HP OpenView Enterprise Discovery

for the Windows operating system

Software Version: 2.20

Reference Guide

Manufacturing Part Number: None
Document Release Date: April 2007
Software Release Date: April 2007

Legal Notices

Warranty

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

The information contained herein is subject to change without notice.

Restricted Rights Legend

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

Copyright Notices

Trademark Notices

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

Windows™ Vista is either a registered trademark or trademark of Microsoft Corporation in the United States and/or other countries.

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

UNIX® is a registered trademark of The Open Group.

Support

You can visit the HP software support web site at:

http://support.openview.hp.com/support.jsp

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

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

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

To find more information about access levels and HP Passport, go to:

http://support.openview.hp.com/new_access_levels.jsp

Contents

1 Introduction 9

2 How Enterprise Discovery Works 11

Introduction 11

What is ‘Discovery’? 11

Network Explorer. 13

Update Network Model 13

Model 14

Scanning Devices for Inventory Data 15

Scanning Devices for Application Utilization Data 15

Filtered Out of Database 16

Add to Device Database 16

Rulebase 17

Data Going Into the Rulebase 17

Creating a Device Type 17

Adding Rules to the Rulebase 18

Printers 19

Table Reader. 20

Generate Add Event. 20

Agent Deployment 20

Schedule Scan 21

The automatic deployment of Scanners. 21

Scheduling of scan execution 21

Collection and storage of scan files 21

Scan file enrichment 22

Poll Device 22

How long does all this take? 23

Enterprise Discovery is always discovering 23

Enterprise Discovery’s Topology Map 24

How much network bandwidth does the server need?. 24

Discovery Ping Rate 25

Table Reading and Polling 25

Scanning/Agent Deployment 25

Benchmarking Data 26

WAN 28

Aggregating Enterprise Discovery servers 30

What’s an Aggregator? 30

How do I use the Aggregator? 30

Aggregator data transfer 31
Connecting the Reports database on the Aggregator server 35
The Optimum time to run a Connect-It scenario 35
RFCs supported by Enterprise Discovery. 36
Communication models 37
Frame relay. 37
FDDI 39
HSRP. 40
Scheduled Events. 41
Software Library 41
Knowledge Updates 42
3 Terms and Concepts 43
Network Terms and Concepts 44
SNMP 44
MIB 44
Domain names 44
Address types 44
Netmask notation 46
Community strings/Users. 46
Bridge aging 47
OSI model layers. 48
Management workstation. 48
Virtual machine 49
Solaris zone 49
Enterprise Discovery Terms and Concepts. 50
The object label 50
Events and Alarms 51
Panel Elements 52
4 Virtualization in Enterprise Discovery 55
Overview. 55
Supported Virtualization Technologies. 56
VMware 56
Solaris Zones 57
Support for Virtual Devices in Enterprise Discovery 58
Discovering Virtual Devices on the Network 58
Scanning Virtual Devices 58
Virtual Devices in the Enterprise Discovery Database 61
Virtual Devices on the Network Map 61
Virtual Devices Window 61
Virtual Devices and the Device Manager 62
Virtualization Configuration Profiles 62
Reports on Virtual Devices. 63
Deactivation and Purging of Virtual Devices 63
5 Recorded Events. 65
Port Add/Deletes 65

Port Changes 65
Device Adds/Deletes 65
Device Changes 65
Exceptions 65
Not Recently Seen 65

6 Scanners 67
The Scanner Types 68
Viewing the Results of the Scan 69
Command Line Parameters and Switches 70
How to Use a Command Line parameter 70
Command Line Parameters for Scanners 71
Viewing Command Line Options in Viewer or Analysis Workbench 74
Using Command Line Switches to Enable and Disable Specific Hardware Tests 75
Starting the Scanners 76
Information Collected by the Scanners 76
Starting the Scanner Manually 76
Hardware Scan 77
Software Scan 77
Scanner Error Level Codes. 77
The MSI Scanner 79
Overview of the MSI Scanner 79
Starting the MSI Scanner 79
Opening the MSI Scanner Output File in the MSI Importer 79
MSI Scanner Error Level Codes 80
Troubleshooting 81
Using Error Level codes 81
Scanner Generator Errors 81
Hardware Scanning Errors 81
Software Scanning Errors 82
Scan File Saving Errors 82
Additional Errors 82
Using the Scanners for Manual Inventories. 84
Walkround Inventory 84
Using a Distribution Tool Such As SMS 84
Command Line Execution 84

7 Logging User Actions 85
Audit Log 86
Discovery Log 87

8 UI Shortcuts 89
Major Components 89
Asset Questionnaire 90
Device Manager 90
Port Manager 91
Line Manager 92

Attribute Manager 93
Service Analyzer 94

9 Copyright 95
ActivePerl 95
Apache Ant 95
Apache HTTPD 95
Apache Tomcat 96
Commons-logging 96
Commons-lang 96
cURL 96
Flex 96
GD 96
Getopt 97
LGPL 97
libicu 104
Log4j 105
mod-dav 105
Mod_Perl 105
MySQL 106
netaddr-ip 106
net-snmp 106
Open_SSL 108
Quartz 110
Unicode 110
Xalan 110
Xerces 110

Index 111

1 Introduction

Welcome to the Enterprise Discovery^{TM} *Reference Guide*.

This guide contains several chapters that you may find useful as you use Enterprise Discovery. Included are many definitions of terms and concepts used in Enterprise Discovery, as well as other reference materials that might help you work with the product.

2 How Enterprise Discovery Works

This section explains the basic functions of Enterprise Discovery. If you are a new user, we recommend reading part of it to help orient yourself with the product and allow you to have a better experience using Enterprise Discovery.

You can use Enterprise Discovery without ever having to read or refer to this section of the manual. However, even experienced Enterprise Discovery Administrators are likely to find it easier to understand the behavior of Enterprise Discovery after reading this section.

Introduction

Enterprise Discovery will provide you with an enormous amount of data about your network and devices. It can discover devices on its own by pinging and polling through a list of IP ranges that you provide, and it can also collect data from devices using scanners that you can customize.

With a full install of Enterprise Discovery, you will receive many tools to help collect, analyze, and report on your network devices. This chapter will explain some of the concepts that will help you understand Enterprise Discovery, and to help you maximize your benefit from using Enterprise Discovery.

What is 'Discovery'?

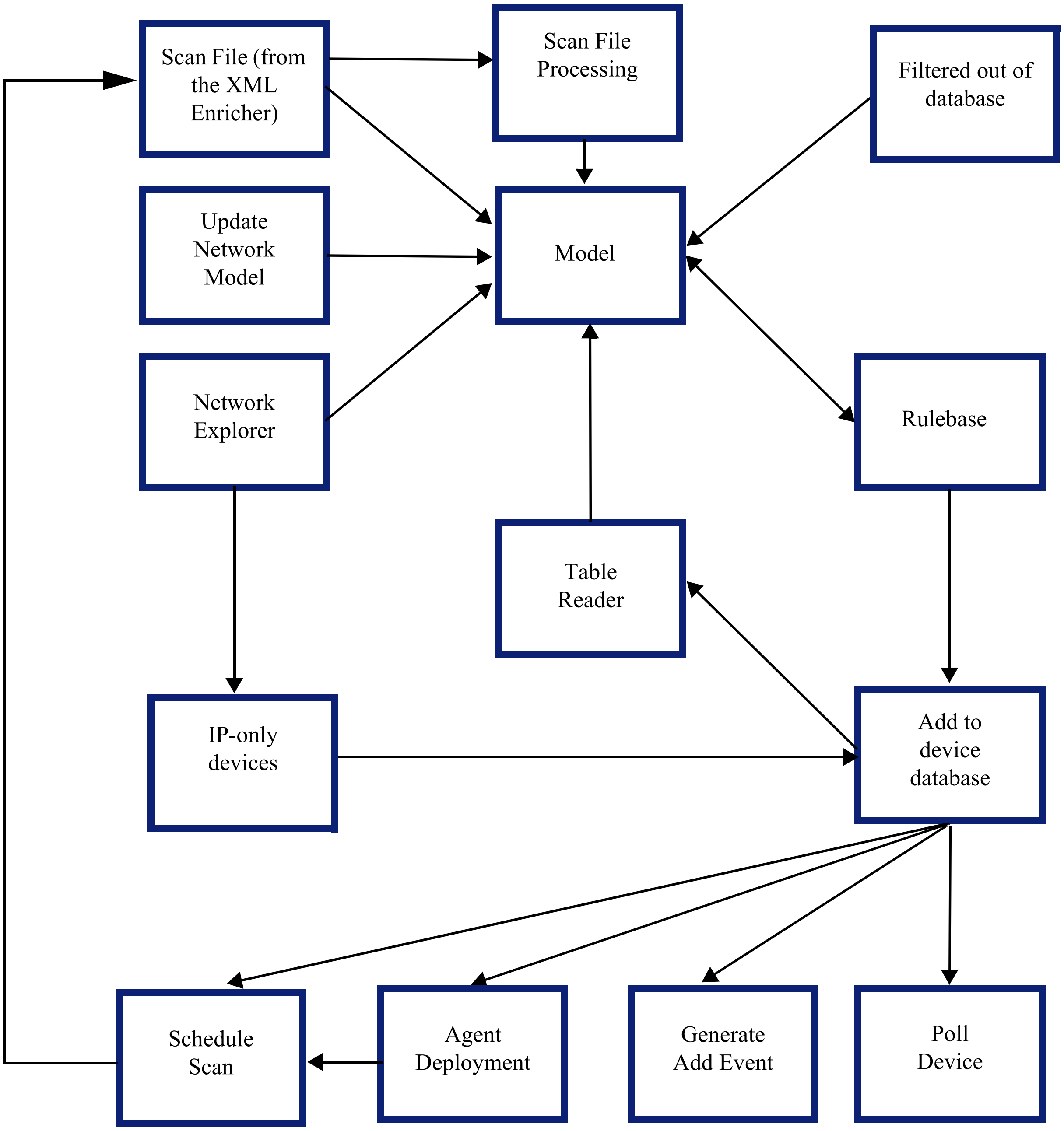
In order for Enterprise Discovery to provide you with network and device data, it must first know what devices are in your network, and what kind of data you want to see.

This process - discovering the network devices is called Discovery.

Once a device has been discovered, Enterprise Discovery creates a device model (a unique description of the device) and adds created device model to the Enterprise Discovery database.

Rules from a Rulebase are run against the model and some characteristics for the devices are determined based on the rule matches.

This flow chart shows a basic representation of how devices are discovered by Enterprise Discovery. Each section of the chart will be described below.



Network Explorer

Discovery is strictly a yes-or-no proposition. The Network Explorer goes through each IP address in random order and pings it once to see whether or not there is a response. Is there a device at this address or not? When there is a positive response, the IP address is added to the database (if it is not already in the database) and is sent to the modeler.

➤ By default, the Network Explorer will ping each address once. You can change the number of pings at **Administration > System Configuration > Network Devices > Number of Explorer pings**.

This means that the average time to discover a device can be calculated as the number of IP addresses in the IPv4 range divided by ping rate.

The Explorer works continuously to find any new devices in the configured IP ranges. For faster discovery, the Explorer also tracks devices that have responded positively and omits them the next time.

Update Network Model

The Update Network Model feature can be used in the following cases:

- To force the discovery of a new device that Enterprise Discovery has not found on its own. The Find command should be used to enter the IP address of a new device and the Update Model command should be issued from the right-click menu.
- To update a model for an existing device. This is helpful if you have made any physical changes to a device and want these changes to be picked up as soon as possible. The Find command can be used to locate the device and the Update Model command can be issued from the right-click menu.

Once the Device Manager for the device is opened, click the **Update Model** button and select **Query Network**. This command requests that the device have its network model updated immediately.

You can also update a model in **Administration > Data management > Update network model**. This command is especially useful in the case when an existing device has changed its IP address and the network model has not yet been updated to reflect this.

➤ The IP address used for model updates via the Device Manager is the primary IP of the device. If you want to use a different IP address use the **Administration > Data management > Update network model**.

There are two basic ways for Enterprise Discovery to discover devices:

Method	**Explanation**
Network Explorer	The Network Explorer is a component of Enterprise Discovery that pings your network looking for devices. You can configure the Network Explorer by setting up IP ranges (which control *where* it looks) and by setting certain preferences like the Explorer Ping Rate (which control *how* it looks). Note: When you prepare Enterprise Discovery for exploration, the Network Explorer begins by exploring the IPv4 ranges you have set up for "Active discovery" in **Administration > Discovery Configuration**. Whenever a device is found in that range, Enterprise Discovery will add it to the database.
Update Network Model	The Update Network Model feature allows you to add a network device to the database before it has been discovered by the Explorer or scanned with a Scanner. This is a manual process, usually done by using the Find command to enter the IP address of a new device. A warning is shown to indicate that the device is not available in the database, but the link to the new device appears. You can double-click on the device name to open the device manager. Alternatively, you can select the Update Model command from the right-click menu of the device name.

Model

Every device has a "device model" in the Enterprise Discovery database.

The two primary sources of data for the device model are the device MIB and the inventory scan file, the combination of which provides very good coverage of the infrastructure. The MIB is retrieved through SNMP and is normally available for core network devices such as routers, switches and bridges as well as network printers, etc. The scan file is generated by the scanner. Scanners are supported for a wide range of desktop and server systems.

Enterprise Discovery attempts to find out the device's community strings, its domain name, NetBIOS name, how many ports each device has, and what type of device it is—whether it supports bridge tables, arp tables, Cisco CDP, source address capture, and so on.

Enterprise Discovery will supplement that information with data from ARP caches from other devices (routers) in the network for unmanaged devices or if you have enabled the "accumulate IP addresses" in **Administration > Discovery Configuration**.

In some cases information from one device’s MIB is used to describe another device. For example if two Cisco devices are directly connected, the CDP information from one gives you information (like SysName, SysDescr, etc.) for the second one, even if the second one is not SNMP managed.

Scanning Devices for Inventory Data

Using Scanners is a powerful way to collect device data. Scanners are distributed to individual devices from the server using the Agent. You can scan each device for its hardware components, and to collect a list of the software installed. The server maintains a schedule dictating which computers should be scanned and when.

Automatic mode

➤ Deploy agents to all computers where you want automatic scan scheduling and scan file retrieval to take place. Refer to the “Setting Agent Deployment Accounts“ and the “Setting Up Scanner Schedules” chapters in the *Installation and Initial Setup Guide*.

When a particular computer needs to be scanned, the server contacts the agent on the computer, sends a copy of the appropriate Scanner configuration file and a copy of the latest Scanner to the computer (if the Scanner on the computer needs to be updated), executes it, and retrieves the resulting scan file.

Manual mode

In Manual Deployment mode the Enterprise Discovery Server is not used to schedule and launch scans. For example, if the scans will be launched from login scripts or on non-networked machines. The Scanner Generator generates self-contained Scanner executables that consist of a combination of the Scanner executable and configuration file.

If you are doing the inventory manually using manual deployment mode, you do not need the agent.

For both modes, retrieved scan files are stored in a directory on the server. The XML Enricher polls this directory for new scan files and processes them so they can be added to the server's inventory database. It also enriches the scan files with application data and stores these as compressed XML files.

For a complete description of what the XML Enricher does, and how to use it, see the *Configuration and Customization Guide*.

To learn how to generate scanners, also see the *Configuration and Customization Guide*.

Scanning Devices for Application Utilization Data

In addition, you can also enable the agent plug-in that collects software utilization information. The agent software utilization plug-in generates individual utilization files, one per day when it runs up to the maximum period for which utilization data is collected.

In addition, it also produces a summary file for the entire utilization period. This file is an XML data file compressed using gzip (Compressed XML utilization). The XML is encoded using the UTF-8 encoding.

The XML Enricher does the following during its processing:

- Extracts and parses the XML data out of the stored file.
- Calculates the software utilization for each recognized application and adds this information to the enriched scan file.
- Adds a 'Utilized' flag to the file attributes, calculates and adds utilization figures for executables that were executed.

Filtered Out of Database

Enterprise Discovery can filter out certain unmanaged devices that you do not want to see on your map or in your Enterprise Discovery database.

You can change these filter settings in **Administration > System Configuration > Input filters > Devices not to add to the database**:

- Unmanaged devices which are MAC plus IP
  - Unmanaged devices which are MAC plus IP and not pingable
- Unmanaged devices which are MAC-only
  - Unmanaged devices which are MAC-only with unknown OUIs
- Unmanaged devices which are IP-only
- Scanned-only devices
- Virtual-only devices

All of these filter settings change which devices can be monitored by Enterprise Discovery. If a device is filtered so it does not appear on the Network Map or in the database, the device data will be found at **Status > Device Status > Filtered Devices**.

Add to Device Database

Once the device has been modeled the device is added to the Enterprise Discovery database.

Enterprise Discovery publishes the data about the devices it has discovered (i.e. the models) in its Discovery Database. This database is hosted on the Discovery server and is managed by a MySQL server that is embedded with Enterprise Discovery.

The Discovery Database is used to generate all of the reports that can be accessed through the User Interface, and should also be used to populate the asset management database. Using the scenario provided with Connect-It 3.6 or later, transferring the Discovery data to AssetCenter is a simple matter.

The database schema is fully documented (the documentation is available through the UI) and is not user-configurable.

Rulebase

The Rulebase is the component of Enterprise Discovery that assigns a device type to your device model. The Rulebase is a large database containing information on hardware available for sale by a wide range of manufacturers. When Enterprise Discovery has a model, it will search the Rulebase to find that particular manufacturer and other pertinent information.

The Rulebase is responsible for giving each device an appropriate icon (as seen throughout the Enterprise Discovery interface), and making sure the basic device information is up to date.

➤ Rulebase updates are made available on a regular basis. Check the support website for updates.

Data Going Into the Rulebase

The following fields are passed to the Rulebase so it can make a determination about the device:

- system.sysDescription
- system.sysObjectID
- DNS name
- MAC address
- NetBIOS name
- forwarding information (whether or not the device is routing IPv4 or IPv6)
- read community string
- write community string
- number of ports

The following fields are available for scanned devices:

• Operating System	hwHostOS
• BiosProductName	hwBiosMachineModel
• BiosProductManufacturer	hwsmbiosSystemManufacturer
• BiosChassis	hwsmbiosChassisType
• BiosDescription	hwBiosMachineDescription
• Operating System ServicePack	hwOSServiceLevel

Creating a Device Type

The Enterprise Discovery Rulebase then assigns a device type to each device. The device type can include the following characteristics:

- Model
- Model URL
- Model Manufacturer
- Family

- Family URL
- Family Manufacturer
- OS
- OS URL
- OS Manufacturer
- Network Function
- Network Function URL
- Network Function Manufacturer
- Software Historical Manufacturer
- Software Historical Manufacturer URL
- Software Present Manufacturer
- Software Present Manufacturer URL
- Historical Manufacturer
- Historical ManufacturerURL
- Present Manufacturer
- Present Manufacturer URL
- Icon
- Title
- Tag
- Priority
- UNSPSC model
- UNSPSC description
- UNSPSC application
- UNSPSC operating system

Adding Rules to the Rulebase

If a specific rule is not in the Rulebase, Hewlett-Packard will add a rule for you, provided that:

- your Enterprise Discovery software is under warranty
- you can identify the devices by model or family (Hewlett-Packard would appreciate the URL for the manufacturer's web site whenever possible)
- you provide Hewlett-Packard with a CSV copy of your inventory containing the device.

