Server System

To install NDAOM on the OVO management server:

1. Login to the OVO management server system as user **root**.
2. Execute the command appropriate for your operating system:
   - HP-UX

     swinstall -s /cdrom/OV_DEPOT/11.0HPUX.sdtape AOM-ND-OVO-HP
   - Sun Solaris

     swinstall -s /cdrom/OV_DEPOT/SOLARIS.sdtape AOM-ND-OVO-SOL

   This installs the NDAOM software and also configures it for the HP OpenView Operations installation.

NOTE During the installation process, the OVO management server processes are automatically stopped and restarted (`opcsv -stop`/`opcsv -start`).

NOTE If no product is specified, (e.g., `AOM-ND-OVO-HP`), swinstall will start with an interactive GUI (on HP-UX systems *only)*.

NOTE Filesets remain the same for all SPIs as written in the installation instructions of each SPI. Only the depot name and location where the depot is located *may* change. Specify the following depot when entering the software install command:

`11.0HPUX.sdtape` or **`SOLARIS.sdtape`**

TIP To unmount the CD-ROM, enter **`unmount /<mount_point>`**, and remove the CD-ROM from the disk drive.

3. Configure the NDAOM as user `root` by modifying the `ndaom.cfg` file as follows and adapt the entries for all variables to fit your environment (`BROWSER, PD_SERVER, PD_SERVER_IP, PD_SERVER_PORT`, `PERFMGR_SERVER`). See appendix A for a full description of the variables.

   Open the file:

   `/etc/opt/OV/ndaom/conf/ndaom.cfg`

   and edit it so that it reflects the following requirements:

   - An entry is required which defines the command to start your web browser. For example:

     `BROWSER=/opt/netscape47/netscape`

   - An entry is required which defines on which network node the OV Performance Manager installation resides. For example:

     `PERFMGR_SERVER=bug.London.mycom.com`

   - An entry is required which defines the name of the OVO management server system where the Problem Diagnosis Server is installed. For example:

     `PD_SERVER=bug.London.mycom.com`

   - An entry is required which defines the IP address of the OVO management server system where the Problem Diagnosis server is also installed. For example:

     `PD_SERVER_IP=16.216.111.55`

   - An entry is required which defines the port number that has been assigned to the Problem Diagnosis server. For example:

     `PD_SERVER_PORT=9085`

     The default value is 9085.

4. NDAOM installs two user profiles in the user profile bank: `NDAOM-Admin-Profile` and `NDAOM-Operator-Profile`. Ensure that you assign one of them to the current user.

5. Start the following command on the OVO management server system:

   `/opt/OV/ndaom/bin/update_templates.sh`

This shell script updates the NDAOM templates that are necessary for the Problem Diagnosis Server integration and uploads them again to the OVO management server.

Installing the NDAOM on the HP OpenView Operations Managed Node Systems

- Make sure that a Java Virtual Machine version 1.2.2.04 or higher is installed on every target managed node.

NOTE

A problem that is sometimes experienced on UNIX nodes is that the Java executable file is not found by the subagent installation process. This is because the path to the Java executable is not contained in the PATH environment variable of this process.

In order to avoid this problem, set a link from the Java executable to `/usr/bin`:

```
ln -s <path to Java executable file> /usr/bin/java
```

- Make sure that the Problem Diagnosis server is running before you start the NDAOM subagent deployment/installation. If the server is not running, start it with:

  UNIX `/opt/OV/pd/app_server/bin/ovpdstart`

  Windows **Start → Programs → HP OpenView → Problem Diagnosis → PD Server-Start**

  If the Problem Diagnosis server is not running during the NDAOM subagent deployment/installation the newly installed Netpath Probe may not be able to register at the Problem Diagnosis server and so the Problem Diagnosis server may not know this Netpath Probe.

Installing the NDAOM Actions, Commands, Templates and Monitors

Use `ovnwlinkmon -deploy` to add the necessary templates, actions, commands, and monitors to the managed nodes. See “ovnwlinkmon -deploy” on page 82 or “Deploy Network Connections” on page 96.

Installing the NDAOM Subagent

The NDAOM subagent consists of the Problem Diagnosis Probe, ovnwmonitor, and ovnwpdc. It is deployed from the OVO management server system using ovnwlinkmon.

1. Use the ovnwlinkmon program with the -add option to add the network connections to be monitored to the global tuple database.

   ovnwlinkmon is located in: /opt/OV/ndaom/bin

   For a detailed description of how to use ovnwlinkmon -add, see “ovnwlinkmon -add [-NoNewObject]” on page 76.

2. View the network connections in the global tuple database using the command:

   ovnwlinkmon -list

3. Deploy the NDAOM subagent to the managed nodes that are listed as Source nodes in the global tuple database using the command:

   ovnwlinkmon -deploy

   For a detailed description of how to use ovnwlinkmon -deploy, see “ovnwlinkmon -deploy” on page 82.

4. Deployment of the NDAOM subagents via ovnwlinkmon -deploy usually takes a few minutes. The NDAOM subagent packages are deployed to the managed nodes via opctranm and a checkinstall script checks the configuration of the managed nodes. A background process then starts the subagent installation and registers the NDAOM agent executable files with the OVO agent. Please wait until you receive an Installation Success message in the OVO message browser on the management server.

If the NDAOM subagent installation fails (for example with the error message:

Deployment of the subagent to ... via opctranm failed. ... opctranm output was /tmp/install.success: No such file or directory)

check the installation log file on the managed node system:

Unix /tmp/install_nwagt.log

Windows \TEMP\install_nwagt.log

Installing the NDAOM Reports

To install the NDAOM reports:

1. Copy the executable file to the Windows NT 4.0/2000 system where HP OpenView Reporter is installed:

   `/opt/OV/ndaom/contrib/ServiceReportsForNDAOM.exe`

   which is part of the NDAOM package and follow the on-screen instructions.

2. Run the executable file on the Windows NT 4.0/2000 system where HP OpenView Reporter is installed:

   `ServiceReportsForNDAOM.exe`

   and follow the on-screen instructions.

3. At the end of the installation process select:

   `Run Service Reporter configuration script.`

   The installation automatically:

   - Copies all report template files to the reporter data directory
   - Creates the NDAOM metrics list
   - Creates the discovered systems group NDAOM
   - Assigns all systems on which the Performance Agent and NDAOM are running to the group NDAOM.

After installation you can manually start the following tasks:

- Start gathering metrics ...
- Start generating reports ...
- Show Reports ...

or wait for the automatic start done by the HP OpenView Reporter scheduler. No manual configuration is necessary.

NOTE Reporting performance data for the first time is time consuming. It will take between some hours to over one day until you are returned any results.

Upgrading from Earlier Versions

If you are upgrading NDAOM from version A.01.60 or A.01.62 to any higher version, you must first uninstall the previous version. Use the following procedure before you attempt to install the current version of NDAOM.

1. Find all the NDAOM services by executing the following command:

   ```
   opcservice -list -all | grep “Net:”
   ```

2. Manually remove all NDAOM services by executing the following command:

   ```
   opcservice -remove Net:*
   ```

3. Find all the NDAOM actions by executing the following command:

   ```
   opcservice -list -all | grep “NW:”
   ```

4. Manually remove all the NDAOM actions by executing the following command:

   ```
   opcservice -remove NW:*
   ```

5. Follow the uninstall instructions in the section called “Uninstalling the NDAOM on the HP OpenView Operations Management Server System” on page 44.

Uninstallation Tasks

Uninstallation of the HP OpenView Network Diagnosis Add-On Module for HP OpenView Operations is divided into two parts:

- Uninstall the NDAOM components from the managed node.
- Uninstall the NDAOM software on the management server.

Uninstalling the NDAOM on the HP OpenView Operations Managed Node Systems

Although the managed node uses the subagent mechanism for the NDAOM, the nodes are not registered at the management server. Thus uninstallation of the probe cannot be done with the HP OpenView Operations subagent de-installation mechanism.

Do one of the following:

- Use the `Remove NDAOM Subagent` application from the application group `NDAOM-Admin for UNIX` or `NDAOM-Admin for Windows` on the management server.
- Use the `ovnwlinkmon -remove_sa` command line call (see page 82 for further details).
- Call the `remove_nwagt` script from the command line directly on the managed node:

  HP-UX/Solaris **`opt/OV/subagent/ndaom/remove_nwagt.sh`**

  Windows NT **`c:\usr\OV\subagent\ndaom\remove_nwagt.bat`**

NOTE

After the uninstallation is complete, the following entities remain in the node:

- NDAOM tools
- NDAOM policies
- NDAOM node group

Uninstalling the NDAOM on the HP OpenView Operations Management Server System

To uninstall the NDAOM from the OVO management server, carry out the following steps:

1. Log in to the OVO management server system as user **root**.
2. The software can be removed from the management server using the command appropriate for your operating system:

HP-UX	`# swremove AOM-ND-OVO-HP`
SUN Solaris	`# swremove AOM-ND-OVO-SOL`

3 Integrating HP OpenView Problem Diagnosis into OVO

This section describes how the NDAOM integrates into the HP OpenView Operations user interface. Service views are covered in

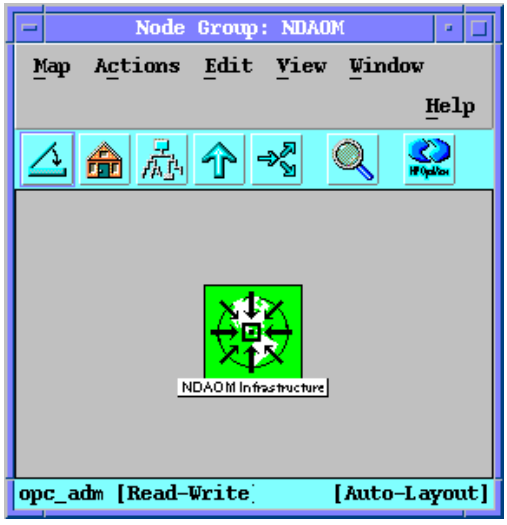
Chapter 4, “Integrating HP OpenView Network Diagnosis Add-On Module into OVO Service View,” on page 73.

Node Bank

The installation of the NDAOM adds a virtual node `NDAOM_Infrastructure` for external events to the list of managed nodes. This node gets messages for the monitored network links. This makes it possible to assign the messages to a service representing a network connection in the Service View.

Attribute	Value
Label	NDAOM Infrastructure
Network Type	Others
Node Pattern	NDAOM_Infrastructure
Type of Node	Message allowed

Figure 3-1 The NDAOM_Infrastructure Virtual Node



Node Group Bank

A new node group, NDAOM, is incorporated containing all nodes that have a probe installed.

This node group can be used within the template assignment and the responsibility matrix of an operator. It is also used in the user profile.

Message Group Bank

Two message groups are used:

- The standard group `Network` that is available within HP OpenView Operations.
- The new message group, `NDAOM`, contains messages generated by the NDAOM due to unexpected behavior of the environment, for example, Cannot find probe on node.

Application Group Bank

The NDAOM installation add the following application groups to the application bank:

- **NDAOM**
- **NDAOM-Admin**
- **NDAOM-Reports**

NDAOM Application Group

The NDAOM application group contains the following applications that are used with the NDAOM in a runtime environment.

- **Error Rate**

  Error rate of interfaces per node.

  Executed On Management Server

  Parameter $OPC_NODES

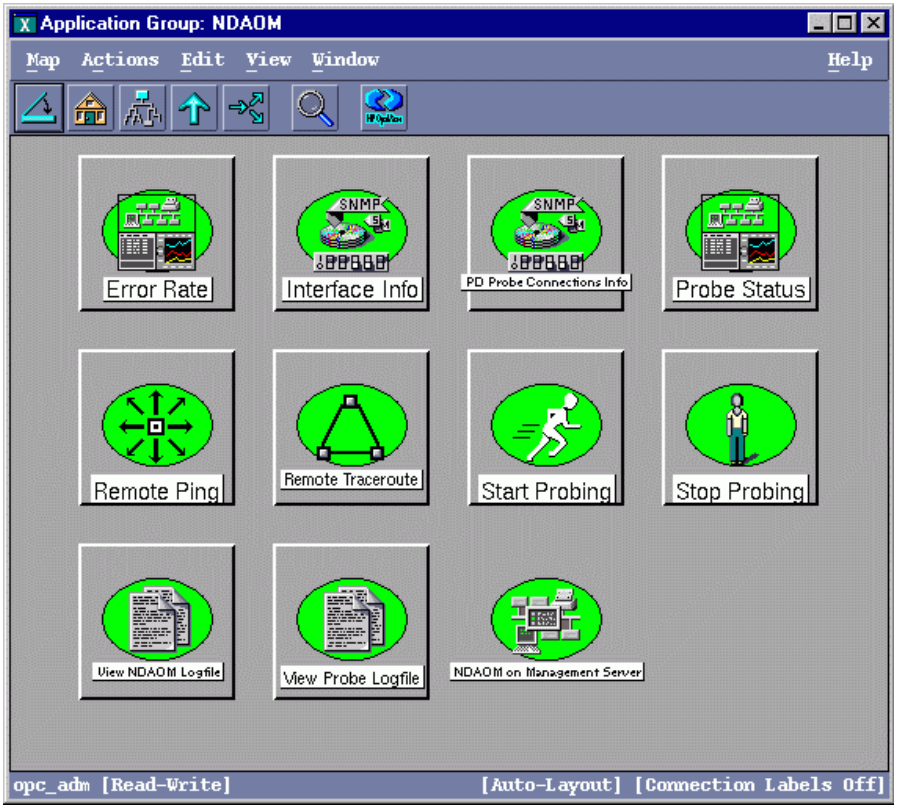
- **PD Probe Connections Info**

  View network connections for connections currently monitored by the PD Probe.

