Network Manager IP Edition
Version 3 Release 9

Getting Started Guide

R2E2

Network Manager IP Edition
Version 3 Release 9

Getting Started Guide

R2E2

Note
Before using this information and the product it supports, read the information in "Notices" on page 55.

This edition applies to version 3, release 9 of IBM Tivoli Network Manager IP Edition (product number 5724-S45) and to all subsequent releases and modifications until otherwise indicated in new editions.

Contents

About this publication

IBM Tivoli Network Manager IP Edition provides detailed network discovery, device monitoring, topology visualization, and root cause analysis (RCA) capabilities. Network Manager can be extensively customized and configured to manage different networks. Network Manager also provides extensive reporting features, and integration with other IBM products, such as IBM Tivoli Application Dependency Discovery Manager, IBM Tivoli Business Service Manager and IBM Systems Director.

The *IBM Tivoli Network Manager Getting Started Guide* describes Describes how to set up IBM Tivoli Network Manager IP Edition after you have installed the product. This guide describes how to start the product, make sure it is running correctly, and discover the network. Getting a good network discovery is central to using Network Manager IP Edition successfully. This guide describes how to configure and monitor a first discovery, verify the results of the discovery, configure a production discovery, and how to keep the network topology up to date. Once you have an up-to-date network topology, this guide describes how to make the network topology available to Network Operators, and how to monitor the network. The essential tasks are covered in this short guide, with references to the more detailed, optional, or advanced tasks and reference material in the rest of the documentation set.

Intended audience

This publication is for administrators who need to install and set up Network Manager IP Edition.

Readers need to be familiar with the following topics:

- Network management
- Operating System configuration

IBM Tivoli Network Manager IP Edition works in conjunction with IBM Tivoli Netcool/OMNIbus; this publication assumes that you understand how IBM Tivoli Netcool/OMNIbus works. For more information on IBM Tivoli Netcool/OMNIbus, see the publications described in "Publications" on page vi.

What this publication contains

This publication contains the following sections:

- Chapter 1, "Network Manager architecture," on page 1

  Describes the architecture of Network Manager, and shows how the functions of the product can be classified into different layers.
- Chapter 2, "Getting started," on page 5

  Describes the typical tasks that you need to perform in order to become familiar with Network Manager.

Publications

This section lists publications in the Network Manager library and related documents. The section also describes how to access Tivoli publications online and how to order Tivoli publications.

Your Network Manager library

The following documents are available in the Network Manager library:

- *IBM Tivoli Network Manager IP Edition Release Notes*, GI11-9354-00

  Gives important and late-breaking information about IBM Tivoli Network Manager IP Edition. This publication is for deployers and administrators, and should be read first.

- *IBM Tivoli Network Manager Getting Started Guide*, GI11-9353-00

  Describes how to set up IBM Tivoli Network Manager IP Edition after you have installed the product. This guide describes how to start the product, make sure it is running correctly, and discover the network. Getting a good network discovery is central to using Network Manager IP Edition successfully. This guide describes how to configure and monitor a first discovery, verify the results of the discovery, configure a production discovery, and how to keep the network topology up to date. Once you have an up-to-date network topology, this guide describes how to make the network topology available to Network Operators, and how to monitor the network. The essential tasks are covered in this short guide, with references to the more detailed, optional, or advanced tasks and reference material in the rest of the documentation set.

- *IBM Tivoli Network Manager IP Edition Product Overview*, GC27-2759-00

  Gives an overview of IBM Tivoli Network Manager IP Edition. It describes the product architecture, components and functionality. This publication is for anyone interested in IBM Tivoli Network Manager IP Edition.

- *IBM Tivoli Network Manager IP Edition Installation and Configuration Guide*, SC27-2760-00

  Describes how to install IBM Tivoli Network Manager IP Edition. It also describes necessary and optional post-installation configuration tasks. This publication is for administrators who need to install and set up IBM Tivoli Network Manager IP Edition.

- *IBM Tivoli Network Manager IP Edition Administration Guide*, SC27-2761-00

  Describes administration tasks for IBM Tivoli Network Manager IP Edition, such as how to administer processes, query databases and start and stop the product. This publication is for administrators who are responsible for the maintenance and availability of IBM Tivoli Network Manager IP Edition.

- *IBM Tivoli Network Manager IP Edition Discovery Guide*, SC27-2762-00

  Describes how to use IBM Tivoli Network Manager IP Edition to discover your network. This publication is for administrators who are responsible for configuring and running network discovery.