Some devices and device families may not match a specific identification rule. Such rules are likely to make good assignments for companies with small product lines and less accurate assignments for companies with large product lines. This is because these rules are based on:

- advance classification of product lines; that is, some devices belonging to certain product lines can be identified by the beginning of the OUI

- pre-identification of specific devices; that is, some devices can be identified because the manufacturers make only switches

➤ These rules may make incorrect assignments. You should contact Hewlett-Packard to request additional specific rules for devices if this happens.

For devices with no SNMP management, the Rulebase can apply rules based only on information about the MAC address and OUI of the device. For each MAC address, the Rulebase identifies the most probable device class, based mostly on the OUI. (Occasionally, manufacturers assign blocks of MAC addresses to specific products, which allows the Rulebase to make more specific identifications.)

The Rulebase also identifies the probability that each non-SNMP device may actually be an SNMP managed device providing network connectivity (such as a gateway, router, concentrator, or switch). SNMP managed devices can appear not to be managed when the device's IP address has been included in the list of Property Groups (see **Administration > Network configuration > Network Property Groups**) or when the community string for the device has not been included in the list of community strings (SNMPv1/v2) and users (SNMPv3) (see **Administration > Network configuration > SNMP Property Groups**). In such a case, you should install or enable the SNMP agent for that device. (You may also need to modify the address scope or community strings.)

As with devices with SNMP management, class assignments for devices with no management work well for companies with small product lines and poorly for companies with large, varied products lines. Larger companies sometimes employ the same OUI for different products, but also use different OUIs for one product.

There is also a capacity to assign icons to unmanaged devices based on information contained in the NetBIOS and domain names deleted. For example:

- a device named “PRINTER3RDFLOOR” or “PRT3RDFLOOR” could be assigned a printer icon
- a device named “marysworkstation” could be assigned a workstation icon
- a device named “webserver.example.com” could be assigned a web server icon

Only the initial segment of the domain name is considered.

Domain and NetBIOS name interpretation is low priority. It never takes precedence in a situation where more accurate information is available. The rules used are not case sensitive.

In some cases information from one device's MIB is used to describe another device. For example if two Cisco devices are directly connected, the CDP information from one **get** gives you information (like SysName, SysDescr, etc.) for the second one, even if the second one is not SNMP managed. This could be used to assign icon/device type to an unmanaged device because we have rules based on information collected from “remote” devices.

Printers

Finally, Enterprise Discovery can identify printers attached to printer servers. Many printer servers (both internal and external) do not provide enough information in their System Description to allow for accurate identification of the specific model of printer attached.

The Enterprise Discovery Rulebase uses information that may be found elsewhere in the device MIB. For example, the System Description of this Hewlett-Packard printer server contains the following:

- HP ETHERNET MULTI-ENVIRONMENT,ROM H.08.01,JETDIRECT EX,JD34,EEPROM H.08.05

Note that this does not provide any information about the printer. The Enterprise MIB contains additional information that allows the Rulebase to identify the printer server as J3263A and the printer model as a HP LaserJet 5.

Table Reader

➤ The Table Reader applies only to SNMP-managed devices.

Unmanaged MAC-only devices (and MAC-only devices with unknown OUIs) will be discovered after Enterprise Discovery completes modeling devices with bridge table reader.

Devices can also be discovered as a MAC+IP pair from information collected from router arp caches.

Connectivity information comes from the Table Reader. If Enterprise Discovery has identified the device as a bridge, the Table Reader reads its bridge tables. If the device has been identified as a router, or if "Force ARP Table Read" has been enabled in **Administration > Discovery Configuration**, the Table Reader reads its ARP table.

Generate Add Event

As soon as the device is added to the database, a "Device Add" event is generated to alert you that Enterprise Discovery is now monitoring the device. You can see the event listed in the Events Browser (see the *Network Data Analysis Guide*).

Agent Deployment

The agent is installed as a service on a remote computer. This service enables the computer to be securely scanned at any given time. In Windows NT®, 2000, 2003, and Windows® XP, a manager can instantly deploy this service on computers using a service like Windows Remote Administration (RPC).

On Unix a compress tar archive is provided for each platform to allow manual or scripted installation.

The agent is able to perform tasks on a computer on behalf of Hewlett-Packard applications.

- For security reasons, agent communications are encrypted and authenticated.
- The agent listens and performs requests for Enterprise Discovery. For example, it can deploy a Scanner, execute a scan, or transfer a scan file to the server.

➤ A newly discovered computer cannot be scanned without first installing the agent.

The agent must be installed on every workstation that will be part of the inventory process. If you are doing the inventory manually, you do not need the agent.

Once installed, the agent is capable of communication with the server. The communication can only be initiated by the server. The agent is not able to initiate any file transfers, scans, etc.

If an Agent is manually removed from a computer, Enterprise Discovery can detect that and automatically redeploy the Agent.

Each agent originating from a server will have a security key from that server. This means that the agent will only be able to communicate with that server. However, it is possible to share the same security keys between servers. See the *Installation and Initial Setup Guide.*

Further information

For more information on agent deployment, see the *Installation and Initial Setup Guide*.

For information about the platforms the agents are available for, see the *Compatibility Matrix*.

Schedule Scan

Schedule Scan is a feature that allows you to control when the scanners are run on the workstations.

For more information on creating scan schedules, see the *Installation and Upgrade Guide*.

The automatic deployment of Scanners

In most cases, once an Agent has been installed on the computer, the Scanners can be automatically deployed as that computer is discovered with Enterprise Discovery, and executed as needed.

This mechanism makes Scanner deployment an easy task, as opposed to situations where deployment has to rely on network login scripts or manual intervention. Thus, the accuracy and completeness of the collected inventory data can be very high.

Scheduling of scan execution

It is possible to specify a schedule for computer scanning. The schedule serves at least two purposes.

- First, although the execution of a scan is designed to be as unobtrusive as possible to the user of a computer, some users do notice and find it distracting. So scans can be scheduled to run at a time of day that tends not to conflict with users.
- Second, the accuracy of the inventory depends on the frequency of scan execution. For example, some users want the data refreshed every week, others every month. So the frequency of scans can be specified.

Collection and storage of scan files

A Scanner writes a scan file to the local disk of the scanned computer, and the scan file is transferred to the server for storage and processing. There are a few ways in which the scan file can be transferred:

- The server contacts the computer and transfers the scan file from the computer. This is the typical case for computers that are permanently connected to the network. The collection of scan files is scheduled and controlled to minimize impact on the network. For example, you can specify what times of day are appropriate for scan file collection, how many files can be transferred in parallel, and how much network bandwidth scan collection is allowed to consume.

  ➤ Collection of scan files is decoupled from the execution of a scan. You can schedule scans for one time of day, but collect the scan files some time later using a different schedule.

- Some computers, for example laptops, are only occasionally connected to the network. In this case, the scan file can be transferred whenever that computer connects. Since the connection speed may be slow and the connection time short, the transfer mechanism gracefully recovers when the connection is interrupted and can be resumed when the connection is re-established.
- Some computers may never be network accessible to the server, for reasons of network topology or security. In this case, it is the responsibility of the administrator to transfer the scan files from such computers to the server.

Scan file enrichment

Once a scan file is transferred to the server, it is further processed to recognize software applications (XML enrichment process) and added to the inventory information stored in the Inventory Database. The resulting enriched scan file is stored on the server for subsequent access by tools such as Viewer, Analysis Workbench or Connect-It.

This enriched scan file is always stored in compressed XML format. At most one scan file for each computer is stored, and the name of the scan file is normally derived from the Asset Tag uniquely identifying the machine. The enriched scan files are optionally backed up along with other server data.

Further information

For further information about the XML Enricher, refer to the *Configuration and Customization Guide*.

Poll Device

➤ Scanned-only devices are not polled.

Once there are device models in the database, the pollers begin collecting data from the devices. There are three pollers: the Realtime Poller, the Resource Poller, and the Environmental Poller.

The Realtime Poller also reads the device’s MIB to get information about the device’s traffic and connectivity on all of its ports. The resulting data is passed to the Mapper to determine connectivity.

How long does all this take?

Generally, “discovery” can work like this:

- A device will be discovered based on the discovery ping rate and the IPv4 ranges you have set.
- The time taken to create a device model will vary depending on the type of device, and the number and order of the community strings.

The time may increase for some devices if Enterprise Discovery needs to try several community strings (or users for SNMPv3) to access the device MIB.

Enterprise Discovery also uses scan files to supplement a device model, and that process is quite different. Once the device has a model in the database, an Agent can be deployed to that device. After the Agent is deployed and the Enterprise Discovery server can contact it, a Scanner is deployed. The time may increase depending on how many Agent Deployment Accounts are configured to access the devices.

There are many factors that affect the time it takes to scan a particular device. The scanner configuration is a major factor. Hardware-only inventories take from a few seconds to a couple of minutes to complete depending on the configured hardware tests and selected scanner priority. On the other hand, the software scan usually takes much longer and depends on the chosen configuration, the scanner priority, and the number of files and directories available on the computer to be scanned. Other factors include the speed of the computer and its hard drive. As a rough estimate, a scan of an average workstation using the default scanner (hardware and targeted software) takes around ten minutes.

The Enterprise Discovery server is able to communicate with up to 76 agents at the same time. Server to agent communication includes uploading scanners and their configuration, starting scanners, and collecting scan file results. However, these communications are usually brief, so the server is able to launch many more than 76 simultaneous scans at the same time. The overall time it takes to scan a network depends on how long it takes to scan each computer and on other factors, such as available bandwidth. It is recommended that you perform a pilot inventory that scans a subset of all managed devices in a particular environment using configured settings to get a better idea of the exact times required.

Enterprise Discovery is always discovering

This process runs the entire time Enterprise Discovery is in operation.

Also, every device is re-modeled at the device remodeling interval specified in **Administration > Discovery configuration**.

This way, Enterprise Discovery constantly strives to present you with an updated view of your network, and constantly strives to improve the accuracy and depth of that view.

Enterprise Discovery's Topology Map

To calculate network connectivity, Enterprise Discovery uses a probability engine with a variety of patented algorithms.

On each device, Enterprise Discovery considers the following:

- the source address capture information
- link training
- bridge tables
- vendors' proprietary tables (including Cisco's CDP)
- VLAN bridge tables

Bridge tables are huge and Enterprise Discovery needs a lot of computation power to analyze them. Enterprise Discovery will delete most of the information in the raw bridge tables and retain the information needed to build a virtual bridge table.

At this point, Enterprise Discovery looks at all the bridge tables and figures out which ports are up ports (one network device with bridge tables connected to another network device with bridge tables), and which ports are down ports (network devices with bridge tables connected to other devices without bridge tables).

Once the ports have been classified as up or down, the bridge tables have been reduced so the computing required for determining connectivity gets much simpler. Typically Enterprise Discovery deletes 95% or more of the information in a bridge table. From this, Enterprise Discovery generates port-to-port connections.

Enterprise Discovery also uses traffic patterns to determine connectivity. Enterprise Discovery is self-similar or fractal, so all the traffic patterns are different.

Traffic pattern matching works by matching the traffic going in and out of one interface, against the traffic going in and out of the other interface. This data is automatically correlated. Based on probability Enterprise Discovery will determine connectivity between two devices once a threshold is met, and then say that two ports are connected based on traffic information. Enterprise Discovery also uses sorting techniques to reduce this intractable N^2 problem down to a usable N*Log(N) computation. Additional heuristics and table logic are introduced to clean the data and optimize the success of resolving connectivity.

How much network bandwidth does the server need?

There are many factors that contribute to the network traffic caused by Enterprise Discovery. The best practice is to connect the server to a major backbone switch. It is estimated that the traffic would total 3-4% on the 10MB dedicated link between the server and the switch. On a 100MB or 1GB link, the impact is proportionally smaller.

The traffic initiated from the server is heaviest on that link to the switch. From the switch, the traffic going to the network is dispersed. It is impossible to say exactly how much bandwidth will be taken by Enterprise Discovery, but in this section, you can read about some of the influences you may want to consider.

Enterprise Discovery must contend with many different types of devices, each of which will contain varying amounts of data to be collected. Also, the many settings in Enterprise Discovery can change how often the data is collected. This makes it difficult to offer an idea of how much bandwidth will be needed in any particular situation.

Also, on the Device Manager, check some of the Statistics graphs such as:

- SNMP Bytes
- SNMP Frames
- ICMP Frames

Discovery Ping Rate

The Discovery Ping Rate (**Administration > System Configuration > Network devices**) is one source of traffic. The ping sweep occurs in the background to look for new devices that have not been found previously. If you turn the ping rate down, it will take longer to discover new devices in your network.

If you turn Enterprise Discovery's Ping feature off (**Administration > System Configuration > Discovery services**), new devices will not be found through this method, the active part of the discovery will not generate any traffic on your network.

If you have configured Enterprise Discovery to ping a large IPv4 range containing very few devices, there may be some network impact as pinging non-existent IP addresses will cause ARP broadcast requests.

Enterprise Discovery has been configured to limit the ARP broadcasts it generates. However, Enterprise Discovery may ping devices on the far side of a router. You should check your router configuration if broadcast levels become unacceptable for your network. On your router, consider the following:

- increase ARP cache size
- increase ARP aging time
- reduce ARP retry rate

Table Reading and Polling

Table reading and polling produce the majority of network traffic from the Enterprise Discovery server. These functions provide:

- connectivity information
- discovery of devices (for example, MAC-only devices)
- collection of statistics to help with finding the topology

A poll is really one frame out and one frame back in most cases. The number of polls for a device will depend on the number of ports in the device, and the number of attributes collected for each port (for example, collisions, broadcasts, etc.). The device itself is also pinged in each poll cycle.

Consider the fact that collecting statistics on a router with 200 ports requires a lot more effort than collecting statistics from a workstation with one port.

Scanning/Agent Deployment

Initial Agent Deployment

Initial automatic agent deployment is available for Windows NT/200x/XP. The Windows agent installation is around 1MB in size (the exact size can be seen by looking at the size of the agent .msi file located in the LiveAgents subdirectory of the ED data directory) – it is transferred to each computer where the deployment is taking place. The server can run up to 50 simultaneous agent deployment sessions at any one time. The exact number used (default is 25) is specified **Administration > System Configuration > Agent communication > Agent deployment concurrent sessions**.

Custom automatic deployment can be done for other platforms by writing a custom deployment script. For example, agent installation on UNIX may be done via SSH or RSH, etc. The exact bandwidth used by this method depends on the detailed implementation.

Agent upgrades / Scanner Deployment / Scanning

When performing any of these activities, by default the ED server can communicate with a maximum of 80 agents at any one time. This maximum number can be configured in **Administration > System Configuration > Agent communication > Agent deployment concurrent sessions**.

- Agent upgrade: when doing an agent upgrade the agent media (an .msi file for Windows and .tar.Z file for UNIX) are transferred to the remote computer. The size of these files depends on the platform and can be seen by looking at the live agent media directory (the LiveAgents subdirectory of the ED data directory).
- Scanner Deployment: when the scanner has not been deployed yet or an old version of the scanner is available on the remote computer, the new copy of the scanner is deployed. The scanner size varies depending on the platform – the exact sizes can be seen by looking at the Scanners subdirectory of the ED data directory.
- Scanning: unlike agent and scanner deployments/upgrades, which are usually one-off events, the scanning activity involves the ED server asking the agent to run the scanner regularly according to a specified schedule. After the scanner has finished executing, the scan file is saved locally and then transferred to the server. The size of the scan file varies depending on the size of the box and how the scanner is configured.

The network bandwidth used for scanner deployment/upgrade and the scan file retrieval can be capped for each ED server to agent connection – this is specified in the scanner property group.

Benchmarking Data

The Enterprise Discovery team has conducted tests to benchmark statistics about network traffic caused by the Enterprise Discovery server.

This test was conducted in a lab with eighty machines. The test began with 20 machines attached to the network and available for discovery. Then, 24 hours later, 20 additional machines where added. Finally, after 24 more hours, 40 additional machines were added, for a total of 80 machines attached to the network.

This incremental approach allowed the test team to gather statistics about the increase in network traffic vs. the number of machines on the network.

Description of the two test series

Two series of tests were completed.

Test series 1 captured packets when Enterprise Discovery was set to simply discover devices in the network. All devices had SNMP enabled.

Test series 2 enabled the deployment of Enterprise Discovery agents and scanner execution, along with the network discovery capabilities.

The benchmark tests were considered successful when:

- SNMP enabled on all of the devices
- A full Discovery and deployment occurred
- Scans were created and retrieved.

Reporting methodology

The reports consist of metrics from the Ethereal 0.10.0.3 network sniffer utility. Data from the Ethereal utility was then exported to Microsoft Excel, and calculations were made on that data.

Network test results

When the number of devices on the network doubled (from 20 to 40) with the second test iteration, the network traffic statistics did not double; the overall number of bytes transferred increased by only 20%. This shows that Enterprise Discovery is scalable in a way that does not proportionally impact the volume of network traffic.

The average packet size almost doubled when the number of devices was increased in the third iteration (from 20 to 80 devices).

Table 1 Network Test Series 1

Statistic	Value with 20 devices on the network	Value with 40 devices on the network	Value with 80 devices on the network
Between first and last packet	86,412.572 sec	86,415.326 sec	86,414.411 sec
Packets Captured	436,445	478,244	310,000
Avg. Packets/sec	5.051	5.534	3.587
Avg. Packet size	68.000 bytes	78.000 bytes	137.000 bytes
Bytes	29,885,826	37,573,212	42,654,214
Avg. bytes/sec	345.850	434.798	493.601
Avg. Mbit/sec	0.003	0.003	0.004

Scanner test results

When scan files are added, the number of bytes transferred on the network grows greatly. There is almost a 500% growth in total bytes compared to the test that did not include scan files. The average packet size also grew close to 500% as compared to the non-packet test.

Comparing the results from this test alone, the growth in network traffic and packet size does not grow in proportion to number of devices on the network.