  Executed On Managed Node

  Parameter $OPC_NODES

Figure 3-2 NDAOM Application Group



- **Interface Info**

  Show processed SNMP interface info per node.

  Executed On Management Server

  Parameter `$OPC_NODES`

- **Probe Status**

  Get the status of the NDAOM subagent and the NetPath Probe (stopped or running).

  Executed On Managed Node

  Parameter `$OPC_NODES`

- **Remote Ping**

  Execute a ping between two selected nodes.

  Executed On Managed Node

  Parameter $OPC_NODES

- **Remote Traceroute**

  Execute a manual Traceroute between two selected nodes.

  Executed On Managed Node

  Parameter `$OPC_NODES`

- **Probing**

  Start the NDAOM subagent (`PD Probe`, `ovnwmonitor`, `ovnwpdc`) on the managed node.

  Executed On Managed Node

  Parameter `$OPC_NODES`

- **Stop Probing**

  Stop the NDAOM subagent and cease probing network connections.

  Executed On Managed Node

  Parameter `$OPC_NODES`

- **View NDAOM Logfile**

  View the NDAOM logfile (log output of `ovnwlinkmon`, `ovnwmonitor`, `ovnwpdc`).

  Executed On Managed Node

  Parameter `$OPC_NODES`

- **View Probe Logfile**

  View the NetPath Probe logfile.

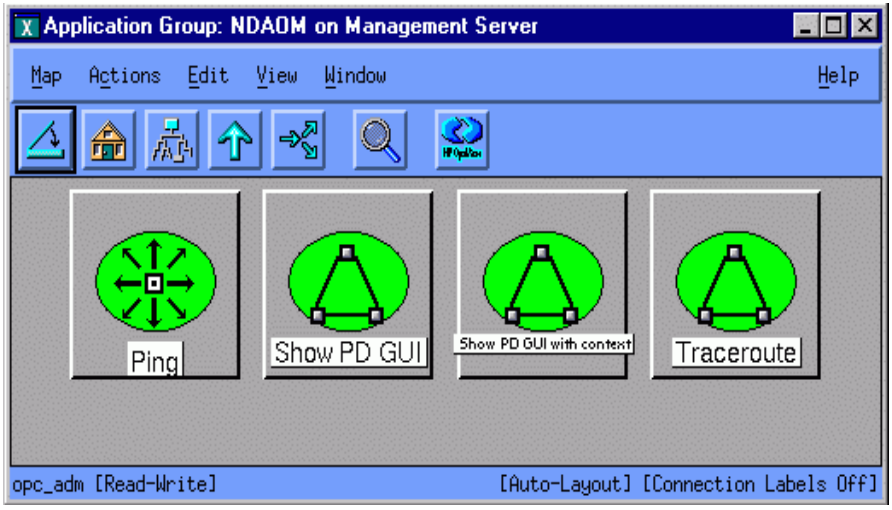
  Executed On Managed Node

  Parameter `$OPC_NODES`

NDAOM on Management Server

The `NDAOM on Management Server` application group contains the following applications that are used with the NDAOM in a runtime environment.

Figure 3-3 NDAOM on Management Server Application Group



- **Ping**

  Execute a ping between server node and selected node(s).

  Executed On Management Server System

  Parameter `$OPC_NODES`

  `-n <count>`

  as parameter that can be customized

- **Show PD GUI**

  Starts the Problem Diagnosis GUI as a standalone tool.

  Executed On Management Server System

  Parameter None

- **Show PD GUI with context**

  Starts the Problem Diagnosis GUI in the context of a selected Network Path Message.

  You must assign the NDAOM applications to the administrator (or any operator) from the User Bank Window in OVO. Only then will the NDAOM applications appear as part of the Java GUI. Make sure that the following file has the entry for the path of the browser:

  `/etc/opt/OV/ndaom/conf/ndaom.cfg`

  Select the network path messages in the Java GUI and execute the application. For example, a network path message might begin with "Response time between..." or "Network path between..."

NOTE This application works only on the Java GUI of OVO 7.0.

Executed On	Management Server System
Parameter	None

- **Traceroute**

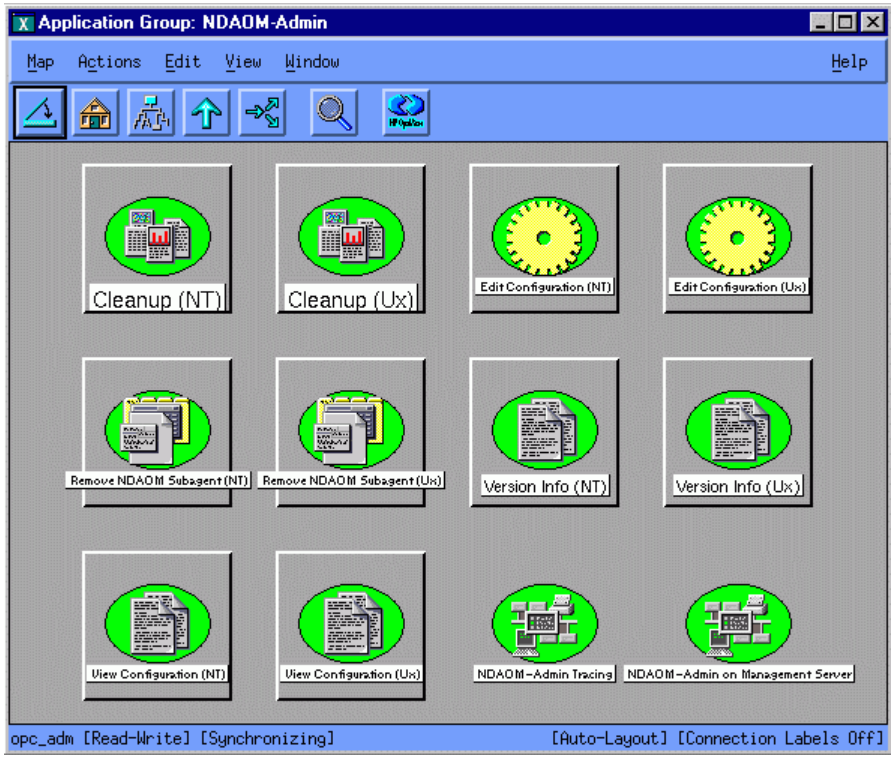
  Execute a Traceroute between server node and selected node(s).

Executed On	Management Server System
Parameter	`$OPC_NODES` as parameter that can be customized

NDAOM-Admin Application Group

The `NDAOM-Admin` application group contains applications that are used to do NDAOM configuration tasks, for example, modifying the tuple DB with the editor.

Figure 3-4 NDAOM-Admin Application Groups



Each application is available for execution on:

- Windows NT managed nodes
- UNIX managed nodes

The user must select the appropriate version.

- **Cleanup (Win/UX)**

  Remove the NDAOM and PD Probe history data by resetting trace and log files and removing temporary files.

  Executed On Managed Node

  Parameter `$OPC_NODES`

- **Edit Configuration (Win/UX)**

  Modify the NDAOM configuration file.

  Executed On Managed Node

  Parameter `$OPC_NODES`

  Not available from the OVO Java GUI (input/output windows not supported).

- **Remove NDAOM Subagent (Win/UX)**

  Completely removes the NDAOM subagent from the selected node.

  Executed On Managed Node

  Parameter `$OPC_NODES`

- **View Version Info (Win/UX)**

  View the version information of the installed NDAOM and PD products.

  Executed On Managed Node

  Parameter `$OPC_NODES`

- **View Configuration (Win/UX)**

  View the NDAOM configuration file (`ndaom.cfg`).

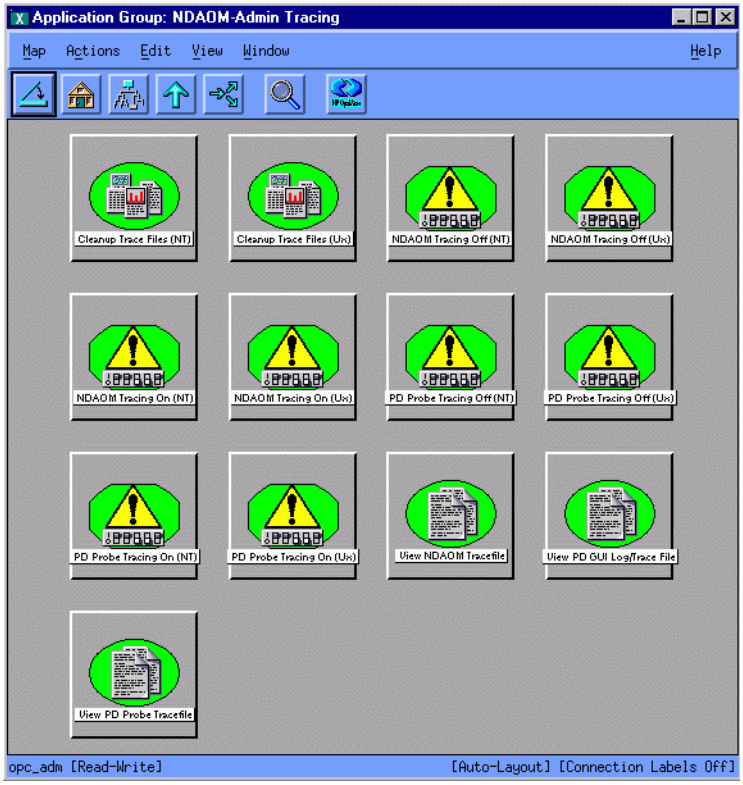
  Executed On Managed Node

  Parameter `$OPC_NODES`

NDAOM-Admin Tracing

`NDAOM-Admin Tracing` contains applications that are used for tracing information from the NetPath Probe and NDAOM executable files.

Figure 3-5 NDAOM-Admin Tracing Application Group



- **Cleanup Trace File (Win/UX)**

  Removes the NDAOM and PD Probe trace files (`ndaom.trc`, `xml.trc.*` and `npprobe.log`).

  Executed On Managed Node

  Parameter `$OPC_NODES`

- **NDAOM Tracing Off (Win/UX)**

  Turns off tracing for the NDAOM executable files (`ovnwmonitor`, `ovnwpdc`, `ovnwlinkmon`).

  Executed On Managed Node

  Parameter `$OPC_NODES`

- **NDAOM Tracing On (Win/UX)**

  Turns on tracing for the NDAOM executable files. Sets TRACE_LEVEL to 2.

  Executed On Managed Node

  Parameter $OPC_NODES

- **PD Probe Tracing Off (Win/UX)**

  Turns off tracing for the PD Probe (Netpath Probe).

  Executed On Managed Node

  Parameter $OPC_NODES

- **PD Probe Tracing On**

  Turns on tracing for the PD Probe (Netpath Probe).

  Executed On Managed Node

  Parameter $OPC_NODES

- **View NDAOM Tracefile**

  Views trace file of the NDAOM executable files.

  Executed On Managed Node

  Parameter $OPC_NODES

- **View PD ProbeTracefile**

  Views trace file of the PD Probe (Netpath Probe).

  Executed On Managed Node

  Parameter $OPC_NODES

- **View PD GUI Log/Trace File**

  Views log/trace file of the PD GUI (Apache server and servlets).

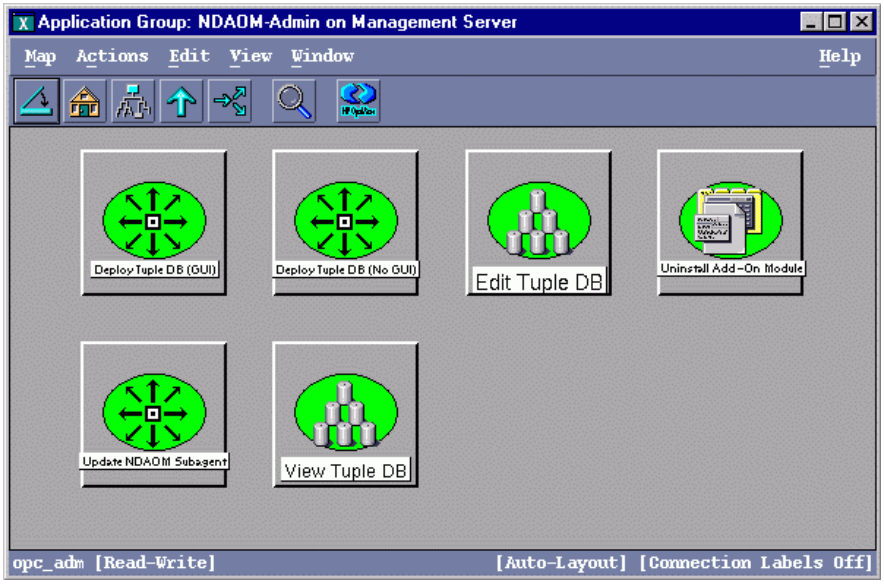
  Executed On Management Server

  Parameter $OPC_NODES

NDAOM-Admin on Management Server

NDAOM-Admin on Management Server contains applications that are used to do NDAOM configuration tasks, e.g.modifying the tuple DB with the editor.