- *IBM Tivoli Network Manager IP Edition Event Management Guide*, SC27-2763-00

  Describes how to use IBM Tivoli Network Manager IP Edition to poll network devices, to configure the enrichment of events from network devices, and to manage plug-ins to the Tivoli Netcool/OMNIbus Event Gateway, including configuration of the RCA plug-in for root-cause analysis purposes. This publication is for administrators who are responsible for configuring and running network polling, event enrichment, root-cause analysis, and Event Gateway plug-ins.

- *IBM Tivoli Network Manager IP Edition Network Troubleshooting Guide*, GC27-2765-00

  Describes how to use IBM Tivoli Network Manager IP Edition to troubleshoot network problems identified by the product. This publication is for network operators who are responsible for identifying or resolving network problems.
- *IBM Tivoli Network Manager IP Edition Network Visualization Setup Guide*, SC27-2764-00

  Describes how to configure the IBM Tivoli Network Manager IP Edition network visualization tools to give your network operators a customized working environment. This publication is for product administrators or team leaders who are responsible for facilitating the work of network operators.
- *IBM Tivoli Network Manager IP Edition Management Database Reference*, SC27-2767-00

  Describes the schemas of the component databases in IBM Tivoli Network Manager IP Edition. This publication is for advanced users who need to query the component databases directly.
- *IBM Tivoli Network Manager IP Edition Topology Database Reference*, SC27-2766-00

  Describes the schemas of the database used for storing topology data in IBM Tivoli Network Manager IP Edition. This publication is for advanced users who need to query the topology database directly.
- *IBM Tivoli Network Manager IP Edition Language Reference*, SC27-2768-00

  Describes the system languages used by IBM Tivoli Network Manager IP Edition, such as the Stitcher language, and the Object Query Language. This publication is for advanced users who need to customize the operation of IBM Tivoli Network Manager IP Edition.
- *IBM Tivoli Network Manager IP Edition Perl API Guide*, SC27-2769-00

  Describes the Perl modules that allow developers to write custom applications that interact with the IBM Tivoli Network Manager IP Edition. Examples of custom applications that developers can write include Polling and Discovery Agents. This publication is for advanced Perl developers who need to write such custom applications.
- *IBM Tivoli Monitoring for Tivoli Network Manager IP User's Guide*, SC27-2770-00

  Provides information about installing and using IBM Tivoli Monitoring for IBM Tivoli Network Manager IP Edition. This publication is for system administrators who install and use IBM Tivoli Monitoring for IBM Tivoli Network Manager IP Edition to monitor and manage IBM Tivoli Network Manager IP Edition resources.

Prerequisite publications

To use the information in this publication effectively, you must have some prerequisite knowledge, which you can obtain from the following publications:

- *IBM Tivoli Netcool/OMNIbus Installation and Deployment Guide*, SC23-9680

  Includes installation and upgrade procedures for Tivoli Netcool/OMNIbus, and describes how to configure security and component communications. The publication also includes examples of Tivoli Netcool/OMNIbus architectures and describes how to implement them.
- *IBM Tivoli Netcool/OMNIbus User's Guide*, SC23-9683

  Provides an overview of the desktop tools and describes the operator tasks related to event management using these tools.
- *IBM Tivoli Netcool/OMNIbus Administration Guide*, SC23-9681

Describes how to perform administrative tasks using the Tivoli Netcool/OMNIbus Administrator GUI, command-line tools, and process control. The publication also contains descriptions and examples of ObjectServer SQL syntax and automations.

- *IBM Tivoli Netcool/OMNIbus Probe and Gateway Guide*, SC23-9684

  Contains introductory and reference information about probes and gateways, including probe rules file syntax and gateway commands.

- *IBM Tivoli Netcool/OMNIbus Web GUI Administration and User's Guide* SC23-9682

  Describes how to perform administrative and event visualization tasks using the Tivoli Netcool/OMNIbus Web GUI.

Accessing terminology online

The IBM Terminology Web site consolidates the terminology from IBM product libraries in one convenient location. You can access the Terminology Web site at the following Web address:

http://www.ibm.com/software/globalization/terminology

Accessing publications online

IBM posts publications for this and all other Tivoli products, as they become available and whenever they are updated, to the IBM Knowledge Center Web site at:

http://www-01.ibm.com/support/knowledgecenter/

Network Manager documentation is located under the **Cloud & Smarter Infrastructure** node on that Web site.