The growth when the number of devices doubled on the networks was about 30%. Packet size growth was close to 10% when the number of devices doubled.

Table 2 Network Test Series 2

Statistic	Value with 20 devices with scanner deployment	Value with 40 devices with scanner deployment	Value with 80 devices with scanner deployment
Between first and last packet	86,415.804 sec	86,413.504 sec	86,416.253 sec
Packets Captured	770,696	938,701	998,428
Avg. Packets/sec	8.918	11.295	12.052
Avg. Packet size	171.000 bytes	195.000 bytes	221.000 bytes
Bytes	132,384,807	183,085,794	221,580,176
Avg. bytes/sec	1531.951	2202.967	2674.692
Avg. Mbit/sec	0.012	0.018	0.021

WAN

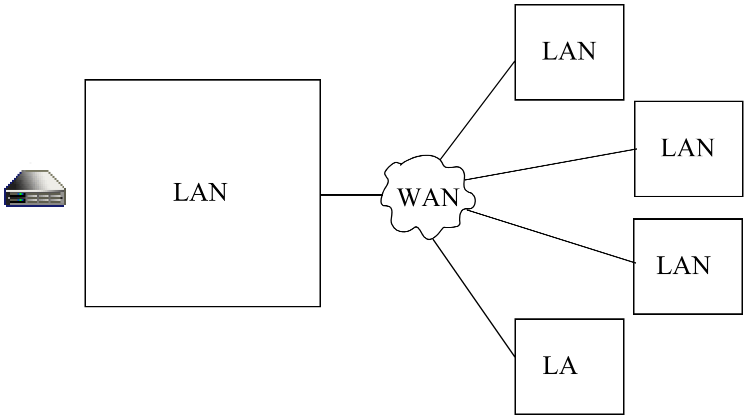
The bandwidth on a WAN link depends on the number of devices in your IPv4 ranges, the types of devices, and the many settings available in Enterprise Discovery.

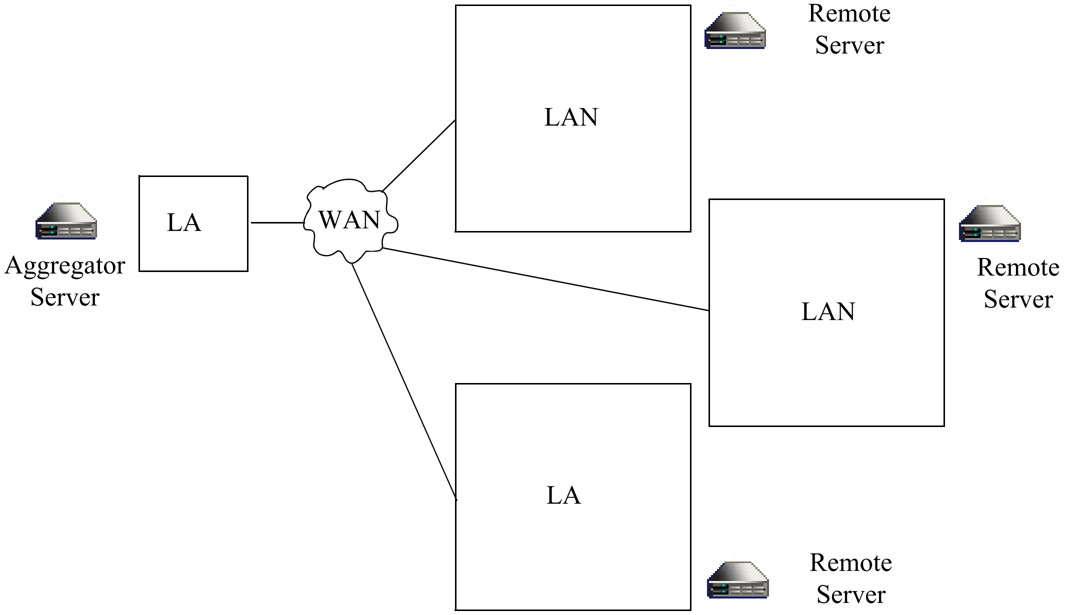
However, the real impact depends on the amount of bandwidth available, how much is used by normal network traffic, and what levels of extra traffic you are willing to accept.

Over a slow link, it would not be practical to fully manage a large network. Depending on the various parameters, you may only want to monitor the core devices. If you need to manage your entire network, it may be wise to get a second Enterprise Discovery server on the other side of your WAN link. The two servers could be aggregated and share data.

There are many other considerations in a WAN scenario. In a large network (for example, 10,000 devices) with high rates, assume that the Enterprise Discovery server will use 5% of a 10MB link for network management.

Using Enterprise Discovery in a WAN:





Aggregating Enterprise Discovery servers

What's an Aggregator?

The Aggregator is a Enterprise Discovery server with a license that also allows it to collect and combine data from several Enterprise Discovery servers in your network. The health data is combined into one Aggregate Health Panel, so you can see the status of the entire network. An Aggregator also allows you to access other individual Enterprise Discovery servers without logging into them directly.

You can aggregate up to 50 Enterprise Discovery servers with a maximum of 500,000 devices. However, the more servers you aggregate, the more slowly the Aggregator processes the data. While performing as an Aggregator, the Enterprise Discovery server can also serve as a regular server, monitoring up to 100 devices.

It is important to remember what is aggregated, and what is not. The following functions are aggregated:

- Health Panel
- Alarms Viewer
- Events Browser

Also, with an Aggregator, you have an integrated data source for exporting onto data access applications using the Open Database Connectivity Standard (ODBC). See the *Network Data Analysis Guide* for more information.

➤ If a remote server is not available, the Aggregator uses the last available imported Health Panel for that remote server. You can detect the status of a remote aggregator server by navigating to **Aggregate Status > Aggregate server health**.

How do I use the Aggregator?

An Aggregator Enterprise Discovery server works like a regular Enterprise Discovery server. The Aggregator has an extra license that allows it to collect data from the other Enterprise Discovery servers in your network. The Aggregator can also be responsible for monitoring a specific part of the network, while simultaneously collecting data from other Enterprise Discovery servers and presenting them in the Aggregate Health Panel.

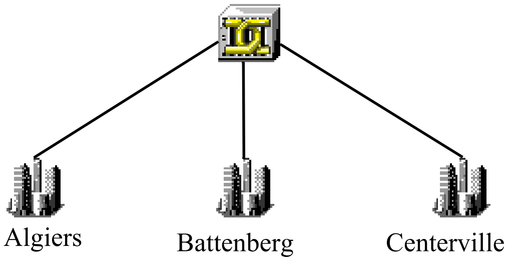
There are many ways you could set up aggregation in your network, depending on the network topology and how many Enterprise Discovery servers you have installed.

- You can use the Aggregator as a regular server to monitor a part of your network, as you would with any of your Enterprise Discovery servers.
- You can use the Aggregator as a regular server to monitor only the backbone of your network, your important routers and servers, as well as the other Enterprise Discovery servers.

If you have the resources available, we recommend the second option. You can use the other servers to monitor the subnets, but this will give you a real center point from which you can access the rest of your network.

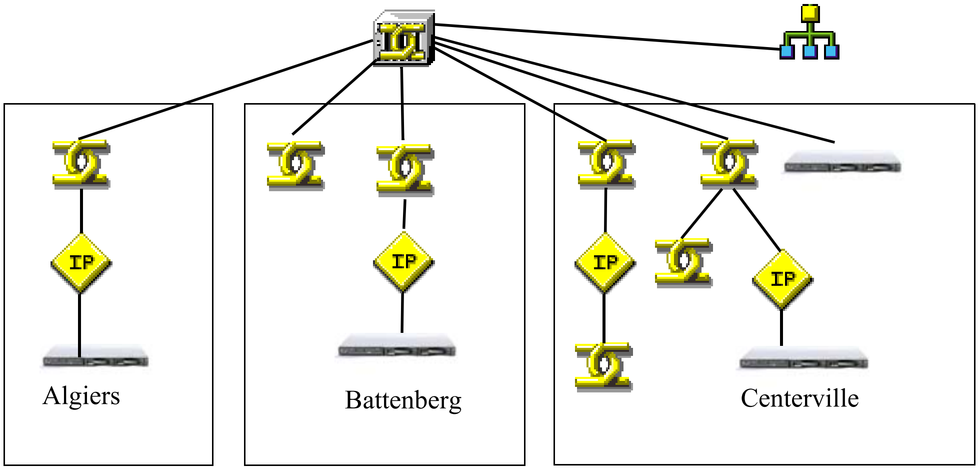
Set-up Example

Suppose that you work with a business, ExampleCorp, that has offices in three cities: Algiers, Battenberg, and Centerville. Each office has 6,000 devices in its subnetwork.



Ideally, you would have purchased 4 Enterprise Discovery servers: one for each office, and one to act as an Aggregator for the central office (in Centerville).

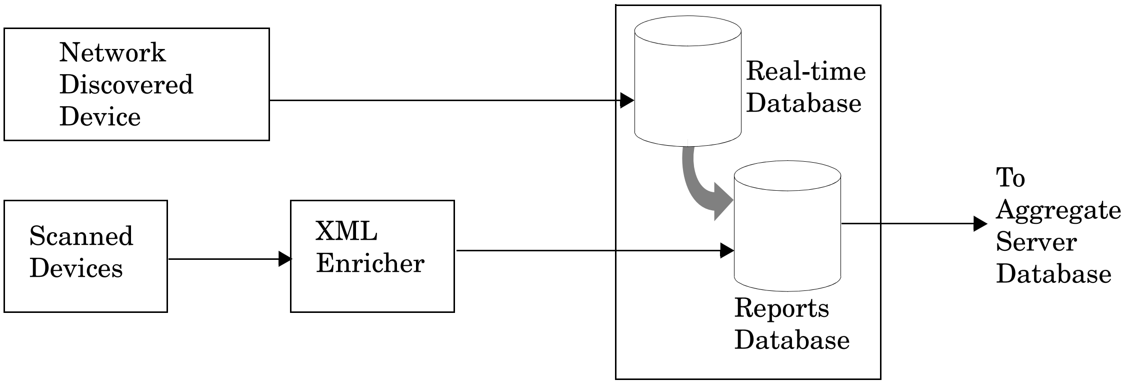
If you set up the Aggregator ranges to include only Enterprise Discovery servers and routers, the resulting network may look like this:



Aggregator data transfer

In order to understand what information is made available from the remote Enterprise Discovery servers, you will need to understand the configuration and data transfers on the remote server.

The following diagram illustrates the internal data transfer for each remote server configured before the data is processed to the Aggregate database server.



Allowing access to the Real-time database could cause performance issues. Therefore, in order to make the Enterprise Discovery data accessible, it is necessary to transfer the data into a separate database (Reports database) where it can be accessed at any time without impacting the performance of Enterprise Discovery.

The information collected by the Inventory Process is device specific and only needs to be added to the device's model information in the Reports database. Information from the Inventory Process is used to merge the information with the information that was collected by the Discovery Process.

Data transfer from the Real-time database to the Reports database

There are five categories of data transferred from the Real-time database to the Aggregate database. Each category of data has its own schedule for being imported into the Reports database. For each category, data is imported in two phases:

- **Extract:** The relevant data is extracted from the Real-time database and passed on to the Reports database. The extracted data is stored in csv files.
- **Import:** At the scheduled time, Enterprise Discovery will import data into the Reports database.

Events

The amount of data and the frequency that it is imported into the database is configured in:

Administration > System Configuration > Attribute export schedule

The default frequency is 5 minutes. By default, data is only imported into the database when it is in an alarm state.

- **Extract:** The extract is scheduled to occur every 5 minutes starting on the hour.
- **Import:** The import is scheduled to occur every 5 minutes starting on the hour.
- **Tables Updated:** Attribute, AttributeState, ConnectionEvent, Event.

Hourly Summary

This information is used to report statistical information on the devices in the network.

- **Extract:** The extract is scheduled to occur every hour on the hour.
- **Import:** The import is scheduled to occur every hour on the hour.
- **Tables Updated:** HourlySummary

Hardware Discovery Data

This data is imported into the Reports database once a day at set schedules. Hardware Discovery Data can be imported into the database on demand from:

Administration > Data Management > Update discovery database

This action performs both the extract and import of the data.

- **Extract:** The extract is scheduled to occur at 6:00 AM daily. Two copies of files are produced one for local consumption and a second copy for the Aggregator server consumption.
- **Import:** The import is scheduled to run at 8:00 AM daily.
- **Tables Updated:** Device, hwAssetData, IPv4, MAC, Port, SerialNumber, SubComponent, VLan, VLanObject

Reports

Enterprise Discovery provides numerous reports to help you analyze and understand what is contained in your network. Used for historical reasons, this information is then updated to the Reports database and associated with each Modeled device.

- **Extract:** The extract is scheduled to run every hour on the hour
- **Import:** The import is scheduled to run every hour on the hour.
- **Tables Updated:** EventSummary, ReportState, ReportUpdateTime

Inventory Data

This is essentially the data collected from the devices that support the Scanner executable. This data passes directly to the reports database.

- **Extract:** When the XML Enricher processes a scan file, certain scan file data is passed to the Real-time database. This allows the device models to be updated with data such as IP addresses, MAC addresses, ports. In addition, the XML Enricher creates files to be imported into the database. Once an hour or after 100 scan files have been processed, these files are transferred to the database import directory.
- **Import:** The import is scheduled to run every 30 minutes, starting on the hour.
- **Tables Updated:** Application, ApplicationCategory, Company, hw* (except hwAssetData), Language, OperatingSystemGroup, Release, SoftwareUtilization, SWSubComponent, User, Version

Data transfer from the Remote server to the Enterprise Discovery Aggregator server

The last section talked about how the data is transferred internally within a remote server. This section covers how the data is transferred from the Remote server to the Aggregate Server.

At 6:00 in the morning, the files are constructed on the remote server. If Aggregation is configured, the copy if the data produced is consumed by the Local Reports Database and it is also transferred to the Aggregate Database.

This is done so the Aggregator does not have to wait for the local import to be done before being able to perform its own import as well as for performance reasons.

The data types aggregated

Five types of data are aggregated. These are the same data categories that are imported into the Reports database on a single Remote server. Data is imported into the Reports database of the Aggregator server. Each category of data has its own schedule for being imported into the database. For each category, data is imported in two phases:

- **Transfer:** Data is copied from the Remote server onto the Aggregator server.
- **Import:** At the scheduled time, Enterprise Discovery will import the data into the database.

Events

As a device's events change during its Discovery and Inventory life cycle this information needs to be in the central database (Aggregator) for administration.

- **Transfer:** Transfer schedules can be configured in:

  Aggregate Administration > Remote server administration > Remote Server Properties

  The default transfer schedule is every hour.
- **Import:** The import is scheduled to occur every 5 minutes starting on the hour.
- **Tables Updated:** Attribute, AttributeState, ConnectionEvent, Event.

Hourly Summary

The data transferred at this point is relative to the information that is passed to the Hourly Summary table in the Reports database. This information is used to report statistical information on the devices in the network.

- **Transfer:** The transfer is scheduled to occur every hour on the hour.
- **Import:** The import is scheduled to occur every hour on the hour.
- **Tables Updated:** HourlySummary

Hardware Discovery Data

This information is used to report statistical information on the devices in the network. Hardware Discovery data can be imported on demand from:

Aggregate View > **Aggregate Status** > **Aggregate server health**.

- **Transfer:** Transfer schedules can be configured in:

  Aggregate Administration > Remote server administration > Remote Server Properties

  The default transfer schedule is Daily at 6am
- **Import:** The import is scheduled to run at 8:00 AM daily
- **Tables Updated:** Device, hwAssetData, IPv4, MAC, Port, SerialNumber, SubComponent, VLan, VLanObject

Reports

The Reports data provides a summary of the events that have occurred on the remote Enterprise Discovery servers.

- **Transfer:** The transfer is scheduled to occur every hour on the hour.
- **Import:** The import is scheduled to run every hour on the hour.

- **Tables Updated:** EventSummary, ReportState, ReportUpdateTime

Inventory Data

This is Aggregate Workstation Inventory: Device data as found in scan files provided by Enterprise Discovery. The default is to never transfer this data.

- **Transfer:** Transfer schedules can be configured in:

  Aggregate Administration > Remote server administration > Remote Server Properties
- **Import:** The import is scheduled to run every 30 minutes, starting on the hour
- **Tables Updated:** Application, ApplicationCategory, Company, hw* (except hwAssetData), Language, OperatingSystemGroup, Release, SoftwareUtilization, SWSubComponent, User, Version

Connecting the Reports database on the Aggregator server

For both the Remote server and the Aggregate server the settings used to connect to the Reports database are the same.

The following parameters are needed to connect to the Reports database on the Aggregator server.

Port: 8108

Database name: Aggregate

Username/password: As defined in Enterprise Discovery.

The Optimum time to run a Connect-It scenario

Connecting to a single Enterprise Discovery server

For an up-to-date inventory, the Connect-It scenario should be run after the Inventory import at 8am.

Connecting to an Aggregator server

For an up-to-date inventory, the Connect-It scenario should be run after the Inventory import at 8am.