Figure 3-6 NDAOM-Admin on Management Server Application Group



- **Deploy Tuple DB (GUI)**

  Deploys the tuple DB and, if necessary, the NDAOM subagent after the user edited the tuple DB.

  Executed On Management Server System

  Parameter -

  Not available from the OVO Java GUI (input/output windows not supported).

- **Deploy Tuple DB (No GUI)**

  Deploys the tuple DB and, if necessary, the NDAOM subagent without user interaction.

  Executed On Management Server System

  Parameter `$OPC_NODES`

- **Edit Tuple DB**

  Modify the Tuple database using the Tuple Editor.

  Executed On Management Server System

  Parameter -

  Not available from the OVO Java GUI (input/output windows not supported).

- **Uninstall Add-On Module**

  Removes the NDAOM product from the management server. You must use the customized start of this application and add the parameter **-YES** to uninstall NDAOM.

  Executed On Management Server

  Parameter `-YES`

- **Update NDAOM subagent**

  Updates the NDAOM subagent on the (selected) managed nodes. It removes the existing subagent and the new one is then deployed.

  Executed On Management Server System

  Parameter `$OPC_NODES`

- **View Tuple DB**

  Show the contents of the Tuple database.

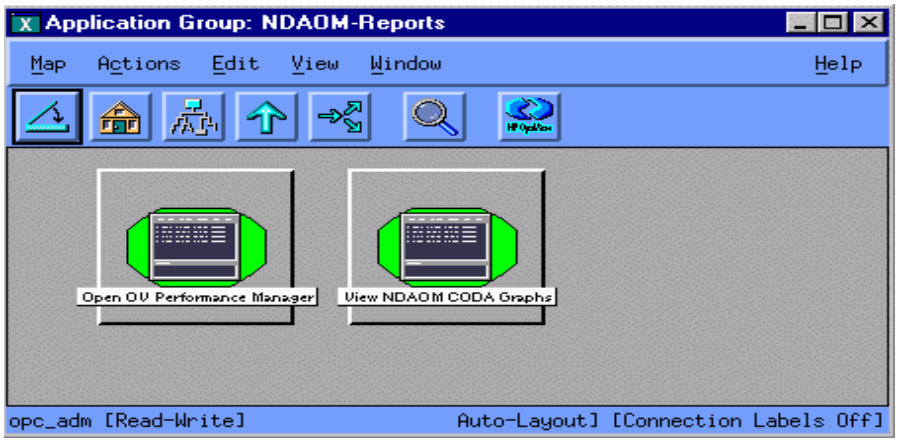
  Executed On Management Server System

  Parameter `$OPC_NODES`

NDAOM-Reports Application Group

`NDAOM-Reports` contains applications to launch a browser with specific reports.

Figure 3-7 NDAOM-Reports Application Groups



These tools are:

- **Open OV Performance Manager**

  Start OV Performance Manager web user interface. The variable PERFMGR_SERVER must be correctly specified in the NDAOM configuration file:

  `/etc/opt/OV/ndaom/conf/ndaom.cfg`

  Perl 5.004 must be installed on the OVO management server system. If Perl was not installed during the NDAOM installation, you must manually start:

  `/opt/OV/ndaom/bin/ndaom_apache.sh`

  which configures the web pages that contain the links to the OV Performance Manager.

  Executed On Management Server System

  Parameter `$OPC_MGMTSV`

- **View NDAOM CODA Graphs**

  Shows OV Performance Manager reports for NDAOM. The variable PERFMGR_SERVER must be correctly specified in the NDAOM configuration file:

  `/etc/opt/OV/ndaom/conf/ndaom.cfg`

Perl 5.004 must be installed on the OVO management server system. If Perl was not installed during the NDAOM installation, you must manually start:

`/opt/OV/ndaom/bin/ndaom_apache.sh`

which configures the web pages that contain the links to the OV Performance Manager.

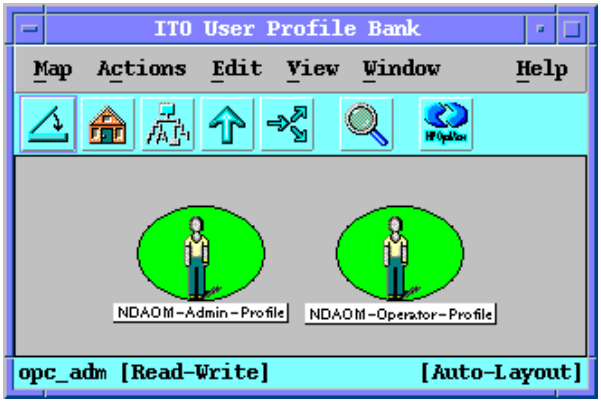
Executed On	Management Server System
Parameter	`$OPC_NODES`, `$OPC_MGMTSV`

User Bank

No users are generated. It is assumed that the operators use the NDAOM as add-on functionality. For example, they already have an SAP SPI installed and will use those users. Thus NDAOM will just offer two profiles that can be assigned to existing HP OpenView Operations users.

User Profile Bank

Figure 3-8 **User Profile Bank**



Two profiles are generated:

- `NDAOM-Admin-Profile`
- `NDAOM-Operator-Profile`

The following table shows the responsibilities associated with the profiles. Differences between the two profiles are mainly the administrative tasks that only the NDAOM administrator is allowed to do.

	NDAOM-Admin-Profile	**NDAOM-Operator-Profile**
Applications	NDAOM-Admin NDAOM-Reports NDAOM	NDAOM-Reports NDAOM
Message Groups	Network NDAOM	Network NDAOM
Node Groups	NDAOM	NDAOM

Message Source Templates

The standard NDAOM templates are distributed using the HP OpenView Operations deployment mechanism. The templates are divided into logical groups and can be assigned to a managed node individually or together.

The policies are available under the policy group:

NDAOM	Templates for internal NDAOM.

Message Templates

The following message templates are available:

- **NDAOM_Messages**

  Message template that adds an operator initiated action to the network path messages originating from the NDAOM monitor (ovnwmonitor). This operator initiated action starts the PD GUI in the context of the network path concerned.

NOTE

This operator initiated action can only be started from the OVO Java GUI, *NOT* from the message browser in the OVO Operator Motif GUI (started with the opc command).

Condition	Network path message
Severity	Unchanged
Description	Adds an operator initiated action to Network Path related messages

This template also intercepts the message “NDAOM Subagent successfully installed/removed” and forwards it to the management server.

Condition	NDAOM Subagent installation message
Severity	Unchanged

Description	Forwards the NDAOM subagent installation/removal messages to the managment server

Logfile Templates

The following logfile templates are available:

- **NDAOM_NDAOM_log_Ux** and **NDAOM_NDAOM_log_Win**

  These templates analyze messages logged in `ndaom.log` and prepares them to be sent to management server. They are located at:

  HP-UX and Solaris `/var/opt/OV/ndaom/log/ndaom.log`

  Windows NT `<OVO Agent Software installation Directory>\ndaom\log\ndaom.log`

  Entries have the format "`<mm/dd/yyyy> <hh:mm:ss> [<severity>] NDAOM-<msg_code>(<program_name>): <msg_text>`".

Condition	Checks for errors in ndaom.log	Checks for warnings in ndaom.log	Checks for info messages in ndaom.log
Severity	Critical	Warning	Normal
Description	Forwards any error condition	Forwards any warning condition	Forwards any information condition
Object	<object>	<object>	<object>
Message Text	<err_txt>	<warn_txt>	<info_txt>

- **NDAOM_NPPROBE_log_Ux** and **NDAOM_NPPROBE_log_Win**

  These templates prepare the messages logged in `npprobe.log` to be sent to management server. They are located at:

  HP-UX / Solaris `/opt/OV/pd/netpath/log/npprobe.log`

  Windows NT `c:\Program Files\HP OpenView \pd\netpath\log\ndaom.log`

Entries have the format “<mm/dd/yyyy> <hh:mm:ss> [<severity>] NDAOM-<msg_code>(<program_name>): <msg_text>”.

Condition	Checks for errors in npprobe.log	Checks for warnings in npprobe.log	Checks for info messages in npprobe.log
Severity	Critical	Warning	Normal
Description	Forwards any error condition	Forwards any warning condition	Forwards any information condition
Message Text	<err_txt>	<warn_txt>	<info_txt>

Monitor Templates

The following logfile templates are available:

- **NDAOM_Monitor**

  A monitor template that uses the NetPath probe to analyze paths.

  Command `ovnwmonitor` (external monitor)

Condition	RespTime Error	RespTime Warning	RespTime Test
Severity	Major	Warning	Normal
Description	Message will be sent if response time is much too long.	Message will be sent if response time is too long.	Only for test purposes.
Threshold Value	1000 ms	500 ms	10 ms

- **NDAOM_Logfiles_Ux**

  A monitor template that checks the size of the NDAOM log and trace files on UNIX systems. If the size exceeds a specified limit, the logfile or tracefile is truncated and compressed.

  Command `ndaom_logsize.sh NDAOM_Logfiles_Ux`

  Action `ndaom_trunclog`

Condition	Size of error log
Severity	Normal
Description	File size limit reached - truncated
Threshold Value	500 KB

- **NDAOM_Logfiles_Win**

  A monitor template that checks the size of the NDAOM log and trace files on Windows NT/2000 systems. If the size exceeds a specified limit, the logfile or tracefile is truncated and compressed.

  Command `ndaom_logsize.vbs NDAOM_Logfiles_Win`

  Action `ndaom_trunclog.vbs`

Condition	Size of error log Size of trace log
Severity	Normal
Description	File size limit reached - truncated
Threshold Value	500 KB

- **NDAOM_Vitalfiles_Ux**

  A monitor template that checks the existence of vital files, such as the tuple database files (nwlmdb_sv, nwlmdb_agt) and the config file (ndaom.cfg) in UNIX systems. If a file is accidentally deleted or moved, a critical message is sent to the management server, identifying the file and informing you that you must restore it.

Command `ndaom_vitalfiles.sh NDAOM_Vitalfiles_Ux`

Conditions	Check for existence of NDAOM Config file Check for existence of NDAOM Tuple DB
Severity	Critical
Description	NDAOM Vital file (TupleDB / ndaom.cfg) missing. Please restore the file.

- **`NDAOM_Vitalfiles_Win`**

  A monitor template that checks the existence of vital files, such as the tuple database files (nwlmdb_sv, nwlmdb_agt) and the config file (ndaom.cfg) in Windows NT/2000 systems. If a file is accidentally deleted or moved, a critical message is sent to the management server, identifying the file and informing you that you must restore it.

  Command `ndaom_vitalfiles.vbs NDAOM_Vitalfiles_Win`

Conditions	Check for existence of NDAOM Config file Check for existence of NDAOM Tuple DB
Severity	Critical
Description	NDAOM Vital file (TupleDB / ndaom.cfg) missing. Please restore the file.

Messages

The following situations caused by events sent from the event port create messages:

Path Has Changed

This message is sent due to an event from the event port of the probe.

Attribute	Value
Severity	Minor
Node	NDAOM_Infrastructure
Application	NDAOM
Message Group	Network
Object	Destination Node
Message Text	Network connection between <$MSG_NODE> and <$OBJECT> has changed from path <$PATH1> to <$PATH2>
Automatic Action	None
Op. Initiated Act.	Start Problem Diagnosis GUI—This operator initiated action can only be started from the OVO Java GUI, NOT from the message browser in the OVO Operator Motif GUI (started with the opc command).

Link Down

This message is sent due to an event from the event port of the probe.

Attribute	Value
Severity	Major
Node	NDAOM_Infrastructure
Application	NDAOM
Message Group	Network
Object	Destination Node
Message Text	Network connection between <$MSG_NODE> and <$OBJECT> failed
Automatic Action	None

Op. Initiated Act.	Start Problem Diagnosis GUI—This operator initiated action can only be started from the OVO Java GUI, NOT from the message browser in the OVO Operator Motif GUI (started with the opc command).

Link Up

This message is sent due to an event from the event port of the probe.

Attribute	Value
Severity	Normal
Node	NDAOM_Infrastructure
Application	NDAOM
Message Group	Network
Object	Destination Node
Message Text	Network connection between <$MSG_NODE> and <$OBJECT> up
Automatic Action	None
Op. Initiated Act.	None

Network Response Time Too High

This message is sent due to an exceeded threshold value. The monitor uses data from the performance port of the probe.

Attribute	Value
Severity	Warning
Node	NDAOM_Infrastructure
Application	NDAOM
Message Group	Network
Object	Destination Node
Message Text	Response time between <$MSG_NODE> and <$OBJECT> was <$VALUE>. Threshold is <$THRESHOLD>.
Automatic Action	None
Op. Initiated Act.	Start Problem Diagnosis GUI—This operator initiated action can only be started from the OVO Java GUI, NOT from the message browser in the OVO Operator Motif GUI (started with the opc command).

Network Path Detection Too Slow

This message is sent when the polling interval that the Netpath Probe uses to monitor the network connections is smaller than the time required by the path detection process.

Attribute	Value
Severity	Normal
Node	NDAOM_Infrastructure
Application	NDAOM
Message Group	Network
Object	Destination Node
Message Text	Network path between <$MSG_NODE> and <$OBJECT> could not be detected in <$INTERVAL> minutes, so the polling interval was increased to <$NEW_INTERVAL> minutes.
Automatic Action	None
Op. Initiated Act.	Start Problem Diagnosis GUI—This operator initiated action can only be started from the OVO Java GUI, NOT from the message browser in the OVO Operator Motif GUI (started with the opc command).