Note: If you print PDF documents on other than letter-sized paper, set the option in the **File** > **Print** window that allows your PDF reading application to print letter-sized pages on your local paper.

Ordering publications

You can order many Tivoli publications online at the following Web site:

http://www.elink.ibmlink.ibm.com/publications/servlet/pbi.wss

You can also order by telephone by calling one of these numbers:

- In the United States: 800-879-2755
- In Canada: 800-426-4968

In other countries, contact your software account representative to order Tivoli publications. To locate the telephone number of your local representative, perform the following steps:

1. Go to the following Web site:

   http://www.elink.ibmlink.ibm.com/publications/servlet/pbi.wss
2. Select your country from the list and click **Go**. The Welcome to the IBM Publications Center page is displayed for your country.
3. On the left side of the page, click **About this site** to see an information page that includes the telephone number of your local representative.

Accessibility

Accessibility features help users with a physical disability, such as restricted mobility or limited vision, to use software products successfully.

Accessibility features

The following list includes the major accessibility features in Network Manager:

- The console-based installer supports keyboard-only operation.
- The console-based installer supports screen reader use.
- Network Manager provides the following features suitable for low vision users:
  - All non-text content used in the GUI has associated alternative text.
  - Low-vision users can adjust the system display settings, including high contrast mode, and can control the font sizes using the browser settings.
  - Color is not used as the only visual means of conveying information, indicating an action, prompting a response, or distinguishing a visual element.
- Network Manager provides the following features suitable for photosensitive epileptic users:
  - Web pages do not contain anything that flashes more than two times in any one second period.

The accessibility of the Network Manager Knowledge Center is described in the Knowledge Center itself.

Extra steps to configure Internet Explorer for accessibility

If you are using Internet Explorer as your web browser, you might need to perform extra configuration steps to enable accessibility features.

To enable high contrast mode, complete the following steps:

1. Click **Tools** > **Internet Options** > **Accessibility**.
2. Select all the check boxes in the Formatting section.

If clicking **View** > **Text Size** > **Largest** does not increase the font size, click **Ctrl +** and **Ctrl -**.

IBM® and accessibility

See the IBM Human Ability and Accessibility Center for more information about the commitment that IBM has to accessibility.

Tivoli technical training

For Tivoli technical training information, refer to the following IBM Tivoli Education Web site:

http://www.ibm.com/software/tivoli/education

Support information

If you have a problem with your IBM software, you want to resolve it quickly. IBM provides the following ways for you to obtain the support you need:

Online
Go to the IBM Software Support site at http://www.ibm.com/software/support/probsub.html and follow the instructions.

IBM Support Assistant
The IBM Support Assistant (ISA) is a free local software serviceability workbench that helps you resolve questions and problems with IBM software products. The ISA provides quick access to support-related information and serviceability tools for problem determination. To install the ISA software, go to http://www.ibm.com/software/support/isa

Conventions used in this publication

This publication uses several conventions for special terms and actions and operating system-dependent commands and paths.

Typeface conventions

This publication uses the following typeface conventions:

Bold

- Lowercase commands and mixed case commands that are otherwise difficult to distinguish from surrounding text
- Interface controls (check boxes, push buttons, radio buttons, spin buttons, fields, folders, icons, list boxes, items inside list boxes, multicolumn lists, containers, menu choices, menu names, tabs, property sheets), labels (such as **Tip:** and **Operating system considerations:**)
- Keywords and parameters in text

Italic

- Citations (examples: titles of publications, diskettes, and CDs)
- Words defined in text (example: a nonswitched line is called a *point-to-point* line)
- Emphasis of words and letters (words as words example: "Use the word *that* to introduce a restrictive clause."; letters as letters example: "The LUN address must start with the letter *L*.")
- New terms in text (except in a definition list): a *view* is a frame in a workspace that contains data
- Variables and values you must provide: ... where *myname* represents....

`Monospace`

- Examples and code examples
- File names, programming keywords, and other elements that are difficult to distinguish from surrounding text
- Message text and prompts addressed to the user
- Text that the user must type
- Values for arguments or command options

Operating system-dependent variables and paths

This publication uses environment variables without platform-specific prefixes and suffixes, unless the command applies only to specific platforms. For example, the directory where the Network Manager core components are installed is represented as NCHOME.