RFCs supported by Enterprise Discovery

Table 3 Supported RFCs

RFC number	Name
RFC 1155	Structure and Identification of Management Information for TCP/IP-based Internets
RFC 1157	A Simple Network Management Protocol (SNMP)
RFC 1213	*see* RFC 2011, RFC 2012, RFC 2013
RFC 1285	FDDI MIB (SMT 6.2); *see also* RFC 1512
RFC 1315	*see* RFC 2115
RFC 1354	*see* RFC 2096
RFC 1398	*see* RFC 1643
RFC 1406	Definitions of Managed Objects for the DS1 and E1 Interface Types
RFC 1407	Definitions of Managed Objects for the DS3/E3 Interface Type
RFC 1493	Definitions of Managed Objects for Bridges (Bridge MIB)
RFC 1512	FDDI MIB (SMT 7.3)
RFC 1513	Token Ring Extensions to the Remote Network Monitoring MIB
RFC 1514	Host Resources MIB
RFC 1516	Definitions of Managed Objects for IEEE 802.3 Repeater Devices
RFC 1643	Definitions of Managed Objects for the Ethernet-Like Interface Types (Ethernet Interface MIB)
RFC 1695	Definitions of Managed Objects for ATM Management Version 8.0 using SMIv2 (ATM MIB)
—	ATM Forum 3.1 UNI specification
RFC 1748	IEEE 802.5 MIB using SMIv2
RFC 1759	Printer MIB
RFC 2011	SNMPv2 Management Information Base for the Internet Protocol using SMIv2
RFC 2012	SNMPv2 Management Information Base for the Transmission Control Protocol using SMIv2
RFC 2013	SNMPv2 Management Information Base for the User Datagram Protocol using SMIv2
RFC 2020	Definitions of Managed Objects for IEEE 802.12 Interfaces (100VG AnyLAN MIB)

Table 3 Supported RFCs

RFC number	Name
RFC 2096	IP Forwarding Table MIB (Router MIB)
RFC 2115	Management Information Base for Frame Relay DTEs Using SMIv2 (Frame Relay MIB)
RFC 2233	Interfaces Group MIB using SMIv2
RFC 2576	Coexistence between Version 1, Version 2, and Version 3 of the Internet-standard Network Management Framework
RFC 2668	Definitions of Managed Objects for IEEE 802.3 Medium Attachment Units (MAUs)
RFC 2674	Definitions of Managed Objects for Bridges with Traffic Classes, Multicast Filtering and Virtual LAN Extensions
RFC 2737	Entity MIB Version 2
RFC 3410	Introduction and Applicability Statements for Internet Standard Management Framework
RFC 3411	An Architecture for Describing Simple Network Management Protocol (SNMP) Management Frameworks
RFC 3412	Message Processing and Dispatching
RFC 3413	SNMP Applications
RFC 3414	User-based Security Model
RFC 3415	View-based Access Control Model

Communication models

Frame relay

Enterprise Discovery supports frame relay devices that conform to:

- RFC 2115, which supersedes RFC 1315

Each physical frame relay port may have one or more circuits associated with it. For some devices, Enterprise Discovery is able to identify the circuits related to each physical port and gather traffic statistics both for the physical port and for each circuit. Enterprise Discovery can also make connections between devices connected by these frame relay circuits.

The Device Manager Ports panel presents the ports so as to make apparent the association between a physical port and its circuits. For devices on which Enterprise Discovery is able to do a physical port mapping, each port is displayed in the form *x.y.z*, where *x* represents the slot or card number on which the port *y* is located, and *z* represents the frame relay circuit.

Using a Cisco 7200 router as an example, here's how Enterprise Discovery arranges the port structure:

...	
1.5	—
1.6	—
1.7	frame relay physical port
1.7.27	frame relay circuit
1.7.32	frame relay circuit
2.1	—
2.2	—
...	

If a device supports frame relay but Enterprise Discovery is not able to map the exact physical ports, each port is displayed in form *x.y*, where *x* represents the MIB-II object ifIndex and *y* represents to frame relay circuit. Using a Cisco 2500 router as an example:

1	—
2	—
3	—
4	frame relay physical port
4.75	frame relay circuit
4.76	frame relay circuit
4.78	frame relay circuit
5	—
6	frame real physical port
6.21	frame relay circuit
6.27	frame relay circuit

The line speed is set for each frame relay circuit. Each circuit should report a Committed Information Rate (CIR).

The CIR has meaning only for frame relay lines. It is used in service-level agreements and contracts for supply of communications bandwidth over frame relay lines. CIR has no functional impact on the performance of frame relay devices. For Enterprise Discovery to read the CIR from the device, it must have been entered into the device's MIB. If Enterprise Discovery cannot find the CIR in the MIB, it sets the frame relay circuit CIR to the line speed for that frame relay physical port.

If Enterprise Discovery has determined the CIR incorrectly, you can use the Port Manager's Port Properties button to redefine it. You may change the interface rate at either end or at both ends.

The following examples and rules describe the effect of setting the interface rate to set the CIR.

Suppose a frame relay line connects device A port 1 and device B port 2. The CIR (A1-B2) is defined from A1 to B2. The CIR (B2-A1) is defined from B2 to A1 and can have a different value from the CIR (A1-B2).

This table shows an example of the effects of setting the CIR.

A1		**B2**		**CIR A1 to B2**	**CIR B2 to A1**
line speed (kb/sec.)	**set by user**	**line speed (kb/sec.)**	**set by user**	**line speed (kb/sec.)**	**line speed (kb/sec.)**
100	no	200	no	100	100
100	no	50	no	50	50
100	no	100	no	100	100
100	yes	50	no	100	50
100	yes	200	no	100	100
100	yes	50	yes	100	50
100	yes	200	yes	100	200

The rules that constructed this table are:

- The line speed is read from the device's MIB unless overridden by the user setting it.
- If the line speed is set by the user at one end, the CIR from this end is defined as that line speed.
- If the line speed is not set by the user at an end, the lower speed at either end defines the CIR for an end.

FDDI

Enterprise Discovery has limited support for FDDI:

- support for the SMT v6.2 MIB (specified by RFC 1285)
- support for the SMT v7.3 MIB (specified by RFC 1512)

Enterprise Discovery makes FDDI connections based on the MAC address and MIB variables for each device, not based on the FDDI port.

SMT (Station ManagemenT) is an integral part of any FDDI implementation. SMT v6.2 can determine the upstream neighbor for an object. SMT v7.3 can determine both the upstream and downstream neighbors for an object.

If you have a device that supports only SMT v6.2, check with the vendor or manufacturer for SMT v7.3 support. This will improve your FDDI connectivity.

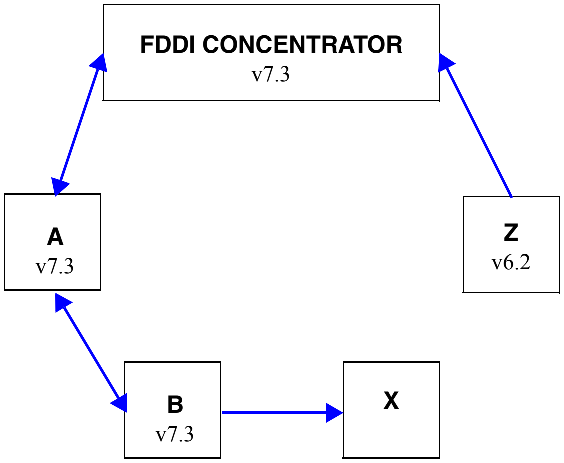
Enterprise Discovery uses the SMT instance—not the FDDI ports—when mapping FDDI objects. For example, if you have an FDDI concentrator with 8 ports, there is a single SMT instance, so Enterprise Discovery shows only one uplink port and one downlink port for that concentrator.

If Enterprise Discovery cannot always close the logical ring for your network, it is because:

- all your FDDI objects have no SNMP management
- all your FDDI objects have SNMP management but support SMT implementations other than v7.3 or v6.2
- at any point in the ring, you have an FDDI object with no SNMP management immediately downstream of an object that supports only SMT v6.2.

To understand this last case, you must realize that the object with no SNMP management (X) is providing no “ring information” about itself to the FDDI ring.

This diagram shows a FDDI ring that cannot be closed.



The only way for the ring to remain unbroken is for the next object upstream (Z) to be able to look back downstream and ask object X about itself. If Z supports only SMT v6.2, then Z cannot see downstream, and therefore the ring cannot be closed.

If Enterprise Discovery is never able to close the ring, check for objects with no SNMP management followed by an object with support for SMT v6.2 only. This is the only likely cause of a broken ring that will not be immediately obvious.

HSRP

Hot Standby Router Protocol (HSRP) is specifically for Cisco routers. There are other, similar protocols, like the Virtual Router Redundancy Protocol (VRRP) that work with other products. This section is dedicated to HSRP, but Enterprise Discovery works similarly with VRRP. For more information on how Enterprise Discovery handles these protocols, contact Customer Support.

HSRP is a routing protocol that allows multiple routers to act as a single “virtual router.” If the main router fails, there are other routers available in “hot standby” mode that will immediately take over the traffic, ensuring constant network connectivity.

➤ For more detailed information on these routing protocols, contact the product manufacturer.

The basic HSRP configuration would be to have a single virtual IP (a.b.c.1) and a set of routers that will respond to this virtual IP (a.b.c.2, a.b.c.3, a.b.c.4, etc.). Only one active router will respond to the virtual IP at any given time.

There are special virtual MAC addresses that are reserved for use with HSRP, in the form of 00000C07ACxx. The same virtual HSRP MAC address can appear in any of the routers (main or standby) that are responding to the virtual IP address.

➤ The routers will all have their own individual MAC addresses as well (appearing in the device MIB).

Enterprise Discovery may see any combination of IP and MAC addresses for the HSRP routers. It could see the real and virtual MAC for each router, depending on how the routers have been configured.

Since all the routers should respond to Enterprise Discovery pings and SNMP queries, each physical router (active and standby) should appear on the Network Map. The device model for each router should include its real MAC address.

The virtual IP address may appear on the Network Map if the virtual IP has been seen in an ARP cache and if the routers are SNMP-managed. If the virtual IP does not appear on the Network Map, there will be an unmanaged IP+MAC device, likely connected to the active physical router by a diamond-shaped IP connector device icon.

If the active router is SNMP-managed, the virtual IP will not appear on the Network Map.

➤ If Enterprise Discovery finds the virtual IP in an ARP cache before it finds the active router, the virtual IP will appear on the Network Map until it is merged with the active router. Once the active router appear on the Network Map, the virtual IP will no longer appear on the Network Map.

Scheduled Events

The majority of data that Enterprise Discovery uses is constantly being collected. However, some information is collected at a set time every day, while other information is summarized once a day.

This is a list of major events, not a complete list.

Table 4

Time	System event
0005–1900[a]	summarize statistics for each attribute update Prime configuration summarize events for reports compile and calculate reports
0005–2330[b]	check devices for deactivating and purging update Health Panel reports (Exceptions, Last Seen, Adds, Deletes, Moves, Changes)

[a] If this series of events is not successfully completed, it will restart in 30 minutes and attempt to complete only the unsuccessful events from the series.

[b] This series only begins once the previous series has finished.

Software Library

The Software Library used in the Application Recognition process consists of a set of Software Application (SAI) files that contain all of the information necessary to deduce the existence of applications from files, registry keys, etc. on a machine. The library also contains license relationship information that allows Enterprise Discovery to automatically discover actual license requirements even for complex suite-based applications like Microsoft Office, Oracle database servers, etc.

The software library contains information about applications from more than 1000 publishers and covers Windows applications in English as well as some applications in French and German. For UNIX, libraries for HP-UX, AIX and Solaris are included with Enterprise Discovery.

In addition to the standard libraries, Enterprise Discovery includes several tools that allow you to create your own library extensions in the form of one or more User SAI files that can easily be applied to the automatic Application Recognition process.

Knowledge Updates

The Rulebase, Software Library and various other data items are updated by Hewlett-Packard on an ongoing basis and updates are made available regularly as Enterprise Discovery Knowledge updates.

A knowledge update consists of a securely signed archive containing all of the latest data files necessary for your Enterprise Discovery server. Once downloaded and copied to the appropriate directory on the server, the knowledge update is automatically verified as authentic and applied to the system. Refer to the “Installing Knowledge Updates” chapter in the *Installation and Upgrade Guide* for further information.

3 Terms and Concepts

This chapter is divided into two basic categories:

- Network Terms and Concepts on page 44 (to review terms and concepts common to network management)
- Enterprise Discovery Terms and Concepts on page 50 (to learn terms and concepts unique to this product)

Network Terms and Concepts

These terms and concepts are common to networks and network management. They are not unique to Enterprise Discovery.

SNMP

Defined by the Internet Engineering Task Force (IETF) in RFC 1157, Simple Network Management Protocol (SNMP) is the protocol that governs network management, and network device monitoring.

MIB

Management Information Base. This database of network management information is used by SNMP. The information contained in this database helps define each device by giving specific information about the device and manufacturer.

Domain names

Example: website.example.com

A domain name such as “website.example.com” is easier to remember than an IP address such as “192.168.96.1”. This ease of remembering is the chief reason for the existence of domain names.

The term “domain name” and “host name” are sometimes used interchangeably. A domain name is a name in the Domain Name System (DNS) format as registered with a DNS server. A host name is purely an internal name, used by a device to refer to itself.

Address types

The two main types of numeric address are the IP address and the MAC address.

IP address

An IP address was intended to be a unique number identifying a unique device or port of a device.

When you see the term “IP address” with no qualifiers in Enterprise Discovery, it means that either an IPv4 address or an IPv6 address is acceptable. The 32-bit address space of IPv4 addresses puts severe limits on the number of unique addresses available, and the supply is fast running out. The IPv6 128-bit address space was created to address this problem.

IPv4 address

An IPv4 address contains four sections separated by periods (or “dots”). Each section, called an octet, contains 8 bits expressed in decimal (0–255).

Example: 192.168.96.1

IPv6 address

An IPv6 address contains eight sections separated by colons. Each section contains 16 bits expressed in hexadecimal (0000–FFFF).

Example: 1234:5678:9ABC:DEF0:1234:5678:9ABC:DEF0

To make it easier to remember and type an IPv6 address, you can use a double colon (::) to indicate multiple contiguous sections of zeros. You can also omit leading zeroes. For example, you can simplify address 0123:0000:0000:0000:0004:0056:789A:BCDE to 123::4:56:789A:BCDE.

MAC address

A MAC (Media Access Control) address is a unique number identifying a unique device or port of a device.

When you see the term “MAC address”, it means a numeric MAC address.

Numeric MAC address

A MAC address contains six sections. Each section contains 8 bits expressed as a hexadecimal number (00–FF).

Sometimes the first three sections and last three sections are separated by one space; sometimes all sections are presented as one, without spaces; sometimes each section is separated by a colon or a space.

Examples: 010203 FDFEFF, 010203FDFEFF, 01:02:03:FD:FE:FF

MAC address including OUI

This type of MAC address is sometimes (inaccurately) referred to simply as an OUI. In fact, the Organization Unique Identifier (OUI) comprises the first three sections of a MAC address. If Enterprise Discovery recognizes the numeric form of the OUI, it replaces the numbers with a short form of the organization name. This makes it easier to identify a device. If Enterprise Discovery uses an alphabetic short form for a device’s OUI, the device is said to have a recognized OUI. Having a recognized OUI is sometimes abbreviated to “having” an OUI.

Example: DELL 59FC91

Netmask notation

Network masks, often referred to as netmasks, can usually be expressed in two formats in IPv4—either the familiar octet notation (also called dotted decimal notation) or CIDR notation.

Example of octet notation: 255.255.255.248

Example of CIDR notation: 29

The shorter CIDR notation is based on the binary equivalent of the octet notation, and refers to the numbers of contiguous 1s. Below are examples of netmask notation:

255.255.255.255	11111111.11111111.11111111.11111111	32 1s
255.255.255.248	11111111.11111111.11111111.11111000	29 1s
255.255.0.0	11111111.11111111.00000000.00000000	16 1s

➤ A valid netmask contains a series of contiguous 1s. Zeros can appear after a sequence of 1s but cannot appear before or between the 1s. If zeroes appear before or between the 1s, it is not a valid netmask.

In IPv6, netmasks can only be written in CIDR notation.

Community strings/Users

Depending on the version of SNMP supported on a device, a management system can access the SNMP MIB with a community string (SNMPv1/v2) or a user (SNMPv3).

Community strings and users are like device-based password that control access to the SNMP MIB of a device. A device controls its own community strings/users, but you must tell Enterprise Discovery about them.

If Enterprise Discovery is not given the correct community strings/users and access to devices on your network, Enterprise Discovery will be unable to read device MIBs. Enterprise Discovery will then assume that each device it cannot read has no SNMP management available.

Multiple Strings

For each device that it discovers, Enterprise Discovery will try all the community strings/users you have provided for that device and use the first string that receives a positive acknowledgement to read or write to the system MIB. This means that Enterprise Discovery may try several community strings/users before it finds one that will cause the device to respond.

The fact that Enterprise Discovery may try several community strings/users has implications for any devices that issue SNMP traps (also known as security traps and authentication traps).

SNMP Traps

Some devices may issue an SNMP trap when Enterprise Discovery attempts to explore them. Even if Enterprise Discovery has the correct community string/user in its list, Enterprise Discovery may still “trip” the trap if Enterprise Discovery tries multiple community strings/users before finding the right one.

For example, Enterprise Discovery might try two invalid community strings before reaching the valid community string. Any invalid community string will “trip” a security trap.

Once a trap has been tripped, the trap may be re-issued periodically until the trap is reset. Enterprise Discovery does not reset traps. Therefore, you should either disable all such traps or use only a single correct community string/user for each device that issues a trap.

➤ If another network management system is used in the same network with Enterprise Discovery, this other system may generate alarms due to these traps.

Directed Community Strings

If a device is programmed with a directed community string (sometimes known as a direct access list), it will reject the attempt by Enterprise Discovery to SNMP QUERY it, even if Enterprise Discovery has been given the correct community string. With a directed community string, each device checks not only the “password,” but also to see if the Enterprise Discovery server is on the list of “trusted” devices.

You can allow Enterprise Discovery to communicate with a device with a directed community string, but you cannot do so merely by configuring Enterprise Discovery. You must also give the device itself an entry for a directed community string associated with the IP address of the Enterprise Discovery server.

Bridge aging

To obtain the best results with Enterprise Discovery, turn bridge aging on. Also, set the aging interval for 2–6 hours, although some circumstances may call for an aging interval as long as 12 or even 24 hours. (Longer aging intervals are not always possible. A common maximum aging interval is 32767 seconds, or just over 9 hours.)

Bridges, routers, and switches generally have tables in which they store the addresses of devices on the network. The tables are periodically purged and relearned in order to keep the list of devices current. The aging interval defines the frequency with which tables are purged and relearned.

When there is no table entry for the address of an incoming packet, the bridge, router, or switch must learn the location of the address. To learn the location, the device sends the incoming packet to all its own ports. (This is often referred to as “flooding” or “leakage”.) When the destination device with the corresponding address responds, the bridge, router, or switch learns the location and makes an entry in the address table.

If the table is full and a new entry must be made, the “oldest” entry is usually replaced by the new entry. Device manufacturers commonly strive to include a table large enough to hold the addresses of all active sessions, but space in a table is always finite.

Enterprise Discovery reads the tables of bridges, routers, and switches to learn the addresses of all the connected devices. Many bridge, router, and switch vendors use a standard aging interval of 300 seconds (5 minutes), which is too short.

If the bridge aging interval is too short:

- Enterprise Discovery may never discover devices that are connected to the network for short periods—for example, laptops.
- Enterprise Discovery may take longer to determine connections between devices that it has discovered.
- Tables will be purged so frequently that flooding will occur regularly, using bandwidth unnecessarily.

If bridge aging is not turned on for a device, or if the bridge aging interval is too long:

- Tables will contain old addresses of devices that may been removed from the network or devices that are broken. As a result, Enterprise Discovery will work from an outdated and possibly confused representation of what is in your network and how it is connected.