4 Integrating HP OpenView Network Diagnosis Add-On Module into OVO Service View

Integration into OVO Service View

Network connections monitored by the NDAOM are integrated into OVO Service View, allowing network problems to be displayed within Service View.

In case of a network problem, the NetPath GUI can be launched from the service object representing the network connection that caused the problem. For this reason, the tool to start the NetPath GUI is always part of the set of tools that can be launched from a network connection service object.

To configure the Service View, you use the tool called `ovnwlinkmon`. There are two ways to execute `ovnwlinkmon`:

- Use the command line interface. See “Command Line Interface ovnwlinkmon” on page 76.
- Use the graphical user interface (GUI). See “Graphical User Interface ovnwlinkmon” on page 92.

NOTE You must not add services representing network connections directly into the Service Configuration, because they will not be monitored. All network connections to be monitored must be defined using `ovnwlinkmon`.

Command Line Call ovnwlinkmon

A Smart Plug-In can manually define the network infrastructure that is to be monitored using `ovnwlinkmon`.

`ovnwlinkmon` offers the Smart Plug-In/ OVO administrator the possibility to decide whether a new service object representing the network connection is to be inserted into the Service View or not.

Command Line Interface ovnwlinkmon

The command line interface ovnwlinkmon has the following options:

ovnwlinkmon -add [-NoNewObject]

[root=<RootServiceID> parent=<ParentServiceID>]
[label=<Label>] interval=<PollingInterval>
source=<SourceNode> target=<TargetNode>

This option adds a new network connection to be monitored into the tuple database, where:

NoNewObject is the option determining whether a new service object is created that gets the messages for the monitored network connections or the messages are sent to the parent service. This option is optional. The default is that a new service object is created.

<RootServiceID> is the Service ID of the service object where the NW Infrastructure is inserted as sub-service and the new service object that is inserted belongs to. If the `-NoNewObject` option is specified then the `RootServiceID` is only needed to determine to which service object the network connection belongs. This information is needed later on for the delete operation.

<ParentServiceID> is the Service ID of the service object that is the parent service of the newly created service object. If the `-NoNewObject` option is specified this service gets the messages for the monitored network connection. The parent service can only be either in the sub tree of the root service or the same as the root service.

<Label>	is the service label of the new created service object. This parameter is optional. If it is not specified the label is the same as the service ID. This parameter is only needed if the option `-NoNewObject` is not specified. Otherwise it is ignored.
<PollingInterval>	is the interval (with unit) that determines how often the state of the network connection is polled. The default unit is minutes.
<SourceNode>	is the IP Address or node name of the start point of the path.
<TargetNode>	is the IP Address or node name of the endpoint of the path.

A new tuple, with all parameters and the `TargetServiceID`, is written into the tuple database. The `TargetServiceID` is the identifier of the service object that receives the messages for the associated network connection. If the option `-NoNewObject` has been specified, the `TargetServiceID` equals the `ParentServiceID`. If the option `-NoNewObject` has not been specified (the default case), the `TargetServiceID` is:

```
<RootServiceID>:NW:<IP Address of source node in Hex>_<IP
Address of target node in Hex>
```

If the hostname is specified for the source node or the target node and they have more than one interface, the IP address of the first interface is taken for the `TargetServiceID`. Therefore the `TargetServiceID` is always unique and it does not matter which IP address of a node is taken for the Service ID, as it is only needed for identification.

If no service view integration is required or if the HP OpenView Service Navigator is not installed, the root and parent options can be omitted.

It is possible to monitor the same network connection for different parent services.

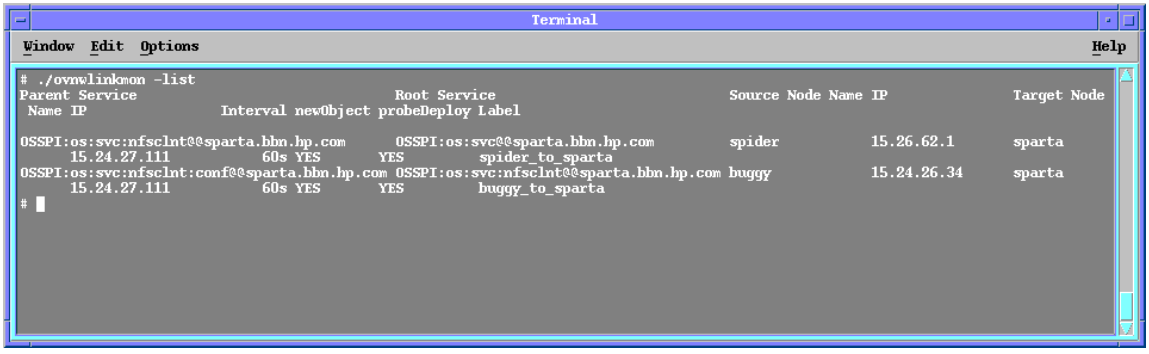
NOTE `ovnwlinkmon -add` only adds connection information to the global tuple database. It does not deploy the tuple database nor the NDAOM subagent.

In order to start monitoring the connections you have added to the tuple database, you must use the command **ovnwlinkmon -deploy** which actually deploys the tuple database and the NDAOM subagent to the managed node.

If `ovnwlinkmon -list` is executed the output of the tuple database is shown (provided that at least one network connection is defined). An example can be seen in Figure 4-1 on page 78:

Parent Service Root Service Source Node Name IP Target Node Name IP Interval newObject probeDeploy Label

Figure 4-1 ovnwlinkmon -list Output



The Tuple Database Formats

The tuple database file can be modified before it is deployed. The tuple database file has the following format:

<State> <TargetSvcID> <ParentSvcID> <RootSvcID> - <Source Name> <Source IP in Hex> - <Target Name> <Target IP in Hex> - <Interval in seconds> <noNewObject> - <probeDeploy> <xmlDeploy> <ldbDeploy> - <Label>

where:

`<State>`	State of the network connection L = Valid Link D = Deleted Link
`<TargetSvcID>`	Created Service ID

`<ParentSvcID>`	Parent Service ID
`<RootSvcID>`	Root Service ID
`<Source Name>`	IP address or hostname of source system
`<Source IP in Hex>`	IP address of source system in Hex
`<Target Name>`	Hostname of target system
`<Target IP in Hex>`	IP address of target system in Hex
`<Interval in seconds>`	The interval (with unit) that determines how often the state of the network connection is polled. The default value for the unit is seconds.
`<noNewObject>`	Determines whether a new service object is created for the network connection 0 = new service object is created 1 = no new service object is created
`<probeDeploy>`	Has the Netpath Probe been deployed to source node? 0 = No 1 = Yes
`<xmlDeploy>`	Has the network connection been inserted into the Service Tree? 0 = No 1 = Yes
`<ldbDeploy>`	Has the tuple database been deployed to the source node? 0 = No 1 = Yes

`<Label>`	The service label of the new created service object. This parameter is optional. If it is not specified the label is the same as the service ID. This parameter is only needed if the option `-NoNewObject` is not specified. Otherwise it is ignored.

An error message is generated if:

- the syntax of the command is incorrect.
- no IP address exists for a node being checked.

ovnwlinkmon -delete

`root=<RootServiceID> [parent=<ParentServiceID>] source=<SourceNode> target=<TargetNode>`

This option marks a network connection in the list of monitored connections in the tuple database as deleted.

Depending on the parameter specified, it is possible to delete the specified network connection only for just one parent service or for all parent services.

If no parent service is specified, all network connections for the specified root service and the correct endpoints from the tuple database are marked as deleted. If no such network connections exist, a logging message is created. If a parent service is specified, only one network connection is deleted from the tuple database (the one with the correct root, parent, source and target).

An error/warning message is generated if:

- the syntax of the command is incorrect.
- a network connection with the specified parameters does not exist in the tuple database.
- the tuple database is empty.

ovnwlinkmon -delete_all

`[root=<RootServiceID>][source=<source node>]`

This option marks all network connections in the tuple database as deleted.

If no root service is specified, all tuples in the tuple DB are deleted.

If a root service is specified, all tuples with the specified RootServiceID as deleted in the tuple DB.

If a source node is specified, only the entries for this source node are deleted.

An error/warning message is generated if:

- the syntax of the command is incorrect.
- the tuple database is empty.
- a network connection with the specified root does not exist in the tuple database.

ovnwlinkmon -clear

This option clears all entries from the tuple database for each source node that is marked for deletion. Using `ovnwlinkmon -clear` is the only way to really remove tuples from the tuple database. All other operations only mark tuples as deleted.

If all tuple database entries of a source node are marked as deleted and all deployment flags of these entries are set (i.e. the entries that are to be deleted have been deployed before), all these entries from the tuple database are deleted.

An error/warning message is generated if:

- the syntax of the command is incorrect.
- the deployment flags are not set. Deployment must be done for these nodes before its entries can be removed.

ovnwlinkmon -list

`[root=<RootServiceID>][-default|-verbose]`

This option shows a list of all monitored network connections in the tuple database. If a root service is specified, only the network connections belonging to that root service are listed.

An error/warning message is generated if:

- the syntax of the command is incorrect.
- the tuple database is empty.
- there are no network connections defined for the root service.

ovnwlinkmon -remove_sa

[source=<source node>]

This option removes the NDAOM subagent for either all or a specified source node.

It checks the tuple database for entries of all or the specified node. If all relevant entries are marked as deleted, the subagent is removed from each specified node.

An error/warning message is generated if:

- the syntax of the command is incorrect.
- the deleted state flags are not set.

ovnwlinkmon -update

[source=<source node>]

This option updates the NDAOM subagent for either all or a specified source node.

It checks the tuple database for source node entries that are not marked as deleted or filters the specified source node. For all marked or specified nodes, the subagent is updated. The deployment process removes the existing subagent and installs the new one.

An error/warning message is generated if:

- the syntax of the command is incorrect.
- no entries can be found with set update flags.

ovnwlinkmon -deploy

[-NoGUI>][source=<source node>]

NOTE

Make sure that the Problem Diagnosis server is running before you start the NDAOM subagent deployment/installation. If it is not running, start it with:

UNIX	`/opt/OV/pd/app_server/bin/ovpdstart`
Windows	**Start** → **Programs** → **HP OpenView** → **Problem Diagnosis** → **PD Server-Start**

If the Problem Diagnosis server is not running during the NDAOM subagent deployment/installation the newly installed Netpath Probe may not be able to register at the Problem Diagnosis server and so the Problem Diagnosis server may not know this Netpath Probe.

This action executes the deployment to the managed nodes in three steps:

- Deployment of the tuple database
- Deployment of the NDAOM Subagent
- Update of the service view in the HP OpenView Service Navigator

If a source node is specified the deployment action will be performed for the specified node only.

Deployment of the Tuple Database

First the contents of the tuple database on the OVO management server are deployed to the source nodes listed in the tuple database.

If the option `-NoGUI` is not used, the Smart Plug-in administrator can edit the contents of the tuple database before it is downloaded to the managed nodes. If the option `-NoGUI` is used, the contents of the tuple database are downloaded to the managed nodes without the option of being able to edit it first. The default is that the SPI/administrator can edit the tuple database before deployment is started.

Deployment of the NDAOM Subagents

Next, the NDAOM subagent (`ovnwmonitor`, `ovnwpdc`, `Netprobe`) are deployed to the source nodes of all network connections, provided that it has not already been deployed to these nodes.

The NDAOM subagent deployment consists of the following steps:

- The NDAOM subagent packages are deployed to the managed node via `opctranm`.
- These packages are unpacked and a `checkinstall` script checks the configuration of the managed node.
- A background process is started which installs and configures the PD Probe and the NDAOM subagent executable files (`ovnwmonitor`, `ovnwpdc`).
- The PD Probe and the NDAOM subagent executable files are registered at the OVO agent via `opcagtreg`.
- An `opcmsg` message is sent to the message browser of the OVO management server which informs whether or not the installation was successful.

  If the NDAOM subagent installation fails (for example with the error message:

  ```
  Deployment of the subagent to ... via opctranm failed.
  ... opctranm output was /tmp/install.success: No such
  file or directory)
  ```

  check the installation log file on the managed node system:

  Unix `/tmp/install_nwagt.log`

  Windows `\TEMP\install_nwagt.log`

NOTE

The NDAOM subagent installation is not finished until this message is sent. It is possible to observe the progress of the NDAOM subagent installation by executing the following command on the managed node:

```
tail -f /tmp/install_nwagt.log
```

Update of the Service View in Service Navigator

If deployment of the NDAOM subagent is successful, the contents of the tuple database is synchronized with the service view on the management server.

For every tuple of the tuple database, a check is made to ascertain whether a new service object has to be inserted into the Service View. If the `-NoNewObject` option is set for that tuple, no new service object has

to be inserted. If the `-NoNewObject` option is not set for that tuple, a check is made to ascertain whether a service object for the NW Infrastructure of the root service already exists in the Service View. If this is not the case, a new service object with the service ID:

`Net:<RootServiceID>`

is inserted as a sub-service of the root service into the Service View.

Next, a check is made to ascertain whether a service object for the NW Infrastructure of the parent service already exists in the Service View. If not, a new service object with the service ID:

`Net:<ParentSrvID>`

is inserted as sub service of the service representing the NW Infrastructure of the root service.