When using the Windows command line, preface and suffix environment variables with the percentage sign **%**, and replace each forward slash (**/**) with a backslash (\) in directory paths. For example, on Windows systems, NCHOME is %NCHOME%.

On UNIX systems, preface environment variables with the dollar sign **$**. For example, on UNIX, NCHOME is $NCHOME.

The names of environment variables are not always the same in the Windows and UNIX environments. For example, %TEMP% in Windows environments is equivalent to $TMPDIR in UNIX environments. If you are using the bash shell on a Windows system, you can use the UNIX conventions.

Chapter 1. Network Manager architecture

The Network Manager architecture can be divided into three layers: network layer, data layer, and visualization layer.

Network
The network layer interacts directly with the network. This layer contains network discovery and polling functionality. Network discovery retrieves topology data and network polling retrieves event data.

Data The data layer stores the topology data retrieved by network discovery and the event data retrieved by network polling. Network polling also includes storage of polled SNMP and ICMP data for reporting and analysis. This layer also provides root-cause analysis functionality that correlates topology and events to determine the source of network faults, and event enrichment functionality that adds topology data to events.

Visualization
The visualization layer provides the tools operators and administrators need to view topology, view events, and run network troubleshooting tools.

The following figure shows a conceptual overview of the Network Manager functional layers. Please note the following points when consulting the figure:

- It is possible to configure the Network Manager to include failover. This is not shown in the figure.
- Network Manager is designed to be installed with Tivoli Netcool/OMNIbus to enhance fault management, including root-cause analysis, and correlation of alerts with the network topology. This figure depicts a standard Network Manager installation, and shows Tivoli Netcool/OMNIbus handling the storage and management of network events and the Tivoli Netcool/OMNIbus Web GUI handling visualization of network events. The Tivoli Netcool/OMNIbus Web GUI was known as Netcool/Webtop in versions 2.2 and below.

  Note: Tivoli Netcool/OMNIbus is a separate product. If you do not already have OMNIbus then you must get a copy and install it. For more information, see the Network Manager installation documentation.
- The Tivoli Integrated Portal Web application framework is an application that runs GUIs from different Tivoli® products, including Network Manager. The GUIs represented in the following figure, including the topology visualization GUIs and the event visualization GUIs all run within the framework of the Tivoli Integrated Portal.
  - The topology visualization GUIs include single-portlet views, such as the Hop View, Network Views, and Structure Browser. Default topology views also include multi-portlet views, such as the Fault-Finding View and the Network Health View.
  - The Tivoli Netcool/OMNIbus Web GUI event visualization GUIs include the Active Event List, the Light Event List and the Table View.
  - Network administrators can also build their own multi-portlet views, which customize combinations of the single portlet views.