OSI model layers

The Open Systems Interconnection (OSI) model has seven layers. Layers 2 and 3 are the most important to Enterprise Discovery:

- Layer 2 is the Data Link layer, at which level MAC addresses are used. Bridges and some switches are layer 2 devices.
- Layer 3 is the Network layer, at which level IP addresses are used. Routers are layer 3 devices.

Some switches are both layer 2 and layer 3.

The seven layers are:

Layer number	Layer
7	Application
6	Presentation
5	Session
4	Transport
3	Network
2	Data Link
1	Physical

Management workstation

Any workstation or personal computer capable of running a supported web browser. There is more detail on requirements for a management workstation in the *Installation and Initial Setup Guide*.

Virtual machine

It is a virtualized environment that emulates in software the hardware of a real machine, including a set of emulated physical devices. Operating systems and other programs can be installed on a virtual machine and will behave as if installed on a real physical computer. The virtual machine utilizes the physical resources, such as RAM and disk space, of the physical host machine it is installed on.

Solaris zone

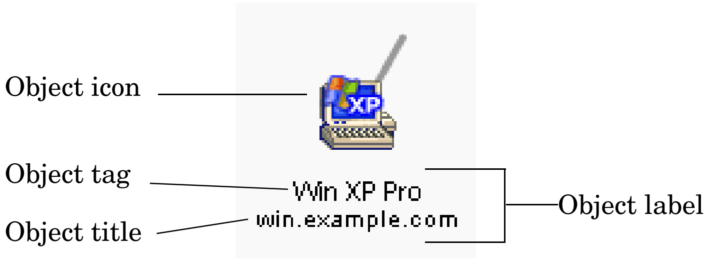
It is a software partitioning technology, which provides a means of virtualizing Solaris operating system services to create isolated environments (zones) for running applications. This isolation prevents processes that are running in one zone from monitoring or affecting processes running in other zones. Different amounts of physical resources (such as RAM, CPU utilization, etc.) can be allocated to individual zones.

Enterprise Discovery Terms and Concepts

These terms and concepts are either unique to Enterprise Discovery, or have a special meaning in this context.

The object label

For devices, the object label tells you what kind of device it is. For packages, the object label tells you how many devices are in the package.



Real device

- device tag classifies the device
- device title identifies a specific device

Tag type	Example
Rule-specific[a]	Cisco NCD?
Model	Cisco 1601
Family	Cisco 1600
Network Function	Optivity
Operating System	Windows 95
Registered SysObjId Manufacturer	Novell Inc
Registered OUI(MAC) Manufacturer	Cisco

[a] Limited information is available, or, a managed device is not listed in the Enterprise Discovery Rulebase; see also the following table.

Ending	Meaning
?	less than 90% probability of identity
NCD?	Enterprise Discovery is relying on the MAC address. The OUI indicates that the device is probably a network connectivity device (NCD), but there is some possibility that it may be an end node.

Connector device

- no device tag
- device title can identify a subnet or can be arbitrary

Package

- package tag shows number of devices contained by package
- package title can identify parent device (automatic package) or top object of package (multi-object package); can also be arbitrary (any package)

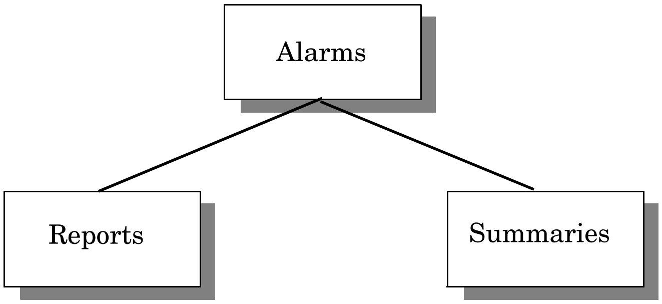
Events and Alarms

Events are caused by changes on the network, such as adding a new device, or changing a device property such as its icon or title. All of these events are reflected in the Events Browser.

There are 4 basic event types:

Event Type	What is Generated	Example
Add	Event in the Events Browser	new device added to network
Delete	Event in the Events Browser	the device is hidden or has been deactivated
Move	Event in the Events Browser	connectivity change, physically moving a device
Property Change	Event in the Events Browser	changing a device icon, priority

An event triggers an alarm. For detailed information about the specific alarms that Enterprise Discovery recognizes, see **Help > Classifications > Alarms**. The following diagram shows the alarm hierarchy:



Reports (for example, MTTR and MTBF) are accumulated data; they summarize data for the past 24 hours. You can change the default “time period” for these alarms in **Administration** > **System Configuration** > **MTTR and MTBF**.

Summaries (for example, Adds and Deletes) pertain to specific events. You can change the default “time period” for these alarms in **Administration** > **System Configuration** > **Adds/Deletes/Changes/Moves**.

Here is a list of the alarm indicators that are visible in the Health Panel and elsewhere in the user interface:

Alarm Type	**Indicator**	
n/a (not an alarm state, this indicates that the attribute is not being monitored)		blank
OK	–	dash
Info	✱	green asterisk
Minor Alarm	▲	gold triangle
Major Alarm	◆	orange diamond
Critical Alarm	🟥	red square

Panel Elements

Certain elements are common to all Device Manager or Port Manager, Line Manager, or Attribute Manager panels:

- When data in a table has a gray background, the data shown is considered stale, because it was obtained before the beginning of your selected time period. (In some cases, data may be shown in parentheses rather than with a gray background.) To change the time before data is considered stale, see the section on Account Properties in the *Configuration and Customization Guide*.
- A blank space indicates that data is not available for a device or port.
- The final line on each panel is the date and time that the panel was refreshed. (This refers to rendering the panel itself, not when the data shown in the panel was last read.) This date can be useful when you print a panel. To change the format of this date, see the section on Account Properties in the *Configuration and Customization Guide*.

Banner

The banner that appears at the top of all Device Manager or Port Manager , Line Manager, or Attribute Manager panels consists of several elements.

Element	Example	Notes
Attribute Name	Total Breaks	This only appears in the Attribute Manager
Device title and IP address	website.example.com / 192.168.96.1	see Device Title on page 53 if the device title is the IP address, the IP address is shown once if there is no IP address, only the device title is shown
Manager name	Device Manager	—
System name of Enterprise Discovery server	ExampleCorp	see the *Installation and Initial Setup Guide*
Web browser name	Netscape | Internet Explorer	—

Device Title

The device title displayed in the banner of Device Manager or Port Manager, Line Manager, or Attribute Manager (and in some panels of those managers) will be the first available of a device title chosen by the Enterprise Discovery Administrator in **Administration > System Configuration > Display preferences**. The Enterprise Discovery Administrator can choose one or several of the following and specify their order of preference:

- Asset tag
- BIOS Asset tag
- NetBIOS name (scan)
- Last name
- First name
- Device-specific title
- Domain name

- Host name
- VM name
- NetBIOS name (network)
- Operating system
- Family
- Model
- Network function
- System description
- System name
- System location
- System contact
- IPv6 address
- IPv4 address
- MAC address including OUI
- MAC address (all-numeric)

Only Administrator or IT Manager accounts can change device titles (**Device Properties** button on the Device Manager). The device titles are global. To determine the default title for a device, see the Diagnosis panel on the Device Manager.

4 Virtualization in Enterprise Discovery

Overview

Virtualization allows administrators to set up several virtual devices on one physical device. This is a popular technology, since there is often a need to consolidate physical machines as virtual machines on one physical server. This provides flexibility to relocate and add virtual machines as needed.

Enterprise Discovery provides you with the ability to discover, scan, and collect utilization data on virtual devices just as if they were physical devices. Some Virtualization technologies offer advanced support. When this support is available, Enterprise Discovery can not only provide the basic discovery, inventory, and software utilization data but can also determine the parent/child relationship between the physical host device and the virtual devices hosted on the physical device. As such, you are able to see the host/virtual device information in several places in Enterprise Discovery including the Network Map, the Virtual Devices Window, the Device Manager, and the Virtual Device Reports.

Supported Virtualization Technologies

Enterprise Discovery supports two virtualization technologies that provide advanced support: VMware and Solaris zones.

VMware on the ESX server 3.0 and higher supports the Web Services interface that allows Enterprise Discovery to determine parent/child relationships between the physical host machine and the virtual machines hosted on the physical machine.

Solaris zones technology on Solaris version 10.0 also provides the support that allows Enterprise Discovery to determine parent/child relationships between the physical host and the virtual zones hosted on the physical host.

This chapter emphasizes VMware on ESX server 3.0 and Solaris zones on Solaris 10 because of the advanced support that these technologies provide. However, Enterprise Discovery also supports several virtual environments that provide just basic support, that is, they do not provide the information required by Enterprise Discovery to determine parent/child relationships. These virtual environments include VMware workstation, VMware GSX server, VMware pre-3.0 ESX server, Virtual PC, and Virtual server. All of these environments can be detected by the scanner, and the scanner can exit if no inventory is required for these virtual environments. If inventory is required, these devices are treated like normal physical devices. As such, scanners in these virtual environments must be executed and configured to not exit in that specific virtual environment. See Scanner Options for Virtualization on page 59.

So while this chapter specifically cites the technologies that provide advanced support, the information that does not depend on knowing the parent/child relationship also applies to the technologies that provide only the basic support.

VMware

This technology provides the ability to host several virtual machine (VM) images on one physical device. The VM image represents a virtual machine with virtual hardware and with an OS (which can be different from the host OS) and applications running on it that are distinct from the host and other VMs. In this implementation, the OS is not aware that it is running on a virtual machine.

Enterprise Discovery enables you to see how VMs are being used in your network environment. VMs are linked to their physical host machines, but each VM is treated as a separate device with respect to discovery, hardware scanning, and software utilization. For example, each VM appears as a distinct device on the Network Map. Enterprise Discovery can be configured so that an agent is deployed to the host machine and to each VM.

You can collect and view discovery, inventory, and software utilization data for individual VMs, and you can view a list of all the VMs hosted by a given server. You can configure the frequency with which Enterprise Discovery queries the host server to track the dynamics of the virtualization environment. You can also set up virtualization profiles that are associated with specific device groups. Refer to the "Virtualization Profiles" section in the "Configuring the Discovery Process" chapter in the *Installation and Initial Setup Guide.*

Solaris Zones

This technology provides the ability to partition a Solaris host system into zones (containers), each running a Solaris OS with applications. The host system is referred to as the global zone, and the hosted zones are referred to as local (or non-global) zones. The local zones are not aware of each other or the host system. This is a very light weight virtualization, implemented at a very high level.

Enterprise Discovery enables you to see how the Solaris server is being used. The Solaris zones are linked to the host server. Each zone is treated as a separate device. The host global zone physical device and the Solaris local zones appear as distinct devices on the Network Map.

As with VMware, you can collect and view discovery, inventory, and utilization data on Solaris zones and these zones are linked to the Solaris host.

Support for Virtual Devices in Enterprise Discovery

Discovering Virtual Devices on the Network

Enterprise Discovery can discover virtual devices in the same manner that it discovers physical devices on the network. It can discover virtual devices on its own by pinging and polling through a list of IP ranges that you provide. As it does with physical devices, after it discovers a virtual device, Enterprise Discovery creates a device model (a unique description of that device) and adds the created device model to the Enterprise Discovery database. You can prevent Enterprise Discovery from adding virtual devices to the database if you like. See Virtual Devices in the Enterprise Discovery Database on page 61.

Scanning Virtual Devices

Scanners can be distributed to individual virtual devices from the server using the Agent.

VMware

In the case of VMware, if you want to collect inventory data on the physical host and all the VMs hosted on the physical machine, you must execute the scanner to the hosting physical machine, as well as inside each VM hosted on the physical machine. You can scan the host physical machine and each VM for its hardware components and collect a list of the software applications installed.

Scans can be scheduled to occur automatically, or they can be performed manually in the same manner as physical devices. For automatic scans, the server maintains a schedule dictating which devices should be scanned and when.

Solaris Zones

In the case of Solaris zones, there are two possible ways to scan for data in the zones. They are the following:

- Global zone scan: This is the *recommended* mode. In this scenario, the scanner on the global zone collects all hardware and software inventory for the server machine and produces a single scan file. Since the global zone has knowledge of the local zones, this global scan file can then be used to generate individual scan files for each of the local Solaris zones. See Processing Inventory for Virtual Devices on page 59. When using this method, you deploy an agent to the global Solaris zone only. This ensures that when a scanner is running in Enterprise mode that only the global zone is scanned. When running a scanner in Manual Deployment mode, you deploy the scanner to the global Solaris zone only to ensure that just the global zone is scanned. The scan file produced using this method contains hardware inventory information mainly for the global zone. The generated local zone scan files contain software inventory for the local zone and minimal hardware inventory information. This is the one disadvantage of this method. It is the preferred method because it avoids creating an additional workload on the Solaris server as described in the next bulleted item.
- Local zone scan: In this scenario, an agent is installed on each local zone. A scanner runs on each zone and produces a scan file for that local zone. Although these scan files contain more detailed hardware inventory information then the recommended global zone scan mode described in the previous bulleted item, the local zone scan mode is *not*

recommended. It produces an extra workload on the Solaris server because the same directories are scanned several times for each zone. Multiple zones are scanned separately and perhaps simultaneously depending on scheduling.

Scanner Options for Virtualization

In the Scanner Generator, you can indicate if you want containers included in a hardware detection scan. The **Containers** check box is an option under the **Operating System** category on the **Hardware Data** screen in the Scanner Generator GUI. Currently this option is relevant only to the Solaris operating system.

The **Containers** check box must be enabled for the global zone to recognize its parent/child relationship with the local zones that it hosts.

Also, on the **Scanner Options** screen on the **Miscellaneous** tab in the Scanner Generator GUI, you can specify that you want the scanner terminated if it is running in a virtual environment.

Scanners can detect if they are in a virtual environment. You can configure the scanner to exit if it encounters a virtual environment. By default, all virtual devices are turned on for scanning except for the non global Solaris zone. You want to enable the termination option for this virtual environment if you plan to run the scanner in the recommended global zone mode because you do not want the scanner to run on the local zones.

Refer to the “Scanner Generator” chapter in the *Configuration and Customization Guide*.

Processing Inventory for Virtual Devices

For VMware devices, individual scan files are created for the physical host and the VMs hosted on the physical machine if you have configured Enterprise Discovery to collect this data. Inventory processing occurs in the same manner that it does for any physical machine on the network.

For Solaris zones, there is an option that allows you to specify to the XML Enricher how you want it to process the global zone scan file (See To enable the generation of local scan files from the global scan file). If this option is enabled, the XML Enricher generates individual scan files for each local zone from the global zone scan file on the Solaris server.

Enable this option if you are running the scanner in the recommended global zone scan mode. By default this option is enabled.

Disable this option if you are running the scanner in the non-recommended local zone mode. Enabling this option causes the XML Enricher to discard the local scan files produced for each local zone by the individual scanners and to replace them with scan files that are generated from the global scan file.

To enable the generation of local scan files from the global scan file

1. Select **Administration** > **System Configuration** > **Scan Processing**.
2. For the **Generate Solaris local zone inventory from the global zone** option, select **Custom**, and then select **Yes**.

Refer to the “Configuring the XML Enricher using the WEB UI” section in the “XML Enricher” chapter in the *Configuration and Customization Guide*.

Collecting Utilization Data for Virtual Devices

In addition, you can also enable agent plug-ins to collect software utilization information on virtual devices.

For VMware, the utilization agent plug-in must be installed on the physical host machine and on each VM running on that machine.

For Solaris zones, the utilization agent plug-in must be installed on the host global zone only.

Summary of Scanning Procedures

For VMware devices:

- Install the scanner agent on the host machine and on each hosted VM.
- Disable scanner termination for VMware (default setting) on the **Miscellaneous** tab of the **Scanner Options** screen in the Scanner Generator GUI so that the scanner will not terminate when it detects this environment.

For Solaris zones using global scan mode:

- Install the scanner agent on the global Solaris zone only.
- Enable **Containers** to be detected on the **Hardware Data** screen in the Scanner Generator so that the global zone can determine its parent/child relationship with the local zones.
- Enable scanner termination for non global zones on the **Miscellaneous** tab of the **Scanner Options** screen in the Scanner Generator GUI so that the scanner will terminate when it detects this environment.
- Enable generation of scan files for local zones from the global zone scan file in the Web UI.

For Solaris zones using local scan mode:

- Install the scanner agent on all the local zones.
- Disable **Containers** to be detected on the **Hardware Data** screen in the Scanner Generator since parent/child relationship information is not needed.
- Disable scanner termination for non global zones on the **Miscellaneous** tab of the **Scanner Options** screen in the Scanner Generator GUI so that the scanner will not terminate when it detects this environment.
- Disable generation of scan files for local zones from the global zone scan file in the Web UI so that the local scan files produced by the scans on local zones are not discarded by the XML Enricher.

For Microsoft Virtual PC/Virtual Server:

- Install the agent on the host machine and on each hosted virtual device.
- Disable scanner termination for Virtual PC (default setting) on the **Miscellaneous** tab of the **Scanner Options** screen in the Scanner Generator GUI so that the scanner will not terminate when it detects this environment.

For Windows Terminal Services:

- Install the agent on the host machine only.

- Disable scanner termination for Terminal Services (default setting) on the **Miscellaneous** tab of the **Scanner Options** screen in the Scanner Generator GUI so that the scanner will not terminate when it detects this environment. If you use Manual Deployment mode to run scanners (for example, via login scripts), you must disable this option in order to prevent the scans from occurring in multiple Terminal Services sessions.

Virtual Devices in the Enterprise Discovery Database

You can specify whether you want Enterprise Discovery to add virtual-only devices, such as Solaris zones and VMware VMs, to the database.

To prevent virtual devices from being added to the database

1. Select **Serve**r > **Administration** > **System Configuration** > **Input Filters**.
2. Under **Devices not to add to the database**, select **Custom** and then check the **Virtual-only devices** option.

Enterprise Discovery treats a virtual device (zone or VM) that has an SNMP agent installed as a managed device. It treats a virtual device with a manual scanner installed, but no SNMP agent, as a scanned device.

Virtual Devices on the Network Map

By default, individual virtual devices such as Solaris zones or VMware virtual machines are displayed on the Network Map. You can choose not to have these devices appear on the Network Map if you prefer.

To prevent virtual devices from appearing on the Network Map

1. Select **Server > Administration > System Configuration > Network Devices**.
2. For the **Show virtual devices on the Network Map** option, select **No**.

Regardless of this setting, you can use the Virtual Devices Window to display a list of the virtual devices associated with a particular host. See the next section.