Then a check is made to establish whether a service object with the same service ID as the TargetServiceID of the tuple already exists in the Service View. If not, a new service object with that service ID as sub service of the NW Infrastructure of the parent service is inserted. If such a service already exists, only a dependency between the NW Infrastructure of the parent service and the service representing the network connection is created.

An error/warning message is generated if:

- the syntax of the command is incorrect.
- the tuple database is empty.
- the tuple database on the management server is not deployed successfully to the managed nodes.
- the NetPath Probe and the Monitor are not deployed successfully to the managed nodes.
- the Service View update was not successful for all network connections.
- the source node of a network connection is not a managed node.
- the root or the parent service does not exist.

Example

The Application Server of a SAP System always needs a working network connection between itself and the Database Server, for example, when it needs to get data out of the database. While the Database Server is not affected if there is a problem on that network connection, this network problem has a high impact on the Application Server. Without the network connection, the Application Server can not retrieve data for its applications from the Database.

In this example, let us assume that the Application Server resides on the node `parsley` and the Database Server resides on the node `sundev01`. To get data out of the database, the Application Server makes a request to the Database Server. To do this, it must have a working network path from the Application Server to the Database Server. To get the retrieved data, a working network path from the Database Server back to the Management Server must also exist. Because of this, the path from the Application Server to the Database Server and back must be monitored. To monitor these paths the following calls of `ovnwlinkmon` are executed:

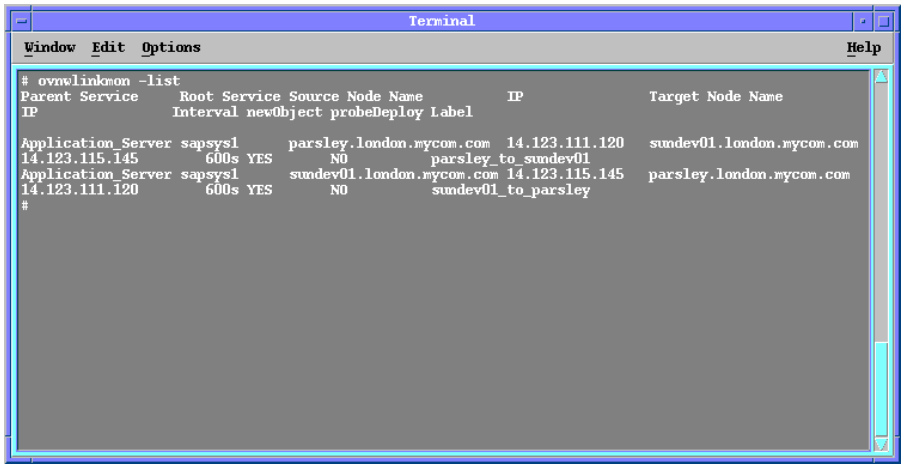
```
ovnwlinkmon -add root=sapsys1 parent=Application_Server
interval=10m source=parsley.london.mycom.com
target=sundev01.london.mycom.com label=parsley_to_sundev01
```

and

```
ovnwlinkmon -add root=sapsys1 parent=Application_Server
interval=10m source=sundev01.london.mycom.com
target=parsley.london.mycom.com label=sundev01_to_parsley
```

As the Application Server and the Database Server are part of the SAP System `sapsys1`, this service is used as the root service. The parent service is the Application Server, as it is dependent on these network paths. The first network connection monitors the path from node `parsley.london.mycom.com` to node `sundev01.london.mycom.com` while the second network connection monitors the path from node `sundev01.london.mycom.com` to node `parsley.london.mycom.com`. The status of these network connections is polled every 10 minutes. A call of **`ovnwlinkmon -list`** shows the following:

Figure 4-2 ovnwlinkmon Output



Although these network connections have been inserted into the tuple database, they will not be monitored before they are deployed. This is done with the following `ovnwlinkmon` call:

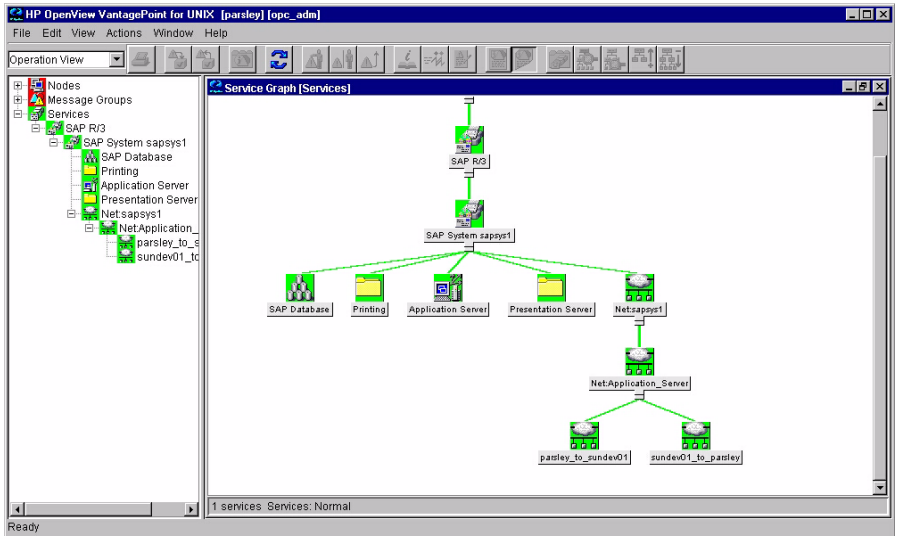
```
ovnwlinkmon -deploy -NoGUI
```

If the `-NoGUI` option is omitted, the SPI/OVO Administrator is asked whether he wants to modify the tuple database before the deployment. With the `-NoGUI` option, the tuple database is deployed without being modified.

NOTE Probe Deployment is only successful for network connections where the source node is a managed node. If the node is not a managed node, the probe can not be deployed and an error message is generated.

The Service View is updated after the tuple database and the Probe have been deployed. However, before this is done the existence of root service and parent service is checked. If the root or parent services do not exist, an error message is generated and the network connection will not be inserted into the service view.

Figure 4-3 Service View with the Net:sapsys1 Subservice

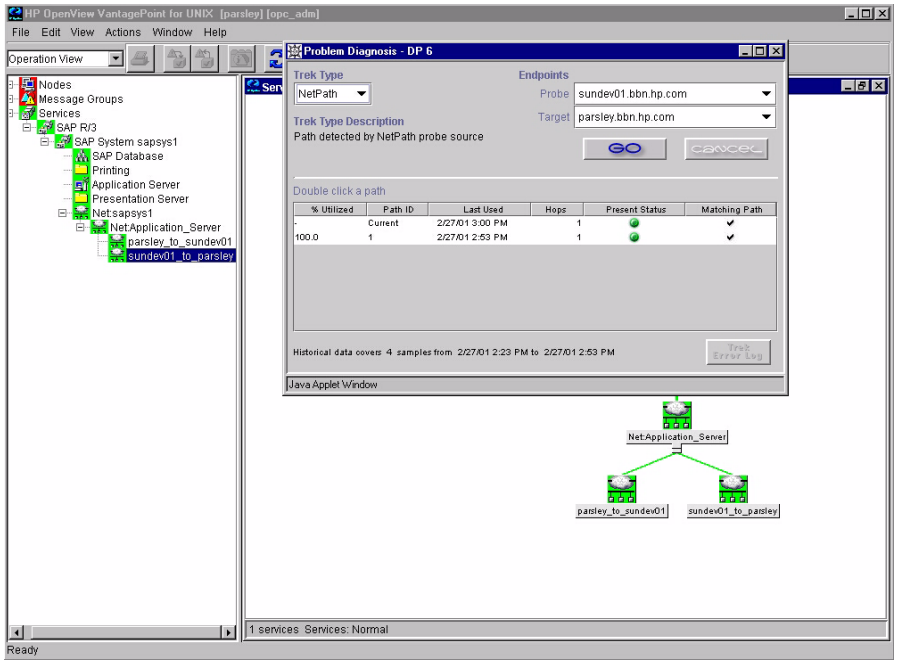


A new service object, `Net:sapsys1`, is inserted as a subservice of the Root Service. This service represents the NW Infrastructure of the SAP System `sapsys1`. It contains just one subservice, the NW Infrastructure of the Application Server. Subsequently, if more network connections are defined with `sapsys1` as the root or parent service, the new services (NW Infrastructures or network connections) are inserted in the `Net:sapsys1` object. The two network connections that were defined with the above commands are inserted as subservices of the NW Infrastructure of the parent service. The service `Net:Application_Server` in this example. Now if network problems occur on the path between parsley and sundev01, for example, one of the interfaces on the path is down, this is displayed immediately in the OVO GUI as the status of a network connection is always propagated up to the root service. Therefore, it is easy to detect if the SAP System has a problem caused by network problems or by any of its applications.

In case of a network problems, the NetPath GUI can be launched from the service object that represents the network connection that caused the problem. To be able to do this, the Network Diagnosis Add-On Module inserted actions into the Service Configuration to start the NetPath GUI from network connection service objects. This is a web interface that displays the network connection information between two given nodes. It

can be launched either in the context of the message that generated the status change or the service object representing the network connection. The NetPath GUI shows all possible paths between the nodes of the network connection as well as more details about a specific path.

Figure 4-4 Service View with NDAOM Actions



With the following `ovnwlinkmon` call, an additional network connection is defined that monitors the path from the Application Server to the Printer on node `escape.london.mycom.com`:

```
ovnwlinkmon -add -NoNewOject root=sapsys1
parent=Application_Server interval=30m
source=parsley.london.mycom.com
target=escape.london.mycom.com
```

As the `-NoNewObject` option is specified, no new service object is created for this network connection. Instead, all messages for this network connection are sent directly to the service object representing the Application Server.

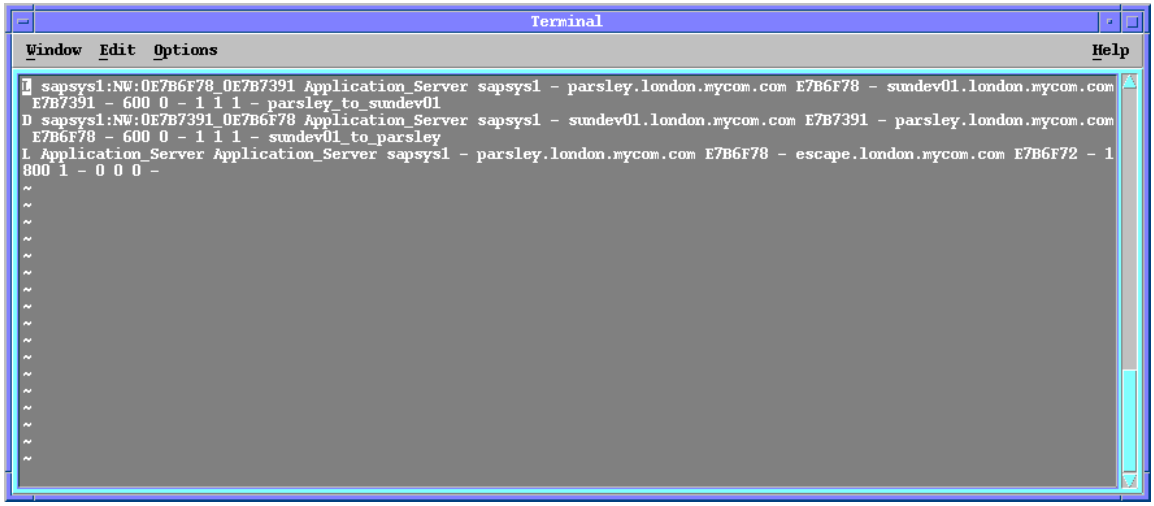
Later, if it is decided that the network connection from sundev01 to parsley is not to be monitored any more, the connection is deleted with the following command:

```
ovnwlinkmon -delete root=sapsys1
source=sundev01.london.mycom.com
target=parsley.london.mycom.com
```

These add-/delete-operations will not become effective before the next deploy operation. Before the network connection between the Application Server and the Printer is not monitored while the network connection between the Database Server and the Application Server is still monitored.

The following illustration shows the content of the tuple database before the deploy operation. The format of the tuple database is described in “The Tuple Database Formats” on page 78.

Figure 4-5 Contents of the Tuple Database before Deploy Operation



The first letter of the definition of a network connection shows the state of the network connection. Network connections that is no longer to be monitored after the next deploy operation, show as state the letter “D” (= Deleted network connection). In the example this is the network connection `sundev01_to_parsley`. For the last network connection, the `noNewObject` flag is set to 1, as no service object is created for this network connection. In addition the `probeDeploy`, `xmlDeploy` and

`ldbDeploy` flags are set to zero. That means that neither the NetPath Probe nor the tuple database has been deployed successfully to the source node of the network connection nor the Service Configuration has been updated successful. These flags should never be modified, otherwise it can lead to inconsistencies. For example, if the `probeDeploy` flag is set to 1 although the Probe has not been deployed to the source of the network connection, this network connection will not be monitored.

Graphical User Interface ovnwlinkmon

The GUI for ovnwlinkmon has the same options as the command line interface. This section describes the appearance of the GUI and explains how to execute each option. For more details about these options, see “Command Line Interface ovnwlinkmon” on page 76.