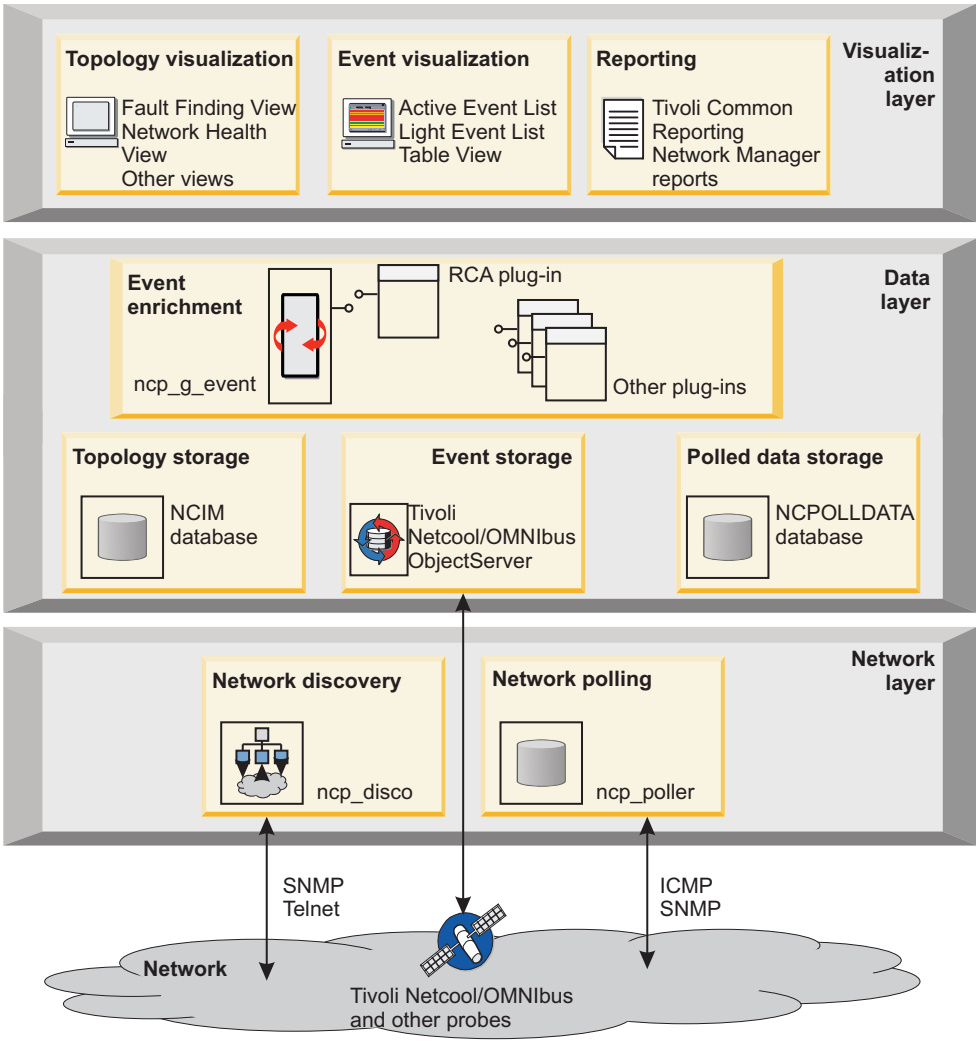

Figure 1. Network Manager functional layers

Network discovery

Network discovery involves discovering your network devices, determining how they are connected (network connectivity), and determining which components each device contains (containment). The complete set of discovered devices, connectivity, and containment is known as a network topology. You build your network topology by performing a discovery and then ensuring that you always have an up-to-date network topology by means of regular rediscoveries.

Network polling

Network polling determines whether a network device is up or down, whether it has exceeded key performance parameters, or whether links between devices are faulty. If a poll fails, Network Manager generates a device alert, which operators can view in the Active Event List

Topology storage

Network topology data is stored in the Network Connectivity and Inventory Model (NCIM) database. The NCIM database is a relational database that

consolidates topology data discovered by Network Manager. The NCIM database can be implemented using any one of the following relational database management systems: DB2®, Informix, MySQL, and Oracle

Event enrichment

Event enrichment is the process by which Network Manager adds topology data to events, thereby enriching the event and making it easier for the network operator to analyze. Examples of topology data that can be used to enrich events include system location and contact information.

Root-cause analysis

Root cause analysis is the process of determining the root cause of one or more device alerts. Network Manager performs root cause analysis by correlating event information with topology information. The process determines cause and symptom events based on the discovered network device and topology data.

Event storage

Event data is generated by Network Manager polls and also by Tivoli Netcool/OMNIbus probes installed on network devices. A probe is a protocol or vendor specific piece of software that resides on a device, detects and acquires event data from that device, and forwards the data to the ObjectServer as alerts. Event data can also be received from other event sources.

Event data from all of these event sources is stored in the Tivoli Netcool/OMNIbus ObjectServer.

Note: Tivoli Netcool/OMNIbus is a separate product. If you do not already have OMNIbus then you must get a copy and install it. For more information, see the Network Manager installation documentation.

Polled data storage

At any time a network administrator can set up polling of specific SNMP and ICMP data on one or more network devices. This data is stored in the NCPOLLDATA historical polled data database. By default, Network Manager implements the NCPOLLDATA database using a database schema within the NCIM database. You can optionally integrate Network Manager with IBM Tivoli Monitoring 6.2, with the integrated Tivoli Data Warehouse, to provide extra reporting capabilities, including better report response times, capacity, and isolation of the operational database (NCIM) from unpredictable reporting traffic.

Topology visualization

Network operators can use several topology visualization GUIs to view the network and to examine network devices. Using these GUIs operators can switch between topology views to explore connectivity or associations, and to see alert details in context. Operators also have access to diagnostic tools such as SNMP MIB Browser, which obtains MIB data for devices.

Event visualization

Operators can view event lists and use alert severity ratings to quickly identify high-priority device alerts. Operators can switch from event lists to topology views

to see which devices are affected by specific alerts. They can also identify root-cause alerts and list the symptom alerts that contribute to the root cause.