Virtual Devices Window

Once you locate a network device through the **Find** command, the Network Map, or any other method, there is an option on the **Device** menu in the various applet windows that allows you to see if the device is a virtual device or a machine hosting virtual devices. If the device is a virtual host or virtual device, the **Show Virtual Devices** option will be enabled on the **Device** pull-down menu. Selecting this option opens the **Virtual Devices** Window. This window displays the host machine name, its IP address, and the number of virtual devices that it is hosting. Below this, the following information is displayed for each virtual device associated with the host device:

- Device: Logical machine name of the virtual device
- IP Address: IP address of the virtual device
- VM Type: Virtual device type (VMware or Solaris zone)
- VM Name: Name assigned to the virtual device
- VM Operating System: Operating system running on the virtual device

- VM Status: Current status of the virtual device
- Update Time: Time that Enterprise Discovery last collected information about this device

If you double-click on any of the devices in this window, the device opens in the Device Manager. You can also access the Device Manager and the Network Map by the icons provided in the upper right-hand corner of the window.

Virtual Devices and the Device Manager

The Device Manager displays specific items when it knows that a device is either a virtual device or a machine that hosts virtual devices.

For VMware hosts, the Configuration panel includes the VMware Credentials table. This table displays the preferred VMware credentials, including the user name and a password hint, for this VMware host. Preferred credentials are the credentials that were used most recently to successfully retrieve information from the VMware host machine.

The VMware Credentials table is present only for VMware host devices. The information in the table is populated after the VMware discovery process is completed for this host.

The Diagnosis panel contains three items that pertain to virtualization.

- The first item is the Configuration Profiles table, which shows the name of the virtualization profile that applies to this device.
- The second item is the Discovery Configuration table, which lists three virtualization parameters for this device: VMware discovery credentials, VMware discovery schedule, and VMware discovery interval. These parameters are specified in the virtualization profile that is associated with this device.
- The third item is the Virtualization Log button on the Diagnosis panel toolbar that displays information logged during the VMware discovery process running on the host.

The Update Model panel contains the **Run VMware Discovery** option in its pull-down menu that allows you to force the discovery of a device that is a VMware host (ESX server 3.0 or later) so that its network model and the model of its hosted VMs are updated immediately in the Enterprise Discovery database. The **Run VMware Discovery** option is also available from any applet window from the right-click menu associated with a device.

For additional information, refer to the “Using the Device Manager” chapter in the *Network Data Analysis Guide*.

Virtualization Configuration Profiles

A virtualization profile enables you to specify two things: (1) VMware credentials, and (2) how often and when the discovery process for virtual devices is initiated. A virtualization profile can be associated with one or more device groups. A device group can be organized either by IP range or device type.

For additional information about discovery configuration profiles, refer to “Configuring the Discovery Process” in the *Installation and Initial Setup Guide*.

Reports on Virtual Devices

You can see a summary and detailed reports of all hardware inventory data collected on virtual devices. As indicated, currently Enterprise Discovery supports two virtualization technologies: Solaris zones and VMware. A detailed report is provided for each of these virtualization types.

To view the virtual devices summary report

1. Select **Server** > **Reports** > **Network Documentation**.
2. Under **Summary Reports**, select the **Virtual Devices by Virtualization Type**. You can click the links on the summary report to see the detailed reports by virtualization type.

Refer to the “Using the Reports” chapter in the *Network Data Analysis Guide*.

Deactivation and Purging of Virtual Devices

You can indicate when you no longer want virtual-only devices included in your reports or in the Enterprise Discovery database.

You can specify a deactivation interval for a virtual-only device. At the end of this deactivation interval, the virtual device is removed from the Reports, but its data is kept in the Enterprise Discovery database. If Enterprise Discovery sees that device again, or if you manually “reactivate” the device, it will reappear on the Reports.

You can also specify a purge interval for a virtual-only device. At the end of this purge interval, the virtual device is removed from the Enterprise Discovery database. The purge interval starts at the end of the deactivation interval. The virtual device’s deactivation interval is dependent on the virtualization discovery interval for the device. The time it takes to deactivate a virtual device is either the virtual devices deactivation interval, or three times the virtualization discovery interval, whichever is longer. For example, if you change the virtual devices deactivation interval to one day, but do not change the virtualization discovery interval of 2 days, Enterprise Discovery will take 6 days (3 x 2 days) to deactivate an unseen virtual device.

At the end of the virtual devices purge interval, the “deactivated” virtual device and all its data are deleted from the Enterprise Discovery database. A purged device may be rediscovered if it is still connected, but it will be considered a new device. A virtual device may be purged before the end of the specified virtual devices purge interval. The storage area for deactivated devices has limited capacity (10% of the device license). Once the number of deactivated devices reaches capacity, some devices will be automatically purged.

You can specify deactivation and purge intervals for a device on the **Server** > **Admin** > **System Configuration** > **Expiry** page.

5 Recorded Events

The Health Panel summarizes changes to the network. Each device should contribute only a single alarm to the Health Panel. If there is more than one alarm per device or per port, these additional alarms are displayed in the Device Manager or Port Manager.

To see a comprehensive list of alarms that are raised by report data, see **Help > Classifications > Alarms**.

Port Add/Deletes

Identifies ports recently added to or deleted from a device. (An added port may or may not be recently discovered.)

➤ Does not include ports on connector devices.

Port Changes

Changing interface rate, interface type, duplex, or line alarm type.

Device Adds/Deletes

Identifies devices recently added to or deleted from the database. (An added device may or may not be recently discovered.)

Device Changes

Changing icon, priority, title, or tag of a device.

Exceptions

Devices with exceptions. See **Help > Classifications > Exceptions**.

Not Recently Seen

There are two types of “not recently seen” events:

- Network Not Recently Seen
- Scan Not Recently Seen

“Network Not Recently Seen” devices are those with which Enterprise Discovery has lost contact and which may soon disappear from the database.

➤ Once Enterprise Discovery has not had contact with a device for a period greater than the threshold (by default, 6 hours), it will appear as “Not Recently Seen.” it will be displayed with a green ring. Once the “not seen” period has exceeded 24 hours, the device will also be appear faded.

“Scan Not Recently Seen” devices are devices for which Enterprise Discovery has not received an updated scan file (by default, 4 weeks and 2 days).

➤ You can change these defaults at **Administration > System Configuration > Report time periods**.

➤ This does not apply to connector devices

6 Scanners

The scanner used to scan each computer can capture any or all of the following types of information, depending on the options selected when the scanner was configured:

- Information about the hardware configuration.
- Information about the system configuration.
- Information about the software on the drives scanned.
- Information about the physical assets and user details that are recorded using the asset questionnaire.

The information collected by a scanner is stored in a Compressed XML File (XSF) file. This information can be viewed immediately with the Analysis Workbench or Viewer, but it can also be enriched using the XML Enricher, and have its data sent to the Enterprise Discovery server database. From there, the data can be viewed through the Scan Data Viewer, Reports, and so on.

The Scanner Types

Scanners can be generated for the several operating systems. See **Help** > **Compatibility Matrix** for a complete list.

The procedure for starting a scanner depends on the native operating system environment for the computer being scanned.

Viewing the Results of the Scan

HP Enterprise Discovery comes with the Viewer program, which allows you to look at the results of your scans. Refer to the *Viewer* Chapter for more information about how to use this application.

For XSF scan files, a tool such as gzip or Winzip can be used to extract the XML data contained in them. The XML file contained inside the XSF file can be viewed with any text editor or XML viewer such as Internet Explorer.

Command Line Parameters and Switches

Although the options for the scanner are normally set using the Scanner Generator, it may be necessary to change some settings to allow better operation on some machines. The operation of a scanner can be modified with the use of the various command line parameters.

Reasons for Overriding the Options in a Configured Scanner

- The scanner may encounter a problem with a particular hardware. Using command line options, the problem hardware can be circumvented.
- Command line parameters can change the configured options such as save path. This allows the scan results to be saved to a local machine without a full network path having to be defined.

How to Use a Command Line parameter

You can specify command line parameters and switches by:

- Typing the command from a command line (for example, the Windows command prompt, or the UNIX/Mac OS X shell). In UNIX/Mac OS X make sure you specify the path to the scanner.

  For example:

  ```
  /tmp/scanlnx -?
  ```

  launches the Linux scanner from the `/tmp directory` and displays a list of valid command line options.
- Creating a Windows shortcut. Type the command line options (if any) after the quotation marks.

  For example:

  ```
  "C:\Program Files\HP OpenView\Enterprise Discovery\2.20\Scanner
  Generator\ScanW32.exe" -?
  ```

  launches the Win32 scanner and displays a list of valid command line options.
- Typing the command in the Windows Run command in the Start menu. Type in or navigate to the location where the scanner executable is located. Type the command line parameter or switch after the quotation marks.

  For example:

  ```
  "C:\Program Files\HP OpenView\Enterprise Discovery\2.20\Scanner
  Generator\ScanW32.exe" -?
  ```

Command Line Parameters for Scanners

Valid command line parameters for the scanners are shown in the following table:

Table 5 Command line parameters for scanners

Command Line Parameter	Function
-force	Do not check disk space saving offsite Scan File. This may be useful in situations where the operating system reports insufficient space, but this is actually due to access rights.

-p:<path>	Override default offsite save path.
	A UNC path can also be entered as the argument to this option. The format for a UNC path is:
	`\\servername\sharename\path\`
	For example:
	`ScanW32 -p:\\HPOpenView\ED\scanfiles\`
	The user running the scanner must have Write permissions to the specified UNC path.

-r:<path>	Override the default path to the original scan files.
	A UNC path can also be entered as the argument to this option. The format for a UNC path is:
	`\\servername\sharename\path\`
	For example:
	`ScanW32 -r:\\HPOpenView\ED\scanfiles\`
	The user running the scanner must have read permissions to the specified UNC path.

Table 5 Command line parameters for scanners

Command Line Parameter	Function

-scandays:<Count>	Scan only if previous scan was more than Count days ago.
	Forces the scanner to perform the scan only if the previous scan was <N> or more days ago. For example:
	`-scandays:7`
	For example, if the scanner is launched from a login script every day, it will only perform the scan every week.
	When the scandays:<N> parameter is specified, the scanner attempts to check when the last scan was run. If no previous scan file is found, no messages are displayed and the scan runs.
	If a scan file is found, the following message is added to the log file:
	`"Checking the age of Scan File "%s"`
	Where %s is the full name of the scan file it uses to check it.
	If there is a problem determining the age of the scan file (for example, if it is a newer version or it is corrupt), it then outputs:
	`The age of the Scan File cannot be determined.`
	If it does manage to obtain the date, it outputs:
	`Last scan was %d days ago`
	Where %d is substituted for an integer number.

Table 5 Command line parameters for scanners

Command Line Parameter	Function

-scandayofweek:< Number>	Scan only on specified day of week(0-Sun,1-Mon, etc).
	<N> can be one of the following:
	`0-Sunday`
	`1-Monday`
	`2-Tuesday`
	`3-Wednesday`
	`4-Thursday`
	`5-Friday`
	`6-Saturday`
	For example:
	`-scandayofweek:5`
	This will cause the scan to be performed on Fridays only.
	The scandays: and scandayofweek: options can be combined. For example:
	`ScanW32 -scandays:14 –scandayofweek:3`
	This causes the scan to be performed every other Wednesday.

-incl:<switch>	Switches for re-enabling individual hardware tests that were disabled in the Scanner Generator. To include tests 10, 20 and 50, you would run: `-incl:10 -incl:20 -incl:50`
-excl:< switch >	Switches for disabling individual hardware tests. To To exclude tests 10, 20 and 50, you would run: `-excl:10 -excl:20 -excl:50`

Table 5 Command line parameters for scanners

Command Line Parameter	Function
-paths	Using this switch, it is possible to define exactly which directories to scan; the parameter can be repeated as many times as necessary. For example: `scan -paths:/etc -paths:/var -paths:/bin` will scan just /etc, /var and /bin and their subdirectories. **Note:** You must ensure that the Allow Command Line Override option is checked in the Scanner Generator Software Data tab for this to work.

-o:<filename>	Takes the offsite scan file name from the command line.
	For example (non UNIX):
	`ScanW32 /o:r:\results\SC002154`
	Where `r:\results\SC002154` is the path to the file `SC002154`.
	If a file name is not entered, the file is named Default.xsf.
	If the path is not specified, the file is placed in the directory configured for offsite scan files in the Scanner Generator (see the *Customization Guide*).
	If the path is specified on the command line (even if it is relative), it replaces the path configured in the Scanner Generator. Here are some examples.
	`scanlnx -o:newname`
	Saves the offsite scan file, `newname.xsf,` to the location configured in the Scanner Generator.
	`scanlnx -o:/tmp/newname`
	Saves the offsite scan file to `/tmp/newname.xsf`.
	`scanlnx -o:subdir/newname`
	Saves the offsite scan file, `newname.xsf,` to the `subdir` subdirectory of the current directory.

-?	The full list of command line options can be obtained by running the scanners with the -? or /? command line option.

Viewing Command Line Options in Viewer or Analysis Workbench

If a command line option or switch has been used, it can be viewed in Analysis Workbench or Viewer.

This can be very useful when you want to check if the scan results were obtained from a scanner that had been run with any special command line options.

For example, if the scanner had been run with the -paths command:

```
scan -paths:/etc -paths:/var -paths:/bin
```

The -paths command line option will be displayed in Viewer (System Data folder in the Hardware and Configuration tab page).

Using Command Line Switches to Enable and Disable Specific Hardware Tests

Hardware test numbers that can be used for enabling/disabling hardware tests in the scanners as part of the `-excl` and `-incl` command line switches are shown in the following table:

Table 6 Hardware Tests to be used with -excl and -incl switches to Enable and Disable specific hardware tests

Hardware Test	Hardware Test
10 : BIOS Data	11 : BIOS Extension
12 : SMBIOS Information	13 : Compaq Asset Tag
14 : Plug and Play Version	30 : Video data
31 : Monitors	40 : Port data
50 : Keyboard and Mouse data	60 : Disk data
70 : Memory Data	72 : Swap Files
80 : CPU Data	90 : Operating System Data
91 : Device driver files	92 : Cluster Data
93 : Services	94 : Virtual Machine Data
95 : User profiles	96 : Installed Applications
97 : Containers	100 : Storage Data
101 : Devices	102 : SCSI/IDE serial numbers
110 : Network data	111 : TCP/IP data
112 : IPX data	113 : Netbios Data
114 : Network Shares	120 : Bus Data
121 : PCI Cards	122 : PCMCIA Cards
123 : MCA Cards	124 : EISA Cards
125 : ISA PnP Card detection	126 : USB Data
130 : Peripherals	150: System Configuration

Starting the Scanners

Information Collected by the Scanners

See **Help** > **Data Collected by the Scanners**.

Starting the Scanner Manually

Enterprise Discovery allows you to automatically launch your scanners using agents. We recommend that you use the agent and Enterprise Mode scanners to schedule scans regularly. However, if you need to launch them manually do the following:

Windows Scanners

The Win32 scanner comes in two versions, namely, normal and hidden.

The normal version of the Win32 scanner is a console application with a command line user interface. It shows command line output as it executes popping up a console window if one is not available.

The hidden version of the Win32 scanner is a GUI application that does not have a user interface. Since it is not a console application, it does not pop up a console window to display output. This is preferable if a completely hidden scanner is needed when executing the scanner in the manual deployment mode, for example, as part of the Windows login script. In such cases, the hidden Win32 scanner can be used. Unlike the normal Win32 scanner, it requires no console, shows no output, and executes in a completely hidden manner. By default it is called scanW3H.exe.

➤ As the hidden scanner has no way to interact with the user, if an error is encountered during its operation, it will fail with the appropriate error level. The reason for failing can usually be found in the error log file.

To start the normal Win32 scanner

1. Locate the scanner by using the Windows Explorer.

   Scanners are located in `<`*`installDir`*`>\Enterprise Discovery\2.20\scanners\scanners`, where `<`*`installDir`*`>` is the Enterprise Discovery installation directory. By default `<`*`installDir`*`>` is `C:\Program Files\HP OpenView`.

2. Double-click on the scanner icon or file name.

   Alternatively, you can start the Windows scanner from a command prompt.

To start the hidden Win32 scanner

1. In a command prompt window, change to the directory where the scanner program is located.
2. Type the name of the scanner program, for example, `scanW3H`, to start the scanner.

   Alternatively, you can start the Windows scanner by double-clicking on its file name in the Windows Explorer.

UNIX/Mac OS X Scanners

The methods for starting the various UNIX scanners (HP-UX, Solaris, Linux and AIX) are identical.

To start UNIX/Mac OS X scanners

1. Copy the scanner executable to the machine to be scanned.
2. Make sure that executable bit has been set (for example, for the Solaris scanner run `chmod +x scansp2` to ensure this)
3. Type the name of the scanner, for example, scansp2, followed by any desired scanner command line options, to run it.

   You will have to type ./ in front of scansp2 if the current directory (.) is not in the `PATH: ./scansp2`

The Scanning Sequence for the Scanners

A console is shown while the scanner is running. This displays the status of the scanning sequence. Any errors encountered are also shown here.

After the scan has been executed, the following events take place:

- Hardware scan (also contains system configuration scan)
- Software scan

Hardware Scan

Initially a copyright message is displayed, after which, hardware is detected (this too, is indicated as text mode messages).

Software Scan

The software scan commences after the hardware scan. It shows a list of directories as they are being scanned.

Scanner Error Level Codes

The scanners produce error level codes which can be used to handle situations if the scanner terminates without producing a scan file.

These error codes can, for example, be used in a batch file so that specified actions can be carried out in the event that particular error codes are returned.

These can be used to control re-scan activities when a scan has not completed successfully.

Table 7 Scanner Error Level Codes

Error Level	Description
20	Scanner terminated because virtual machine was detected
6	Another scanner instance is already running.
5	Too Early – It is earlier than the scan days variable.
4	Fatal Error – Scanner encountered a fatal error.
3	Help Screen – Command line help screen has been requested. It is also returned if invalid command line options are specified.
2	User Abort – User aborted the scanner.
1	Exception – Scanner terminated because of an exception
0	Normal/successful exit

See Using Error Level codes on page 81 for further information.

The MSI Scanner

In This Section...

- Overview of the MSI Scanner on page 79
- Starting the MSI Scanner on page 79
- Opening the MSI Scanner Output File in the MSI Importer on page 79
- MSI Scanner Error Level Codes on page 80

Overview of the MSI Scanner

The MSI scanner is a command line utility used to scan an MSI based installer, extract all required file information and write an XML file describing the installer and its contents. This XML file can then be sent to the central office where the person maintaining the application library can load it into the SAI Editor exactly as if it was the original MSI based Installer.