GUI Configuration

When you launch the ovnwlinkmon GUI, it connects to a Perl script file on the management server. The script file is named ovnwscript.pl, and it is located in this path:

/opt/OV/httpd/htdocs/cgi-bin/ovnwscript.pl

To enable the GUI, follow these steps:

1. Create a link for the Perl executable in the /usr/bin path, for example,

   ln -s/opt/perl/bin/perl/usr/bin/perl

2. To configure the Apache server for NDAOM, execute this script:

   /opt/OV/ndaom/bin/ndaom_apache.sh

Text Files

The Perl script file saves information about the nodes and services that can be referenced on the GUI screens. The information is stored in four text files in this directory: /opt/OV/httpd/htdocs/ndaom/conf.

nodes.txt	List of previously entered Source nodes and Target nodes.
rservice.txt	List of previously entered Root Services.
pservice.txt	List of previously entered Parent Services.
lastvalue.txt	List of the values most recently entered on the Add screen.

Information is added to these files when you enter values for the services and nodes on the Options screen (see “Maintain Frequently Used Values” on page 102).

Launch the ovnwlinkmon GUI

To display the ovnwlinkmon GUI, perform these steps:

1. Open a web browser, such as Netscape Navigator or Microsoft Internet Explorer.
2. Enter the following in the browser address line:

   http://<mgmt_server>:<port>/ndaom/ovnwlinkmonUI.html

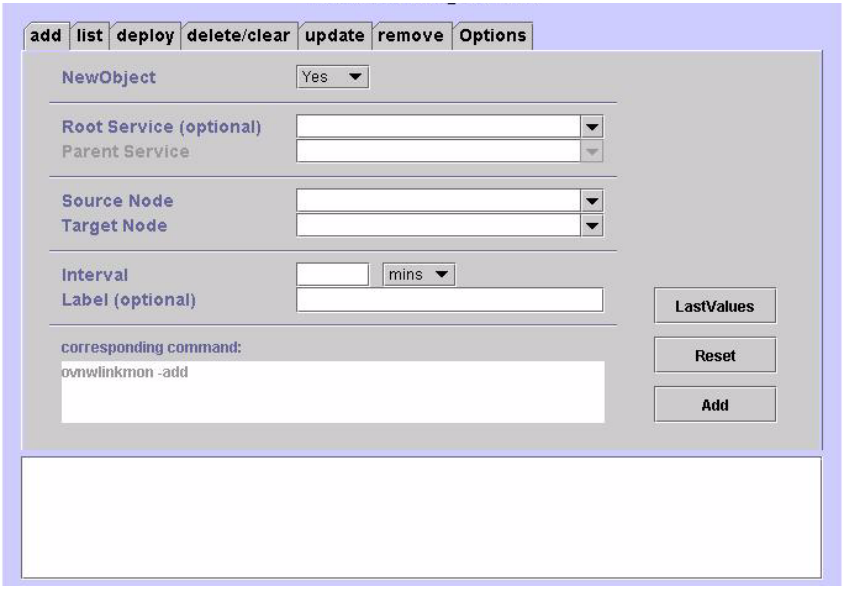
   where:

   <mgmt_server> is the name of the management server where NDAOM is installed

   <port> is the number of the port used by the management server (default = 8880)

   The following screen displays:

Figure 4-6 First ovnwlinkmon GUI Screen



NOTE The first time you access the GUI, there may be no items in the drop-down lists associated with the fields. After you have entered values for the services and nodes on the Options screen (see “Maintain Frequently Used Values” on page 102), these values are saved in the previously described text files (for example, `nodes.txt`), and they are available for selection.

Add Network Connection Data Records

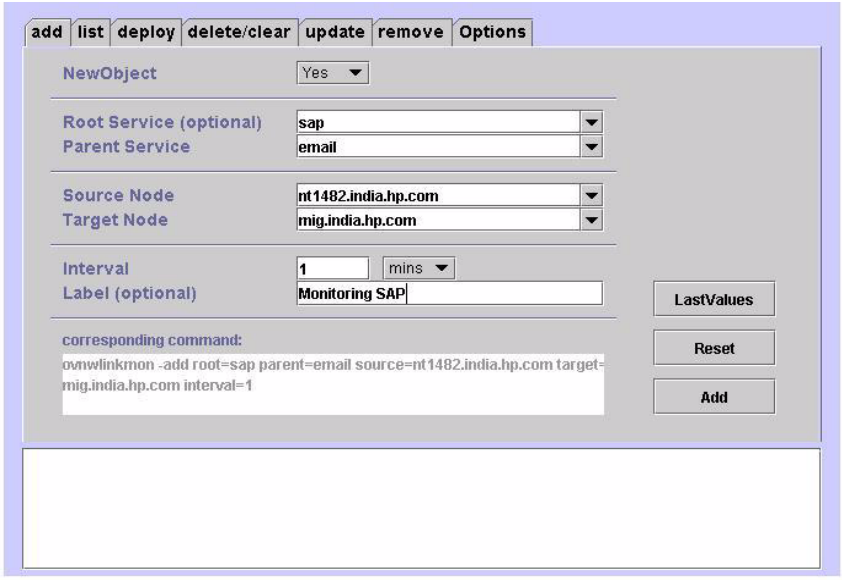
To add a network connection to the tuple database for monitoring, perform the steps below.

NOTE Adding a network connection only creates a data record in the global tuple database. To start monitoring the connections that you add to the tuple database, you must also deploy the database and the NDAOM subagent. See “Deploy Network Connections” on page 96.

1. Select the **add** tab.

   The following screen displays.

Figure 4-7 Add Screen



2. Complete all required fields. (See “ovnwlinkmon -add [-NoNewObject]” on page 76 for details about the fields.)

   - You can click **LastValues** to fill the fields with the data from the last successfully added data record.
   - You can click **Reset** to remove the data from all fields.

   As you enter data, the `corresponding command` pane displays the information in the form of a command line call.

3. When you finish entering data, click **Add**.

   The program checks the entered data and displays a message at the bottom of the screen if any errors are found.

4. If errors are found, repeat step 2 and step 3 until no error messages are displayed.

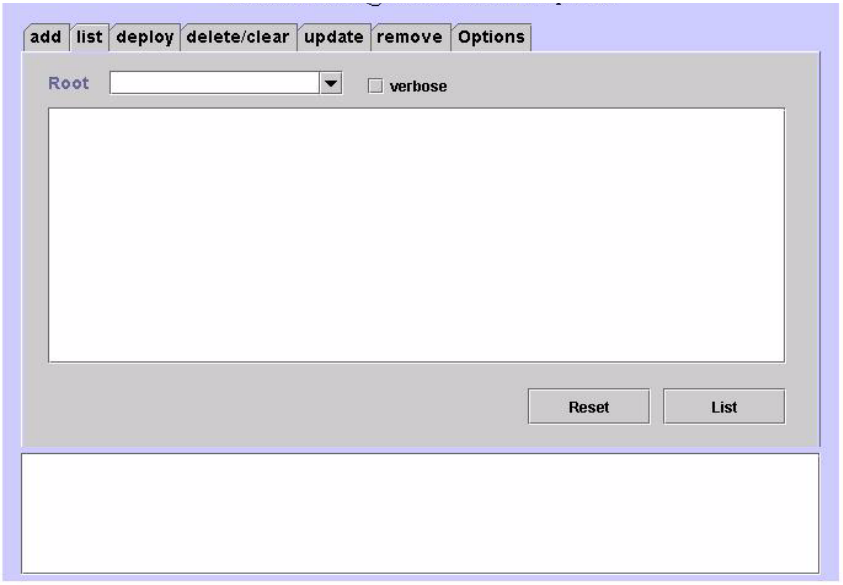
List Data Records

To display a list of the records in the tuple database, complete these steps:

1. Select the **list** tab.

   The following screen displays a list of the data records in the tuple database.

Figure 4-8 List Screen



2. You can modify the display by doing any of these actions:

   - Click **Verbose** to display all available information about each displayed record. If Verbose is not activated, the screen omits some information, such as data record labels.
   - In the **Root** field, enter a Root Service ID (or select one from the drop-down list) and click **List**. The information area displays only the records included in that Root Service ID.
   - Click **Reset** to remove all of the displayed data from the screen and turn off the Verbose setting.

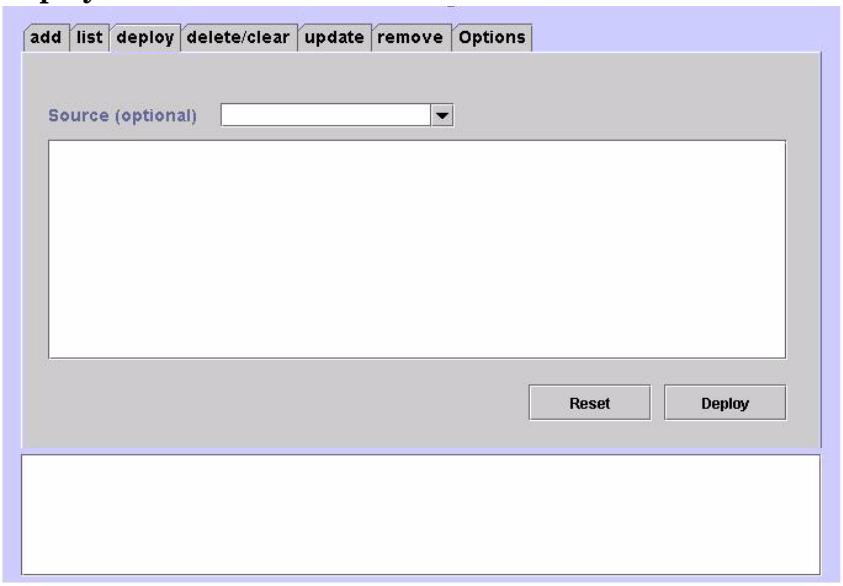
Deploy Network Connections

To deploy the tuple database and the NDAOM subagent, complete the steps below. See “ovnwlinkmon -deploy” on page 82 for more information about deployment and service view update.

1. Select the **deploy** tab.

   The following screen displays.

Figure 4-9 Deploy Screen



2. Do one of the following:

   - If you want to perform the deployment action to only one node, use the `Source` field to specify the desired node.
   - If you want to perform the deployment action for all nodes that have not yet been deployed, leave the `Source` field blank.

3. Click **Deploy**.

   The tuple database and the NDAOM subagent are deployed to the source nodes of the network connections that have not been previously deployed, or to the specific source node you specified in step 2 (if any). The service view of HP OpenView Service Navigator is also updated to add the new service object(s). A confirmation message appears in the message area of the screen.

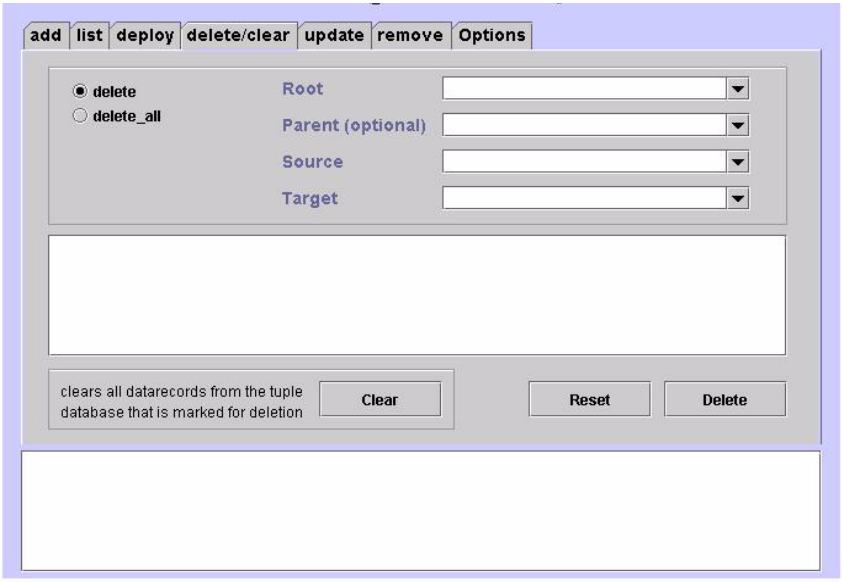
Delete and Clear Network Connections

To delete and clear network connections, complete the steps below. See “ovnwlinkmon -delete” on page 80 and “ovnwlinkmon -delete_all” on page 80 for more information about deleting and clearing network connections.

1. Select the **delete/clear** tab.

   The following screen displays.

Figure 4-10 Delete/Clear Screen



2. Do one of the following:

   - Select the **delete** radio button (default) to delete only the specified network connection.
   - Select the **delete_all** radio button to delete all network connections for the specified Root Service ID and Source Node.

3. Select a Root Service ID in the `Root` field.
4. Do one of the following:

   - If you want to delete the network connection for only one parent service, select the service in the `Parent` field.

- If you want to delete the network connections for all parent services, leave the `Parent` field blank.

5. Select the source node in the `Source` field.
6. Select the target node in the `Target` field.
7. Click **Delete**.

   One of the following happens:

   - The specified network connections are deleted, and the corresponding command line call displays in the message area.
   - If you did not enter the values correctly, an error message appears, and you can begin again with step 2.

8. When the network connections are successfully deleted, click **Clear**.

   The corresponding records in the tuple database are cleared.

Update the NDAOM Subagent

To update the subagent for all source nodes or for a specified node, follow the steps below. See “ovnwlinkmon -update” on page 82 for more information about updating the NDAOM subagent.

1. Select the **update** tab.

   The following screen displays.