The MSI scanner (msiscanner.exe) is not generated by the Scanner Generator. It is supplied with the software in the following location by default: C:\Program Files\HP OpenView\Enterprise Discovery\2.20\Common\bin

Starting the MSI Scanner

To start the MSI scanner:

1 From the command prompt, type the following:

```
msiscanner <setup_package> <output_file>
```

Where:

- <setup_package> is the path and file name of the MSI-based installer.
- <output_file> is the path and file name of the output XML file. Note that if the specified file name does not end in .xml, the MSI scanner will append an .xml extension to it.

Opening the MSI Scanner Output File in the MSI Importer

The output from the MSI scanner is usable in the MSI Importer so that you can browse the MSI and teach from it based on the XML file only.

To open the MSI scanner output file:

1 In the SAI Editor, select the Import MSI based Installer option from the Tools menu.

The File Open dialog box is displayed.

2 In the Files of Type drop-down box, select the MSI scanner output file.

3 Navigate to the file to be opened.

4 Click OK

MSI Scanner Error Level Codes

The MSI scanner produce error level codes which can be used to handle situations if the scanner terminates without producing a scan file.

These error codes can, for example, be used in a batch file so that specified actions can be carried out in the event that particular error codes are returned.

Table 8 MSI Scanner Error Level Codes

Error Level	Description
6	Unexpected error
5	Unable to open the output file
4	Insufficient space available in the Temp directory.
3	Unable to open input MSI
2	Unrecognized package
1	Incorrect parameters
0	Success

Troubleshooting

Using Error Level codes

Windows scanners produce Error Level codes that can be used to handle situations if the scanners terminate without producing an audit file.

These can be used to control re-scan activities when a scan has not completed for some reason.

Because the Error Level is available as an environment variable when the scanner finishes this can be incorporated in a log file.

For example, a Windows NT/200x/XP/Vista Scanner with ComputerName and UserName available, a simple batch file could include:

```
echo %computername%, %errorlevel%, %username% to a flag file >

%computername%.flg
```

➤ If the scanner is terminated using the Windows Task Manager, then it is reported as successful.

Scanner Generator Errors

The most usual problems encountered when the scanner is generated result in an error message:

ERROR {value} Generating scanner

The causes can usually be identified as follows:

- The path defined for the executable scanner executable file does not exist.
- The scanner file already exists and is currently being used by another application.
- The file name chosen for the scanner executable file is invalid (that is, does not follow the MS DOS file naming conventions and may include illegal characters).
- Some virus protection software may prevent the Scanner Generator from creating and writing to scanner executable files.

To Resolve the Problem

- Try generating the scanner using the default settings, path and file name.
- If the previous step fails, try again selecting a file name which you have checked does not exist.

Hardware Scanning Errors

If a hardware scanning error occurs, the screen will appear to stop responding, or ‘hang’ during hardware scanning.

Note the test which is failing. The error message is displayed in the console/shell window.

The scanner provides several command line parameter switches for disabling specific hardware detection tests. Use these command line parameter switches to disable the specific test which has failed.

➤ Typing -? following the scanner file name at the command line displays a list of the available parameter switches, for example, `ScanW32 -?`

To use a parameter switch to disable a specific hardware test, enter it on the command line after the scanner file name when the scanner is started.

For example:

```
scanlnx -excl:60
```

Software Scanning Errors

The following errors may occur during the examination of files and collection of software information.

Out of Swap Space – Cannot Store More Files in Scan File

This message is displayed if there is not enough room on the hard disk drive for storing a file that has been marked for collection in the Scanner Generator.

Could Not read File <file name> - File Not Saved

This message is displayed if a file marked for collection in the Scanner Generator cannot be stored. Check to see if the file is being locked (used) by another process.

Scan File Saving Errors

The following messages may be displayed when the scanner tries to save a scan file:

Error {value} Saving Local Scan File

There may be insufficient space on the local drive that the scanner is attempting to save the scan file to. Check the available space on the local disk drive.

Another cause of this error message appearing might be that sufficient privileges do not exist to write the file or the drive cannot be accessed.

Error {value} Saving Offsite Scan File

There may be insufficient space on the offsite drive (for example the floppy disk or network drive) when the scanner is attempting to save the scan file. Check the available space on the offsite disk drive.

Additional Errors

Additional errors the user may encounter running the scanner include:

- Not Enough Temp Space
- Compression on Netware Servers
- Slow Scanning

- Virus Warning

Not Enough Temp Space

Check that the TEMP environment variable points to a valid directory with enough disk space available. If it is missing or points to an incorrect directory, set it up accordingly (for example: SET TEMP=C:\TEMP).

Compression on Netware Servers

All signatures should be off or override.ini must be set to ignore all files. This ensures that files are not opened and stored files are not collected. Netware compression is not dynamic such as NTFS compression in Windows NT/2000/XP/2003/Vista. Running scanners could have detrimental effects in Netware Servers if compression is being used. This is because to signature a file, the file must be decompressed and then opened by the scanner. Netware will not recompress the file, thus a capacity problem could result if the compressed volume is greater than the actual disk space available.

Slow Scanning

This may be due to real-time antivirus software being run. Any file that is opened will be checked for virus infection. Although this can be tedious, it is not advisable to disable the antivirus software for the reasons discussed in the next section.

Virus Warning

Because the scanner opens files on the computer, if there is real-time antivirus software in operation, it may detect a virus being present in a file. Depending on the virus product being used, they will have an action defined to deal with the virus. Some will try to deal with the problem and immediately disinfect the file. Others will try to move the infected file to a quarantine directory and rename its file extension.

In this case, the quarantine directory may be scanned by the scanner later during its scan.

To prevent this from happening, use the override.ini file with *.vir (where .vir is a typical quarantine file extension). Check the specific product to find the extension for this type of file.

Using the Scanners for Manual Inventories

Enterprise Discovery allows you to automatically launch your scanners using agents. We recommend that you use the Windows agent and Enterprise Mode to schedule scans regularly. However, Enterprise Discovery scanners can be generated as stand-alone executables that can be run in a number of ways.

Once you have configured and generated the correct type of scanners for your computer population, the next issue you will face is how to execute them.

Walkround Inventory

When starting your inventory project it may be necessary to initially conduct a walkround inventory. There may be machines that are not connected to the network, or there may be a closet full of older or broken machines which may only be discovered by physically finding them.

All of these machines need to be accounted for as part of a sound asset management program. Additionally, there is user asset information such as user first name, last name and location which must initially be manually entered.

With a walkround inventory, you can execute the scanner from a floppy disk, USB memory stick or connect to a network share and run it from there.

Using a Distribution Tool Such As SMS

The advantage of a distribution tool, such as Microsoft's SMS is, it allows an administrator to determine at their discretion when an inventory needs to take place. An administrator has the power from their System Management Console to push a command onto a remote machine at their will. This could include the execution of a scanner. The disadvantage of a product such as SMS is that an agent must be present on each desktop that the administrator would like to control. This requires time and expertise. Enterprise Discovery comes with a the capability to produce MIF files. These basically allow all SMS clients who are scanned to have their scan files converted to a standard MIF format which SMS can store, read and process.

Command Line Execution

Although the options for the scanner are normally set using the Scanner Generator, it may be necessary to change some settings to allow better operation on some machines being scanned. This may be to accommodate a 'quirky' machine or to simply change the name given to the scan file. The advantage of running a scanner from the command line is that there are numerous switches available to override options configured in the Scanner Generator. In addition, new features become available such as the option to run a scan on a scheduled basis.

For more information about command line options for the scanners, see Command Line Parameters and Switches on page 70.

7 Logging User Actions

Some users need a method of checking the Enterprise Discovery logs to see the actions initiated by different accounts.

The best way to find this kind of information is to go through the audit.log and discovery.log files. Both are available (by default) at this location: C:\Documents and Settings\All Users\Application Data\Peregrine\Enterprise Discovery\Logs.

By default, Enterprise Discovery does not log these events. If you would like to log them, you must enable the **Log User Actions** option at **Administration > System Configuration > Server configuration**.

Audit Log

Before using the audit log, make sure that the data you want to see is recorded in the audit log. The log contains information on most changes you make through the web user interface (for example, account changes and System Configuration). The resulting data appears in the following format:

```
2005-09-21 09:37:20,453 [4380] - OV::ED::Audit::log_info
C:\Perl\site\lib\/OV/ED/Audit.pm (46): "admin@127.0.0.1" set
"audit_log" === to y (ConfigOption)


2005-09-21 09:41:45,040 [4380] - OV::ED::Audit::log_info
C:\Perl\site\lib\/OV/ED/Audit.pm (46): "admin@127.0.0.1" set
"max_login_failure_count" === to 4 (ConfigOption)


2005-09-21 09:42:01,062 [4380] - OV::ED::Audit::log_info
C:\Perl\site\lib\/OV/ED/Audit.pm (46): "admin@127.0.0.1" add
"account" "test_account"


2005-09-21 09:42:17,562 [4380] - OV::ED::Audit::log_info
C:\Perl\site\lib\/OV/ED/Audit.pm (46): "admin@127.0.0.1" change
"password" of "test_account" === *****
```

For each entry, you will see the following information:

Example	Explanation
`2005-09-21 09:41:45,040 [4380] -`	Date and time of the change.

`OV::ED::Audit::log_info`
`C:\Perl\site\lib\/OV/ED/Audit.pm (46):`
	The name of the script.

`admin@127.0.0.1`	The account name and the IP address from which the server was accessed.
`set "audit_log" === to y (ConfigOption)`	The UI option that was changed.

Discovery Log

The Discovery log contains events relating to discovering the network. For example, it records whenever a user updates the model of a device, or adds a new device to the range of IP addresses.

If you would like to search the log for these events, enter the following grep command at the DOS prompt on your Enterprise Discovery server.

If you have installed Enterprise Discovery to its default location, you can use the following command. Enter the command exactly as shown, including all punctuation marks.

```
"C:\Program Files\HP OpenView\Enterprise
Discovery\2.20\support\bin\grep.exe" "Event write:"
"C:\Documents and Settings\All Users\Application Data\HP
OpenView\Enterprise Discovery\Logs\discovery.log" | "C:\Program
Files\HP OpenView\Enterprise
Discovery\2.20\support\bin\grep.exe" "User=" > "C:\events.txt"
```

This grep command filters the data in the Discovery.log file twice. First, it looks for data containing the text “Event write”. Second, it filters again by looking for the text “User=”.

When installing Enterprise Discovery 2.20, you may have changed the default location of the folders. If that is the case, make sure to change the command accordingly.

Example	Explanation
`"<program files>\support\bin\grep.exe"`	The location in your “program files” folder that contains the grep executable.
`"Event write:"`	The first filter for text to be found in the discovery log.
`"<data directory>\Logs\discovery.log" |`	The location of the discovery log in your “data directory”. There are likely several discovery.log files in this directory, as Enterprise Discovery will split the file once it reaches a certain size. So, you may have to search through several log files, all following this naming convention: discovery.log, discovery2.log, discovery3.log.
`"<program files>\support\bin\grep.exe"`	The location in your “program files” folder that contains the grep executable.
`"User=" >`	The second filter for text to be found in the discovery log.
`"C:\events.txt"`	The name of the output file.

The resulting “events.txt” file will contain data in the following format.

```
2005-09-13 17:08:49,267 LogServerClientHandler
ovedDiscEng:Event[6236]: INFO - Event write: change Object:
node=10 Property=Title Owner=User from=<xxxx> to=<yyyy>
User="abcd" FromIP="ipv4:172.1.1.1"


2005-09-13 17:09:10,249 LogServerClientHandler
ovedDiscEng:Event[6236]: INFO - Event write: Model update mdupd
FromIP="4D58DC30" User="abcd" Type="UpdateModel" Value="Query
Network" EventTime=1126645750 Node=10


2005-09-13 17:09:17,827 LogServerClientHandler
ovedDiscEng:Event[6236]: INFO - Event write: Model update mdupd
FromIP="ipv4:172.1.1.1" User="abcd" Type="UpdateModel"
Value="Run Rulebase" EventTime=1126645757 Node=10


2005-09-13 17:09:29,763 LogServerClientHandler
ovedDiscEng:Event[6236]: INFO - Event write: Model update mdupd
FromIP="ipv4:172.1.1.1" User="abcd" Type="UpdateModel"
Value="Enrich XML" EventTime=1126645769 Node=10


2005-09-13 17:09:39,903 LogServerClientHandler
ovedDiscEng:Event[6236]: INFO - Event write: delete auto_trash
node(s) 10 User="abcd" FromIP="ipv4:172.1.1.1"


2005-09-13 17:09:45,308 LogServerClientHandler
ovedDiscEng:Event[6236]: INFO - Event write: add new device
port(s) 501 User="abcd" FromIP="ipv4:172.1.1.1"


2005-09-13 17:09:45,339 LogServerClientHandler
ovedDiscEng:Event[6236]: INFO - Event write: add node(s) 10
User="abcd" FromIP="ipv4:172.1.1.1"


2005-09-13 17:12:53,539 LogServerClientHandler
ovedDiscEng:Portal.1[9840]: INFO - Event write: Network
Configuration changed by User=abcd FromIP=ipv4:172.1.1.1
```

8 UI Shortcuts

You can launch major components of Enterprise Discovery from outside of the Enterprise Discovery interface. These components include the Device Manager, Port Manager, Line Manager, and all features available from the Home Page.

To launch components from outside Enterprise Discovery, you must use the "?go=" commands. The "?go=" commands associated with the Home Page require only a single argument. The "?go=" commands associated with the Managers can have multiple arguments.

➤ To launch a component on a remote Enterprise Discovery server from a server running in Aggregator mode, use the optional argument “remote_id”.

Optional arguments are shown in [square brackets]. Variables (which you must replace with a value) are shown in angle brackets and <*this font*>. You should omit the square brackets, angle brackets, and spaces between arguments when you type the actual text.

Major Components

You can launch the following major components with a single argument from a web browser:

Function	**Command**
Health Panel	https://<*my_server*>/nm/?go=health_panel
Network Map	https://<*my_server*>/nm/?go=network_map
Events Browser	https://<*my_server*>/nm/?go=events
Find	https://<*my_server*>/nm/?go=find
MIB Browser	https://<*my_server*>/nm/?go=mib_browser
Scan Data Viewer	https://<*my_server*>/nm/?go=viewer
Home	https://<*my_server*>/nm/?go=home
Status	https://<*my_server*>/nm/?go=status
Reports	https://<*my_server*>/nm/?go=reports
Administration	https://<*my_server*>/nm/?go=administration
Help	https://<*my_server*>/nm/?go=help

Asset Questionnaire

Syntax:

https://<*my_server*>/nm/?go=waq ;device=<*device_id*> [;device_type=<*device_type*>]

Parameter	**Description**
device_id	any string
device_type	one of the following options: OID, NMID, PortOID, PortNMID, IP, IPv4, IPv6, MAC, Cloud, DNS, LabelPrefix, Label, NetBIOS

Examples:

- `https://my_server.example.com/nm/?go=waq;device=172.17.1.1`
- `https://my_server.example.com/nm/?go=waq;device=172.17.1.1;device_type=IPv4`

Device Manager

Syntax:

https://<*my_server*>/nm/?go=device ;device=<*device_id*> [;device_type=<*device_type*>] [;panel=<*panel*>]

Parameter	**Description**
device_id	any string
device_type	one of the following options: OID, NMID, PortOID, PortNMID, IP, IPv4, IPv6, MAC, Cloud, DNS, LabelPrefix, Label, NetBIOS
panel	one of the following options: about, state, reports, diagnosis, stats, ports, manage, update, visibility Note: If this is omitted, Enterprise Discovery will use the account's default setting.

Examples:

- `https://my_server.example.com/nm/?go=device;device=172.17.1.1` (open device by IP address)
- `https://my_server.example.com/nm/?go=device;device=172.17.1.1;device_type=IPv4` (open device by IP address; more efficient)

- `https://my_server.example.com/nm/?go=device;device=56;device_type=NMID`
(open device by internal ID)

Port Manager

Syntax:

https://<*my_server*>/nm/?go=port [;device=<*device_id*>] [;device_type=<*device_type*>] ;port=<*port_id*> [;port_type=<*port_type*>] [;panel=<*panel*>]

Parameter	Description
device_id	any string
device_type	one of the following options: OID, NMID, PortOID, PortNMID, IP, IPv4, IPv6, MAC, Cloud, DNS, Label, LabelPrefix, NetBIOS
port_id	any string
port_type	one of the following options: OID, NMID, Index, Description
panel	one of the following options: about, state, reports, diagnosis, stats, purge, connect, disconnect Note: If this is omitted, Enterprise Discovery will use the account's default setting.

Examples:

- `https://my_server.example.com/nm/?go=port;device=172.17.1.1;port=eth0`
(open port by IP address and description)
- `https://my_server.example.com/nm/?go=port;port=238;port_type=NMID`
(open port by internal ID)

Line Manager

Syntax:

https://<*my_server*>/nm/?go=line [;device=<*device_id*>] [;device_type=<*device_type*>] ;port=<*port_id*> [;port_type=<*port_type*>] [;panel=<*panel*>]

Parameter	**Description**
device_id	any string
device_type	one of the following options: OID, NMID, PortOID, PortNMID, IP, IPv4, IPv6, MAC, Cloud, DNS, Label, LabelPrefix, NetBIOS
port_id	any string
port_type	one of the following options: OID, NMID, Index, Description
panel	about (the only option available)

This works the same as the Port Manager. The only additional restriction is that the Line Manager will only work if the specified port is connected to something. You may specify either end of the line.

Examples:

```
https://my_server.example.com/nm/?go=line;device=172.17.1.1;port=eth0
```
(open line by IP address and description)

```
https://my_server.example.com/nm/?go=line;port=238;port_type=NMID
```
(open line by internal ID)

Attribute Manager

Syntax:

https://<*my_server*>/nm/?go=attribute [;device=<*device_id*>] [;device_type=<*device_type*>] [;port=<*port_id*>] [;port_type=<*port_type*>] ;attribute=<*attribute_id*> [;attribute_type=<*attribute_type*>] [;panel=<*panel*>]

Parameter	**Description**
device_id	any string
device_type	one of the following options: OID, NMID, PortOID, PortNMID, IP, IPv4, IPv6, MAC, Cloud, DNS, Label, LabelPrefix, NetBIOS
port_id	any string
port_type	one of the following options: OID, NMID, Index, Description
attribute_id	any string
attribute_type	A full list of the internal names of these attributes is available on the web UI at **Help > Classifications > Supported Device/Port Attributes**.
panel	one of the following options: about, stats Note: If this is omitted, Enterprise Discovery will use the account's default setting.