Figure 4-11 Update Screen

2. Do one of the following:
   - If you want to update the subagent for one specific source node, select the node in the `Source` field.
   - If you want to update the subagent for all nodes in the tuple database that have not been marked for deletion, leave the `Source` field blank.
3. Click **Update**.

   One of the following happens:
   - The program removes the existing subagent and installs the new one. A confirmation message displays in the message area.
   - An error message displays if the tuple database is empty or if all entries in the database have been marked for deletion.

Remove the NDAOM Subagent

To remove the subagent for all source nodes or for a specified node, follow the steps below. See

for more information about removing the subagent.

1. Select the **remove** tab.

   The following screen displays.

Figure 4-12 Remove Screen

2. Do one of the following:

   - If you want to remove the subagent from one specific source node, select the node in the `Source` field.
   - If you want to remove the subagent from all nodes, leave the `Source` field blank.

3. Click **Remove**.

   The program removes the subagent from the specified node or all nodes. A confirmation message displays in the message area.

Maintain Frequently Used Values

The Options screen enables the user to save a list of frequently used values for Root services, Parent services, and Source/Target nodes for future use. This helps users reduce the amount of time required to enter data on the other screens.

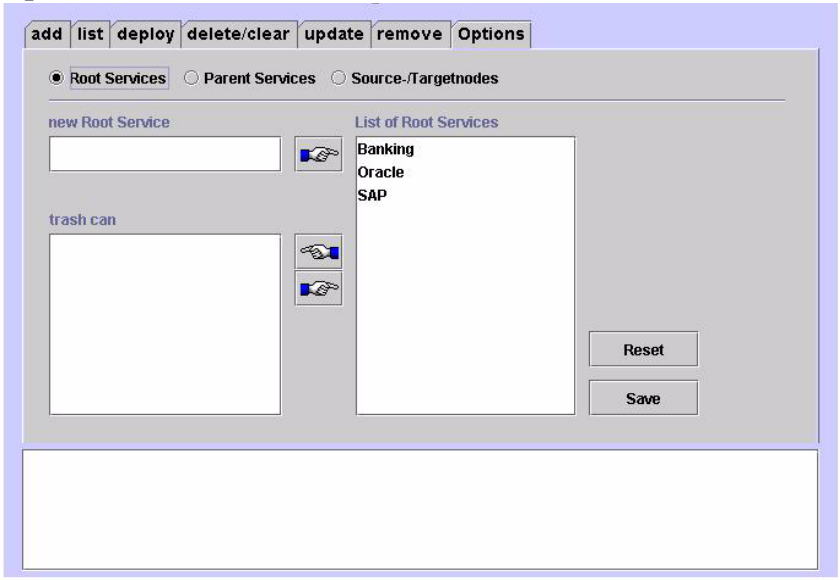
NOTE The information you enter on this screen is stored in the files described in “Text Files” on page 92.

To add values to the Options screen, follow these steps:

1. Select the **Options** tab.

   The following screen displays. (The first time you access this screen, all the fields may be blank.)

Figure 4-13 Options Screen



2. Click one of the radio buttons at the top of the screen. For example, if you want to add a Root Service, click the **Root Services** button.

The field names on the screen reflect the type of values you can enter. For example, if you click **Root Services**, the field in the upper left area is named `new Root Service` and the box to the right displays the Root Services that currently exist.

3. In the upper left field, enter the name of the Root Service, Parent Service, or Source/Target node you want to add.
4. Click the icon that points to the right.

   The new item appears in the list box at the right side of the screen.
5. To save the item, click **Save**.

   The new value is stored in the appropriate text file. For example, Root Services are stored in `rservice.txt`.

Each time you launch the `ovnwlinkmon` GUI in the future, the Options screen will display the stored values. You can add more values, and you can delete values that you no longer need. (See the procedure below.)

You can also access these values from other screens. For example, on the Add screen you can click the drop-down arrow on the `Root Service` combo box to display a list of the Root Service values that are stored in `rservice.txt`.

To delete values from the text files, follow these steps:

1. Click one of the radio buttons.

   Depending on the button you choose, a list of items appears in the list box.
2. Select a value in the box that you want to delete from the text file.
3. Click the icon that points to the left.

   The item moves to the box marked `trash can`. If you change your mind about an item, you can select it in the trash can box and click the icon that points to the right.
4. Keep selecting items and clicking the pointing icons until you are finished deleting items from the list.
5. Click **Save**.

   The selected items are deleted from the list box.

5 Integrating HP OpenView Network Diagnosis Add-On Module into OVP

In this chapter you will find information on integrating the NDAOM into HP OpenView Performance. The chapter is divided into two sections:

- Integration of performance data provided by the Netpath Probe
- The performance port of the Probe and the Embedded Performance Component mechanism of HP OpenView Performance

Integration of Performance Data with Embedded Performance Component

Performance metrics are collected by the embedded performance component that is part of the OVO agents. The performance component collects performance counter and instance data from a variety of sources, primarily operating systems. The collected values are stored in a proprietary persistent data store from which they are retrieved and transformed into presentation values. The presentation values can be used by extraction, visualization, and analysis tools such as HP OpenView Reporter and HP OpenView Performance for Windows. You cannot extract/export, view, or aggregate the data directly on the managed node.

The Embedded Performance Component is the new data collection tool which is distributed with HP OpenView Operations version A.07.00 and later. Embedded Performance Component collection is the preferred data collection mechanism and is always used when the Embedded Performance Component is installed on the managed node.

The Embedded Performance Component offers a powerful API and has many advantages in comparison to MWA, the original HP OpenView Performance Agent. For example, changes in configuration does not require a restart of the agent. For compatibility, some wrapper functions are used which provide the same interface as MWA but actually uses the Embedded Performance Component.

Embedded Performance Component Dynamic Data Feed (DDF)

The performance port of the Probe will be evaluated and the DDF mechanism of the Embedded Performance Component is used to collect the data.

The process of collecting data can be divided into the following steps:

- Scanning and parsing the netpath probe output by the NDAOM performance data collector (PDC).

- **For HP OpenView Reporter integration:**

  Pipe data to one log file for all destinations, using predefined log file metrics and the ddflog wrapper function. This log file contains all available data inclusive performance data to HOPs.

  For HP OpenView Performance Manager integration:

  Pipe data to one log file for all destinations, using predefined log file metrics and the ddflog wrapper function. This log file contains only performance data to the end destination and only the 4 major metrics (`DESTINATION`, `MIN`, `MEAN`, `MAX`).

HP OpenView Performance Manager and HP OpenView Reporter Integration

For the HP OpenView Reporter integration, one plain log file is created. It contains all performance data from all traced destinations paths, including all HOPs. The metrics are defined for the minimum possible amount of data.

For the HP OpenView Performance Manager (OVPM) integration another log file is created. This log file contains only performance data to the end destination and only the 4 major metrics (`DESTINATION`, `MIN`, `MEAN`, `MAX`).

The integrated data is accessible in HP OpenView Performance Manager immediately and in the HP OpenView Reporter database after a period of logging by the Embedded Performance Component and the HP OpenView Reporter (typical 24 hours). Data from the HP OpenView Reporter database can be used for generating reports and for further calculations.

The DDF process

The DDF process is used by calling wrapper functions `ddfcomp`, `ddflog` and `ddfutil`.

A DSI Class Specification file, see Appendix B, “DDF Class Specification Files,” on page 133, containing all NDAOM metric definitions must be compiled with the `ddfcomp` program at start of `ovnwpdc`. As result of the compilation, a logfile file set is created. The output of the PDC program must be piped to the `ddflog` program. The `ddflog` program processes the incoming data, using the log file set and integrates the data in the Embedded Performance Component agent database for collection and further use.

NDAOM Performance Data Collector (PDC)

Due to the possible varieties in the probe output, the performance data must be scanned and processed before integrating into the Embedded Performance Component. The NDAOM Performance Data Collector (PDC) handles these tasks and is installed as a subagent.

For generating detailed reports from every HOP (node in the network path) in HP OpenView Reporter or HP OpenView Performance Manager, incoming data must be split up into plain data records, with each record containing the full header information.

PDC Online Activities

The netpath probe traces performance data from the defined destination paths, generates an XML file and sends this file on the performance data port. The PDC runs as a subagent, permanently scanning the performance data port. When an XML file is detected, it is scanned for essential data, such as the hostname of the destination path. The transformed performance data from the XML file is logged in the Embedded Performance Component via a `ddflog` process.

PDC Configuration

The configuration of the netpath probe can change. For example, a path can be added or deleted by the user. There is no trigger event available which informs the PDC that something has changed. For this reason, the PDC reads the actual configuration via the configuration port once every minute and compares it with the existing configuration stored by the PDC and the tuple database.

Incoming performance data will be filtered before being sent to the Embedded Performance Component. One logfile is used for all destinations, no further action is required.

PDC Startup

At startup the following sequence runs

- Read the configuration via the configuration port.
- Generate the class specification file `ndaom.spec`
- Compile class specification file with `ddfcomp`, generating a logfile set.

PDC Output for HP OpenView Reporter

After processing the probe output in the NDAOM Performance Data Collector (PDC), the following data fields and data types are provided for the Embedded Performance Component.

Table 5-1 Data Fields in PDC Output for HP OpenView Reporter

Metrics Name in DSI Definition File	Description	Type	Precision	Length
NDAOM_SOURCE	Hostname of source node	Text		32
NDAOM_DESTINATION	Hostname of destination node	Text		32
NDAOM_PATH_ID	Netpath ID	Numeric	0	
NDAOM_HOP	Hostname of HOP	Text		32
NDAOM_HOP_NUM	Number of HOP	Numeric	0	
NDAOM_MIN	Minimum response time between source and HOP	Numeric	0	
NDAOM_MAX	Maximum response time between source and HOP	Numeric	0	
NDAOM_RANGE	Range of response times from min to max	Numeric	0	
NDAOM_MEAN	Mean value of response times between source and HOP	Numeric	4	
NDAOM_DEVIATION	Deviation of response times between source and HOP	Numeric	4	
NDAOM_MEDIAN	Median value of response times between source and HOP	Numeric	4	

Table 5-1 Data Fields in PDC Output for HP OpenView Reporter

Metrics Name in DSI Definition File	Description	Type	Precision	Length
NDAOM_MODE	Mode	Numeric	0	
NDAOM_MODE_COUNT	Mode count	Numeric	0	
NDAOM_COUNT	Internal counter	Numeric	0	
NDAOM_HOP_IP	IP address of HOP	Text		12

See Appendix B, “DDF Class Specification Files,” on page 133 for the examples of DDF specification files.

TIP For every HOP in the probe output, one record will be created.

PDC Output for HP OpenView Performance Manager

After processing the probe output in the NDAOM Performance Data Collector (PDC) the following data fields and data types are provided for the Embedded Performance Component.

Table 5-2 Data Fields in PDC Output for HP OpenView Performance Manager

Metrics Name in DSI Definition File	Description	Type	Precision	Length
NDAOM_DESTINATION	Hostname of destination node	Text		32
NDAOM_MIN	Minimum response time between source and HOP	Numeric	0	
NDAOM_MAX	Maximum response time between source and HOP	Numeric	0	

Table 5-2 Data Fields in PDC Output for HP OpenView Performance Manager (Continued)

Metrics Name in DSI Definition File	Description	Type	Precision	Length
NDAOM_MEAN	Mean value of response times between source and HOP	Numeric	4	

See Appendix B, “DDF Class Specification Files,” on page 133 for the examples of DDF specification files.

Overriding the Embedded Performance Component to Use MeasureWare with the nocoda.opt File

Performance metrics are collected by the Embedded Performance Component that is part of the OVO agents. The Embedded Performance Component performance counter and instance data from a variety of sources, primarily operating systems. The collected values are stored in a proprietary persistent data store from which they are retrieved and transformed into presentation values. The presentation values can be used by extraction, visualization, and analysis tools such as HP OpenView Reporter and HP OpenView Performance for Windows. You cannot extract/export, view, or aggregate the data directly on the managed node.

The Embedded Performance Component is the OpenView performance subagent that is automatically deployed with all OVO 7.0 agents. All Smart Plug-ins use the Embedded Performance Component as the default performance agent to store performance data for graphing in HP OpenView Performance Manager and HP OpenView Reporter. Previously installed OpenView products that use the MeasureWare agent will continue to use MeasureWare.

However, you may prefer to use MeasureWare as the agent for newer OpenView products in place of the Embedded Performance Component (for example, to use PerfView, which does not support the Embedded Performance Component). You can override the use of the Embedded Performance Component by setting up a simple text file that changes the default to use MeasureWare. This file, `nocoda.opt`, must be stored on the managed node in a specific location. Its location varies according to OVO management server and managed node operating system.

nocoda.opt File Location on Managed Nodes

HP-UX / Solaris	`/var/opt/OV/conf/dsi2ddf/nocoda.opt`
Windows	`\usr\OV\conf\dsi2ddf\nocoda.opt`

To log the NDAOM performance into MeasureWare instead of the Embedded Performance Component, you must insert the following two lines in the `nocoda.opt` file:

```
NDAOM

NDAOM_INSIGHT
```

You must restart the NDAOM subagent on the managed node after changing the `nocoda.opt` file in order for the changes to take effect.

NOTE If the `nocoda.opt` file is empty, NDAOM will log data into the Embedded Performance Component.