Examples:

- `https://my_server.example.com/nm/?go=attribute;device=172.17.1.1;port=eth0;attribute=in_util;attribute_type=Name`
  (open attribute by IP address and description and show the utilization in attribute)
- `https://my_server.example.com/nm/?go=attribute;attribute=49234;attribute_type=NMID`
  (open attribute by internal ID)

Service Analyzer

Syntax:

https://<*my_server*>/nm/?go=service_analyzer [;device1=<*device1_id*>] [;device1_type=<*device1_type*>] [;device2=<*device2_id*>] [;device2_type=<*device2_type*>]

Parameter	**Description**
device1_id	any string
device1_type	one of the following options: OID, NMID, PortOID, PortNMID, IP, IPv4, IPv6, MAC, Cloud, DNS, LabelPrefix, Label, NetBIOS
device2_id	any string
device2_type	one of the following options: OID, NMID, PortOID, PortNMID, IP, IPv4, IPv6, MAC, Cloud, DNS, LabelPrefix, Label, NetBIOS

Examples:

-
  ```
  https://my_server.example.com/nm/
  ?go=service_analyzer;device1=172.17.1.1;device2=nmc
  ```

-
  ```
  https://my_server.example.com/nm/
  ?go=service_analyzer;device1=32;device1_type=NMID;device2=78;device2_type
  =NMID
  ```

9 Copyright

HP acknowledges the copyrights belonging to the following third parties. (This page constitutes a continuation of the copyright page.)

ActivePerl

Commercial support for ActivePerl is available through ActiveState at: http://www.ActiveState.com/Support/Enterprise/.

For peer support resources for ActivePerl issues see: http://www.ActiveState.com/Support/

ActivePerl is the up-to-date, quality-assured Perl binary distribution from ActiveState. Current releases, and other professional tools for open source language developers are available at http://www.ActiveState.com.

Apache Ant

This product includes software developed by The Apache Software Foundation (http://www.apache.org/).

This product includes also software developed by :

- the W3C consortium (http://www.w3c.org) ,

- the SAX project (http://www.saxproject.org)

Please read the different LICENSE files present in the root directory of this distribution.

Apache HTTPD

This product includes software developed by The Apache Software Foundation (http://www.apache.org/).

Portions of this software were developed at the National Center for Supercomputing Applications (NCSA) at the University of Illinois at Urbana-Champaign.

This software contains code derived from the RSA Data Security Inc. MD5 Message-Digest Algorithm, including various modifications by Spyglass Inc., Carnegie Mellon University, and Bell Communications Research, Inc (Bellcore).

Regular expression support is provided by the PCRE library package, which is open source software, written by Philip Hazel, and copyright by the University of Cambridge, England. The original software is available from ftp://ftp.csx.cam.ac.uk/pub/software/programming/pcre/

Apache Tomcat

This product includes software developed by The Apache Software Foundation (http://www.apache.org/).

Java™ Management Extensions (JMX) support is provided by the MX4J package, which is open source software. The original software and related information is available at http://mx4j.sourceforge.net.

The Windows Installer is built with the Nullsoft Scriptable Install Sysem (NSIS), which is open source software. The original software and related information is available at http://nsis.sourceforge.net.

Commons-logging

This product includes software developed by The Apache Software Foundation (http://www.apache.org/).

Commons-lang

This product includes software developed by The Apache Software Foundation (http://www.apache.org/).

cURL

Flex

This product includes software developed by the University of California, Berkeley and its contributors.

GD

Portions relating to JPEG and to color quantization copyright 2000, 2001, 2002, 2003, 2004, Doug Becker and copyright (C) 1994, 1995, 1996, 1997, 1998, 1999, 2000, 2001, 2002, 2003, 2004 Thomas G. Lane. This software is based in part on the work of the Independent JPEG Group. See the file README-JPEG.TXT for more information.

Portions relating to GIF compression copyright 1989 by Jef Poskanzer and David Rowley, with modifications for thread safety by Thomas Boutell.

Portions relating to GIF decompression copyright 1990, 1991, 1993 by David Koblas, with modifications for thread safety by Thomas Boutell.

Portions relating to WBMP copyright 2000, 2001, 2002, 2003, 2004 Maurice Szmurlo and Johan Van den Brande.

Portions relating to GIF animations copyright 2004 Jaakko Hyvätti (jaakko.hyvatti@iki.fi)

Permission has been granted to copy, distribute and modify gd in any context without fee, including a commercial application, provided that this notice is present in user-accessible supporting documentation.

This does not affect your ownership of the derived work itself, and the intent is to assure proper credit for the authors of gd, not to interfere with your productive use of gd. If you have questions, ask. "Derived works" includes all programs that utilize the library. Credit must be given in user-accessible documentation.

This software is provided "AS IS." The copyright holders disclaim all warranties, either express or implied, including but not limited to implied warranties of merchantability and fitness for a particular purpose, with respect to this code and accompanying documentation.

Although their code does not appear in the current release, the authors also wish to thank Hutchison Avenue Software Corporation for their prior contributions.

Getopt

This product includes software developed by the NetBSD Foundation, Inc. and its contributors.

LGPL

GNU C library version 2.2.3 for Linux Copyright (C) 1991-2001 Free Software Foundation

The Linux versions of the Enterprise Discovery scanner and agent are linked with GNU libc statically. This is done to allow wider platform support eliminating the dependency on dynamically linked shared libraries.

To comply with the requirements of LGPL under which libc is licensed (see the license itself below), HP Software makes an offer to provide the object code of the Linux scanner and agent together with the source code for GNU libc, other third party files used and instructions that enable the re-linking of the scanner or agent code with a different version of GNU libc.

This offer is valid for 3 years since the release of the software.

Please contact technical support for further details and to receive your copy.

GNU LIBRARY GENERAL PUBLIC LICENSE

Version 2, June 1991

Copyright (C) 1991 Free Software Foundation, Inc.

[This is the first released version of the library GPL. It is numbered 2 because it goes with version 2 of the ordinary GPL.]

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users.

This license, the Library General Public License, applies to some specially designated Free Software Foundation software, and to any other libraries whose authors decide to use it. You can use it for your libraries, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library, or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link a program with the library, you must provide complete object files to the recipients so that they can relink them with the library, after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

Our method of protecting your rights has two steps: (1) copyright the library, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the library.

Also, for each distributor's protection, we want to make certain that everyone understands that there is no warranty for this free library. If the library is modified by someone else and passed on, we want its recipients to know that what they have is not the original version, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that companies distributing free software will individually obtain patent licenses, thus in effect transforming the program into proprietary software. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License, which was designed for utility programs. This license, the GNU Library General Public License, applies to certain designated libraries. This license is quite different from the ordinary one; be sure to read it in full, and don't assume that anything in it is the same as in the ordinary license.

The reason we have a separate public license for some libraries is that they blur the distinction we usually make between modifying or adding to a program and simply using it. Linking a program with a library, without changing the library, is in some sense simply using the library, and is analogous to running a utility program or application program. However, in a textual and legal sense, the linked executable is a combined work, a derivative of the original library, and the ordinary General Public License treats it as such.

Because of this blurred distinction, using the ordinary General Public License for libraries did not effectively promote software sharing, because most developers did not use the libraries. We concluded that weaker conditions might promote sharing better.

However, unrestricted linking of non-free programs would deprive the users of those programs of all benefit from the free status of the libraries themselves. This Library General Public License is intended to permit developers of non-free programs to use free libraries, while preserving your freedom as a user of such programs to change the free libraries that are incorporated in them. (We have not seen how to achieve this as regards changes in header files, but we have achieved it as regards changes in the actual functions of the Library.) The hope is that this will lead to faster development of free libraries.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, while the latter only works together with the library.

Note that it is possible for a library to be covered by the ordinary General Public License rather than by this special one.

GNU LIBRARY GENERAL PUBLIC LICENSE

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Library General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) The modified work must itself be a software library.

b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library

with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License.

Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also compile or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

c) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

d) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Library General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

Appendix: How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and a brief idea of what it does.>

Copyright (C) <year> <name of author>

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Library General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Library General Public License for more details.

You should have received a copy of the GNU Library General Public License along with this library; if not, write to the Free Software Foundation, Inc., 59 Temple Place - Suite 330, Boston, MA 02111-1307, USA

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the library `Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990

Ty Coon, President of Vice

That's all there is to it!

libicu

Copyright © 1995-2002 International Business Machines Corporation and others. All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the ìSoftwareî), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, provided that the above copyright notice(s) and this permission notice appear in all copies of the Software and that both the above copyright notice(s) and this permission notice appear in supporting documentation.

THE SOFTWARE IS PROVIDED ìAS ISî, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS NOTICE BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization of the copyright holder.

Log4j

This product includes software developed by The Apache Software Foundation (http://www.apache.org/).

mod-dav

This product includes software developed by Greg Stein <gstein@lyra.org> for use in the mod_dav module for Apache (http://www.webdav.org/mod_dav/).

Mod_Perl

The Apache Software License, Version 1.1

Copyright (c) 2000 The Apache Software Foundation. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. The end-user documentation included with the redistribution, if any, must include the following acknowledgment:

> This product includes software developed by the Apache Software Foundation (http://www.apache.org/)."

Alternately, this acknowledgment may appear in the software itself, if and wherever such third-party acknowledgments normally appear.

4. The names "Apache" and "Apache Software Foundation" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact apache@apache.org.

5. Products derived from this software may not be called "Apache", nor may "Apache" appear in their name, without prior written permission of the Apache Software Foundation.

THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This software consists of voluntary contributions made by many individuals on behalf of the Apache Software Foundation. For more information on the Apache Software Foundation, please see <http://www.apache.org/>.

Portions of this software are based upon public domain software originally written at the National Center for Supercomputing Applications, University of Illinois, Urbana-Champaign.

MySQL

Enterprise Discovery includes software whose copyright is owned by MySQL, A.B.

netaddr-ip

This software is (c) Luis E. Muñoz. It can be used under the terms of the perl artistic license provided that proper credit for the work of the author is preserved in the form of this copyright notice and license for this module.

net-snmp

Various copyrights apply to this package, listed in separate parts below.

Part 1: CMU/UCD copyright notice: (BSD like)

Copyright 1989, 1991, 1992 by Carnegie Mellon University; Derivative Work - 1996, 1998-2000 Copyright 1996, 1998-2000 The Regents of the University of California. All Rights Reserved

Permission to use, copy, modify and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appears in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of CMU and The Regents of the University of California not be used in advertising or publicity pertaining to distribution of the software without specific written permission.

Part 2: Networks Associates Technology, Inc copyright notice (BSD)

Copyright (c) 2001-2003, Networks Associates Technology, Inc. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met: (1) Redistributions of source code or in binary form must retain the above copyright notice, this list of conditions and the following disclaimer (THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDERS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE); (2) Neither the name of the Networks Associates Technology, Inc nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

Part 3: Cambridge Broadband Ltd. copyright notice (BSD)

Portions of this code are copyright (c) 2001-2003, Cambridge Broadband Ltd. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met: (1) Redistributions of source code or in binary form must retain the above copyright notice, this list of conditions and the following disclaimer (THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDER ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE); (2) The name of Cambridge Broadband Ltd. may not be used to endorse or promote products derived from this software without specific prior written permission.

Part 4: Sun Microsystems, Inc. copyright notice (BSD)

Copyright © 2003 Sun Microsystems, Inc., 4150 Network Circle, Santa Clara, California 95054, U.S.A. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met: (1) Redistributions of source code or in binary form must retain the above copyright notice, this list of conditions and the following disclaimer (THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDERS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE); (2) Neither the name of the Sun Microsystems, Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

Part 5: Sparta, Inc copyright notice (BSD)

Copyright (c) 2003, Sparta, Inc. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met: (1) Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer (THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDERS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR

BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE); (2) Neither the name of the Networks Associates Technology, Inc nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

Open_SSL

LICENSE ISSUES

The OpenSSL toolkit stays under a dual license, i.e. both the conditions of the OpenSSL License and the original SSLeay license apply to the toolkit. See below for the actual license texts. Actually both licenses are BSD-style Open Source licenses. In case of any license issues related to OpenSSL please contact openssl-core@openssl.org.

OpenSSL License

Copyright (c) 1998-2006 The OpenSSL Project. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. All advertising materials mentioning features or use of this software must display the following acknowledgment:

> “This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit. (http://www.openssl.org/)”

4. The names "OpenSSL Toolkit" and "OpenSSL Project" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact openssl-core@openssl.org.

5. Products derived from this software may not be called "OpenSSL" nor may "OpenSSL" appear in their names without prior written permission of the OpenSSL Project.

6. Redistributions of any form whatsoever must retain the following acknowledgment:

> “This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit (http://www.openssl.org/)"

THIS SOFTWARE IS PROVIDED BY THE OpenSSL PROJECT ``AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE OpenSSL PROJECT OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Quartz

This product includes software developed by the OpenSymphony Group (http"//www.opensymphony.com/).

Unicode

International Components for Unicode: Copyright (c) 1995-2003 International Business Machines Corporation and others. All rights reserved. Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, provided that the above copyright notice(s) and this permission notice appear in all copies of the Software and that both the above copyright notice(s) and this permission notice appear in supporting documentation.

Xalan

NOTICE file corresponding to the section 4d of the Apache License, Version 2.0, in this case for the Apache Xalan distribution.

This product includes software developed by The Apache Software Foundation (http://www.apache.org/).

Portions of this software was originally based on the following:

- software copyright (c) 1999-2002, Lotus Development Corporation., http://www.lotus.com.
- software copyright (c) 2001-2002, Sun Microsystems., http://www.sun.com.
- software copyright (c) 2003, IBM Corporation., http://www.ibm.com.
- voluntary contributions made by Ovidiu Predescu (ovidiu@cup.hp.com) on behalf of the Apache Software Foundation and was originally developed at Hewlett Packard Company.

Xerces

Copyright (c) 1999 The Apache Software Foundation. All rights reserved. This product includes software developed by the Apache Software Foundation ([1]http://www.apache.org/).

Index

Numerics

24 hours
- bridge aging interval, 47
- not seen, 66

A

ActivePerl, 95

ActiveState, 95

address types (definitions), 44

aging interval, 47

AIX Scanner
- default scanner name, 68
- scanning sequence, 77
- software scan, 77
- starting, 77

apache ant, 95

apache httpd, 95

apache tomcat, 96

Attribute Manager
- panel elements, 52

audit.log, 86

B

banner
- Device Manager, 53

brackets around numerals, 53

bridge aging, 47

C

CIDR notation, 46

CIR, redefining, 38

Command line
- hardware scanning errors, 81
- how to use with scanners, 70
- MSI scanner, 79
- options for scanners, 71
- starting unix scanners, 77
- viewing options in the analysis tools, 74

command line execution, 84

commons-lang, 96

commons-logging, 96

communication models
- FDDI, 39 to 40
- frame relay, 37 to 39

community strings
- definition, 46
- directed, 47
- multiple strings, 46
- SNMP traps, 47

CSV, 18

cURL, 96

D

definitions
- Enterprise Discovery terms and concepts, 50 to 54
- network terms and concepts, 44 to 49

device
- icon, 17
- title, 50

Device Adds/Deletes, 65

Device Changes, 65

Device Manager
- banner, 53
- panel elements, 52
- title, 53

directed community strings, 47

discovery.log, 87

Disk space
- compression on netware servers, 83
- not enough to run scanner, 83

domain name
- definition, 44

E

Errors
hardware scanning, 81
other scanner, 82
scan file saving, 82
scanner generation, 81
software scanning, 82

Exceptions, 65

F

FDDI, 39 to 40

flex, 96

frame relay, 37 to 39

G

gd, 96

getopt, 97

gray background
Manager data, 53

H

Hardware data
scanning errors, 81

Health Panel
Device Adds/Deletes, 65
Device Changes, 65
Exceptions, 65
Not Recently Seen, 65
Port Add/Deletes, 65
Port Changes, 65

HP-UX Scanner
default scanner name, 68
scanning sequence, 77
software scan, 77
starting, 77

HSRP, 40

I

icons
object label, 50

IP address
definition, 44

IPv4 address (definition), 44

IPv6 address (definition), 45

K

Keyboard
disabling hardware detection routine, 75

L

layers 2 and 3 (OSI), 48

libicu, 104

Linux Scanner
default scanner name, 68
scanning sequence, 77
software scan, 77
starting, 77

log4j, 105

M

MAC address (definition), 45
numeric, 45
with OUI, 45

management workstation requirements, 48

manual inventories, 84
command line execution, 84
using SMS, 84
walkaround, 84

mask, network see netmask

MIB, 44

mod_perl, 105

mod-dav, 105

Mouse
disabling hardware detection routine, 75

MSI Scanner
browsing the MSI in MSI Importer, 79
default scanner name, 68
error codes, 80
starting, 79
what it does, 79

multiple community strings, 46

mysql, 106

N

negative statistics, 53

netaddr-ip, 106

netmask notation (definition), 46

net-snmp, 106

Network inventory
using SMS, 84

Not Recently Seen, 65

O

object label, 50

open_ssl, 108

Organization Unique Identifier (OUI), 45

OSI model, 48

OUI, 45

P

parentheses around numerals, 53

PC requirements, 48

PC Scanners
- default scanner names, 68

Port Add/Deletes, 65

Port Changes, 65

Port Manager
- panel elements, 52

Q

quartz, 110

R

recorded events
- Device Adds/Deletes, 65
- Device Changes, 65
- Exceptions, 65
- Not Recently Seen, 65
- Port Add/Deletes, 65
- Port Changes, 65

S

Saving
- scan file errors, 82

scan execution, 21

scan files
- collect and store, 21
- enrichment, 22

Scanner
- default scanner name, 68

scanner deployment
- automatic, 21

scanners, 67

schedule, Network Discovery, 41

SMS, 84

SMT, 39

SNMP, 44
- traps, 47

Solaris Scanner
- scanning sequence, 77
- software scan, 77
- starting, 77

stale data, 53

Starting
- MSI scanner, 79
- UNIX scanners, 77
- windows scanners, 76

Support, 95

T

terminology definitions
- address types, 44
- community string, 46
- domain name, 44
- Enterprise Discovery, 50 to 54
- IP address, 44
- IPv4 address, 44
- IPv6 address, 45
- layers 2 and 3 (OSI), 48
- MAC address (numeric), 45
- MAC address (with OUI), 45
- netmask notation, 46
- network, 44 to 49
- OSI, 48

title
- device, 50
- icon, 53
- object, 53

title bar see banner

traps, SNMP, 47

U

unicode, 110

Unix Scanners
- default scanner names, 68
- scanning sequence, 77
- software scan, 77
- starting, 77

W

walkaround inventory, 84

Windows 16-bit Scanner
- error codes, 77
- information collected, 76
- starting, 76

Windows 32-bit Scanner
- default scanner name, 68
- error codes, 77
- information collected, 76
- starting, 76

X

Xalan, 110

xerces, 110