Performance Data

Performance Data can be divided into two different categories:

- Performance data from the node itself that runs the Netpath Probe
- Performance data gathered from the probe (via DSI)

  The link performance data is gathered from the netpath probe and sent on port 9874 (default). The data is sent in XML format. Times and numbers of send outs per hour vary over a wide range, depending on the probe configuration (e.g., on how many links should be checked by the probe). Size and content of the sent XML files depends on how many nodes were detected in the node path (HOPs) and varies between 1 HOP and 256 possible HOPs.

  The netpath probe XML output provides the following data fields in the context of performance measurement.

Table 5-3 Netpath Probe XML Output

XML Keyword	Description	Example
Header Information		
SOURCE:DNS	Host name of source node	bug.bob.hp.com
SOURCE:IP_OUT	IP address of source node	15.136.122.33
DESTINATION name	Host name of destination node	web.bob.hp.com
HOP Information		
IP_IN	IP address of HOP input	15.136.123.1
IP_OUT	IP address of HOP input, if different IP_IN	15.136.123.1
DNS	Host name of HOP	sv3.bob.hp.com
HISTORY_LIST	List of history items	
BASELINE	Performance data items	

Table 5-3 Netpath Probe XML Output (Continued)

XML Keyword	Description	Example
History Information		
Timestamp	Time of event	970215392
Ping	Response time in ms	71
Baseline Information		
MEAN	Mean value of response time	4.428
RANGE	Range between Min and Max	70
MIN	Minimum	1
MAX	Maximum	71
DEVIATION	Deviation	14.88
MEDIAN	Median value	1
MODE	Unknown	1
MODE_COUNT	Unknown	18

6 Reports and Integrating HP OpenView Network Diagnosis Add-On Module into HP OpenView Reporter

In this chapter you will find information on NDAOM reports and integrating the NDAOM into HP OpenView Reporter. The chapter is divided into two sections:

- Integration of the NDAOM into HP OpenView Reporter
- The available reports and their structure

HP OpenView Reporter Integration

The NDAOM is a plug-in package to HP OpenView Reporter. It modifies the metric list and reports sections. There is no automatic assignments between defined metrics and nodes. This is done either by the SPI developers or by the customer/administrator itself.

A new system group, `NDAOM`, is defined, which as default will be empty. Its is provided to contain all systems that belong to the corresponding node group of HP OpenView Operations.

NDAOM Reports

There are four types of NDAOM reports:

- Customized Standard Reports
- NNM Based Reports
- HP OpenView Performance Agent Data Reports
- NetPath Probe Performance Agent Data Reports

Specific NDAOM Reports

The following table lists the reports that can be reused from HP OpenView Reporter and customized for the `Network View` component.

Table 6-1 Reports That Can Be Customized for Network View

Category	Report Name	Data Sources	Report Type*
Applications	Message Trend by Application	OVO	All
Message Groups	Message Trend by Message Group	OVO	All
	Top Active Messages by Message Group	OVO	All
Nodes	Top Active Messages by Node	OVO	All
Node Groups	Message Trend by Node Group	OVO	All
	Node Assignment by Node Group	OVO	All
	Top Active Messages by Node Group	OVO	All
Services	Top Active Messages by Service	OVO	All
	Message Trend by Services	OVO	All

* Report type can be All, Group, or Single System.

NNM-Based Reports

Reports that are available with NNM:

`/opt/OV/conf/ServiceReporterInt`

were used to create a set of reports that use data from the NNM database. The following reports are available.

Table 6-2 NNM-Based Reports

Category	Report Name	Data Sources	Report Type*
Event Summary	NNM Events by Node Name	NNM	All
	NNM Event Trend Summary	NNM	All
	NNM Events by Severity	NNM	All
Event Details	NNM Events by Node Name (Details)	NNM	All
Inventory Summary	NNM Topology Summary	NNM	All
Inventory Details	NNM Topology Details - All Devices	NNM	All
	NNM Topology Details - All Network Devices	NNM	All
	NNM Topology Details - Analyzers	NNM	All
	NNM Topology Details - DMI Devices	NNM	All
SNMP Trend Summary	NNM SNMP Trend - Top 10 if%util Summary	NNM	All
SNMP Trend Details	NNM SNMP Trend - Top 10 if%util Details	NNM	All

Table 6-2 NNM-Based Reports (Continued)

Category	Report Name	Data Sources	Report Type*
Weekly Summary	NNM 7 Day Event Severity Trend	NNM	All
Weekly Summary	NNM 7 Day Event Category Trend	NNM	All

* Report type can be All, Group, or Single System.

NetPath Probe Performance Agent Data Reports

There is a set of reports based on the data delivered from the Netpath Probe. The reports reflect performance data from the single network links between two systems.

Table 6-3 NetPath Probe Performance Agent Data Reports

Category	Report Name	Data Sources	Report Type*
Link Performance	Detail Response per HOP	Embedded Performance Component via DDF	All
	Detail Response Times per Route		All
	Percentage of HOP used in any route to destination		All
	HOP Top 10 Response Times		All
	HOP Worst 10 Response Times		All
	Least percentages of routes used to reach destination		All
	Overview Average Response Times per top 32 routes		All
	Overview Average Response Times per worst 32 routes		All
	Table of routes used to reach destination		All
	Top percentages of route used to reach destination		All
	Response times comparison by worst ten routes		All

* Report type can be All, Group, or Single System.

7 Troubleshooting

In this chapter you will find information on:

- Basic troubleshooting
- How to set tracing for UNIX scripts

Basic Troubleshooting

If you are having problems with your NDAOM installation, as a first step to solving the problems, please check the following points:

- Check that the required patches are installed.
- Check that the NetPath Probes are running.

  You can check this with the command:

  `telnet <managed node> 9876`

  If telnet can connect to the NetPath Probe port, the NetPath Probe is running.
- Check that the configuration file is correctly configured. The file is located at:

  `/etc/opt/OV/ndaom/conf/ndaom.cfg`
- Check the tuple database (especially the flags). The tuple database is located at:

  `/var/opt/OV/share/ndaom/nwlmdb_sv`
- Check the NDAOM logfiles:

  — `/tmp/install_ndaom.log`

  — `/var/opt/OV/ndaom/log/install_nwagt.log`

  All log and trace files are stored in the `/var/opt/OV/ndaom/log` directory on both the management server and the managed nodes.
- Make sure that the database is unlocked using the following command:

  `ovnwlinkmon -unlock`

Set Tracing for UNIX Scripts

UNIX scripts on managed node systems can be traced using the HP OpenView Operations tracing facility, which helps you to investigate the causes of problems. For example, if processes or programs abort, performance is greatly reduced, or unexpected results appear. Trace logfiles can provide pointers to where and when the problem occurred.

Tracing can be activated for selected nodes specific management server and/or agent processes. To simplify the interpretation of the trace logfile, tracing can be activated for specific functional areas by specifying one or more functional areas in the trace statement. Activating Tracing shows how to use the trace statement, and Functional Tracing Areas gives a list of all available functional areas that may be used for tracing. Note that some areas are not available for some processes.

NetPath Probe and Monitor Tracing

Tracing from the NetPath Probe or Monitor:

1. Select system from node bank that you wish to have traced.
2. Double click the **Tracing On** application in **NDAOM** application group.

   This will enable tracing on the selected system and restart the probe in trace mode.
3. Check the `NP Probe Log` and `NDAOM Log` files on the selected system using the **View Tracefile** application from the `Applications` bank.
4. Analyze contents of these files for information on potential causes of problems.

Advanced script tracing can be set using the **set -x** entry on UNIX nodes.

The script can be modified directly on the managed node

or

by executing the command: **`ksh -x <script> <paras>`**

On Windows nodes, comment out the `@echo off` statement in the first line (using the comment identifier **`rem`**).

A NDAOM Configuration File ndaom.cfg

In this appendix you will find information on the variables that can be configured within the NDAOM configuration file, ndaom.cfg.

The NDAOM Configuration File

The NDAOM configuration file, `ndaom.cfg`, contains the configuration information for the NDAOM:

- executable files
- scripts
- applications

It is located in:

`/etc/opt/OV/ndaom/conf/ndaom.cfg`

Each line in the `ndaom.cfg` file can be a comment or a variable assignment.

Comment: every line beginning with a hash (#) is considered as comment and thus skipped during the parsing.

Variable assignment: This must be of the form:

<VAR>=<value>

where <VAR> is the name of the variable and <value> is the value that should be assigned to this variable. The value must not be enclosed by quotes. Blanks are not allowed before or after the equal sign (=). The keyword <VAR> must always be written in uppercase letters.

- **`BROWSER=s`**

  Defines the command to start a web browser that is capable of handling Java applets. For example:

  `BROWSER=/opt/netscape47/netscape`

- **`PD_SERVER=s`**

  The fully qualified domain name of the network node where the Problem Diagnosis (PD) server installation resides. Currently the PD product must be installed on the OVO management server. For example:

  `PD_SERVER=bug.London.mycom.com`

- **PD_SERVER_IP=xxx.xxx.xxx.xxx**

  Defines the IP address of the network node where the PD server installation resides. For example:

  PD_SERVER_IP=16.216.111.55

- **PD_SERVER_PORT=n**

  Defines the port number that the PD server uses to wait for information from its Probes. For example:

  PD_SERVER_PORT=9085

  The default value is 9085.

- **PD_MAIN_PATH=s**

  Defines the path to the PD server installation. The default is:

  PD_MAIN_PATH=/opt/OV/pd

  It can be changed during the PD installation process.

- **PERFMGR_SERVER=s**

  Defines the fully qualified domain name of the network node where the OV Performance Manager installation resides. For example:

  PERFMGR_SERVER=frog.London.mycom.com

- **TRACE_LEVEL=n**

  Defines the trace level to be used by the NDAOM executable files. The trace level can be an integer value between 0 and 9.

  0 corresponds to **no tracing** and 9 requests **maximum detail**. For example:

  TRACE_LEVEL=9

Table A-1 Trace Level Settings

Level	Trace Objects (cumulative)
0	Tracing is off
1	Major program actions
2	Major program events

Table A-1 Trace Level Settings (Continued)

Level	Trace Objects (cumulative)
3	Minor program actions
4	Minor program events
5	Function calls
6	Function parameters and results
7	Complete Embedded Performance Component dynamic data feed output of ovnwpdc
8	Complete TCP communication will be written to files (Filenames: xml.trc.xxx in the ndaom/log directory, where xxx is a number from 0 to 999). Every `send` or `receive` event creates a new file with incremented filename. After reaching file xml.trc.999 the next one will be xml.trc.000 and overwrites the old one.
9	Maximum detailed tracing. (Beware, this trace level produces very large trace files in a very short time).

- **`TRACE_AREA=S`**

  Defines the trace area that should be used when tracing is turned on (`TRACE_LEVEL > 0`). The trace area must be the name of an executable. For example `ovnwmonitor` or `ovnwpdc`.

  If a trace area is specified, then tracing is only turned on for this trace area.

B DDF Class Specification Files

In this appendix you will find an example of a DDF class specification files for each of the following:

- HP OpenView Reporter

- HP OpenView Performance Manager integration

DDF Class Specification File for HP OpenView Reporter

An example of a class specification file for HP OpenView Reporter.

```
CLASS C_NDAOM = 321
LABEL "NDAOM"
CAPACITY 48000
;

METRICS

NDAOM_SOURCE = 100
LABEL "NDAOM source"
TYPE TEXT LENGTH 32
;
NDAOM_DESTINATION = 101
LABEL "NDAOM destination"
TYPE TEXT LENGTH 32
;
NDAOM_PATH_ID = 102
LABEL "NDAOM PATH ID"
PRECISION 0
;
NDAOM_HOP = 103
LABEL "NDAOM HOP"
TYPE TEXT LENGTH 32
;
NDAOM_HOP_NUM = 104
```

```
LABEL "NDAOM HOP number"
PRECISION 0
;
NDAOM_MIN = 105
LABEL "NDAOM minimum"
PRECISION 0
;
NDAOM_MAX = 106
LABEL "NDAOM maximum"
PRECISION 0
;
NDAOM_RANGE = 107
LABEL "NDAOM Range"
PRECISION 0
;
NDAOM_MEAN = 108
LABEL "NDAOM mean value"
PRECISION 4
;
NDAOM_DEV = 109
LABEL "NDAOM deviation"
PRECISION 4
;
NDAOM_MEDIAN = 110
LABEL "NDAOM Median"
PRECISION 4
;
NDAOM_MODE = 111
```

```
LABEL "NDAOM Mode"
PRECISION 0
;
NDAOM_MODE_COUNT = 112
LABEL "NDAOM Mode count"
PRECISION 0
;
NDAOM_COUNT = 113
LABEL "NDAOM count"
PRECISION 0
;
NDAOM_HOP_IP = 114
LABEL "NDAOM HOP IP"
TYPE TEXT LENGTH 12
;
```

DDF Class Specification File for the HP OpenView Performance Manager Integration

An example of a class specification file for the HP OpenView Performance Manager (OVPM) integration.

```
CLASS C_NDAOM_INSIGHT = 322
LABEL "NDAOM for Insight"
CAPACITY 12000
;

METRICS

NDAOM_DESTINATION = 200
LABEL "NDAOM destination"
TYPE TEXT LENGTH 32
;
NDAOM_MIN = 201
LABEL "NDAOM minimum"
PRECISION 0
;
NDAOM_MAX = 202
LABEL "NDAOM maximum"
PRECISION 0
;
NDAOM_MEAN = 203
LABEL "NDAOM mean value"
PRECISION 4
;
```

Index

Index

O

P

Index

Index