Reporting

Network Manager provides a wide range of reports, including performance reports, troubleshooting reports, asset reports, and device monitoring reports. Right click tools provide immediate access to reports from topology maps.

More information

For more information on the architecture and functionality of Network Manager, see the *IBM Tivoli Network Manager IP Edition Product Overview*.

Chapter 2. Getting started

After you have installed Network Manager, start the product, make sure it is running correctly, and discover your network.

After configuring a first discovery, verify the results, and configure a production discovery. Schedule further discoveries to keep the network topology up to date. Once you have an up-to-date network topology, you can make the network topology available to Network Operators, and monitor the network for problems.

Starting Network Manager

If they are installed on the same server, you can start the Tivoli Integrated Portal, Tivoli Netcool/OMNIbus, and all Network Manager processes, using the **itnm_start** command.

Restriction: The **itnm_start** command is applicable only to single server UNIX installations, and only on servers on which the Network Manager core components are installed, or Tivoli Integrated Portal is installed by the Network Manager installer. In the case of distributed installations, the **itnm_start** command is not available on servers that contain only Tivoli Netcool/OMNIbus, even if Tivoli Netcool/OMNIbus was installed by the Network Manager installer. For more information about starting and stopping Network Manager on a distributed installation or on a Windows installation see the *IBM Tivoli Network Manager IP Edition Administration Guide*.

Use the **itnm_start** command to start the following components:

- The Network Manager domain process controller, the **ncp_ctrl** process (which then starts all required Network Manager processes). Network Manager processes are responsible for network discovery, polling, root-cause analysis, network visualization and related activities.
- The Tivoli Integrated Portal. This controls the Web server that allows access to the user interface.
- Tivoli Netcool/OMNIbus. This manages collection, storage, display and processing of events.

To run the **itnm_start** command, complete the following steps.

1. Source the configuration file containing the Network Manager environment variables by entering the following command:
   `/opt/IBM/tivoli/netcool/env.sh`
2. Change to the `$NCHOME/precision/bin` directory.
3. Type this command:`itnm_start -domain` *`domain`* This command starts all of the Network Manager components that are installed on the server.

You must now ensure that all Network Manager processes are up and running.

In addition, ensure your database is running as well. For example, to start your DB2 database, log in as the DB2 database administrator and run the **db2start** command.

Ensuring that all processes are up and running

You can check that processes are up and running using the itnm_status command.

On UNIX, you can check the status of IBM Tivoli Netcool/OMNIbus, the Tivoli Integrated Portal, and Network Manager using the itnm_status command.

Restriction: This task only applies to single server UNIX installations. If you have a distributed or Windows installation (or both) then see *Checking process status* in the *IBM Tivoli Network Manager IP Edition Administration Guide*.

To check the status of all Network Manager components on the current server, complete the following steps.

1. Change to the `$NCHOME/precision/bin` directory.
2. Type this command: `itnm_status`. This command displays the status of all of the Network Manager components that are installed on the server.

Sample output: All processes up and running This output shows that all Network Manager, Tivoli Netcool/OMNIbus, and Tivoli Integrated Portal processes are up and running.

```
OMNIbus:
nco_pad RUNNING PID=11877 NCO_PA
nco_objserv RUNNING PID=11878 NCOMS

ITNM :
ncp_ctrl RUNNING PID=16621 NCOMS
ncp_store RUNN ING PID=16984 NCOMS
ncp_class RUNNING PID=16985 NCOMS
ncp_model RUNNING PID=17379 NCOMS
ncp_disco RUNNING PID=17434 NCOMS
ncp_d_helpserv RUNNING PID=17390 NCOMS
ncp_config RUNNING PID=16987 NCOMS
ncp_poller RUNNING PID=17485 NCOMS
nco_p_ncpmonitor RUNNING PID=16994 NCOMS
ncp_g_event RUNNING PID=17456 NCOMS

Tivoli Integrated Portal:
Server RUNNING PID=12887
```

Sample output: Startup problems This shows the output of the command in the case of startup problems. In this example, the following processes have not started: The Discovery Engine, ncp_disco, and the Polling Engine, ncp_poller.

```
OMNIbus:
nco_pad RUNNING PID=11877 NCO_PA
nco_objserv RUNNING PID=11878 NCOMS

ITNM :
ncp_ctrl RUNNING PID=16621 NCOMS
ncp_store RUNN ING PID=16984 NCOMS
ncp_class RUNNING PID=16985 NCOMS
ncp_model RUNNING PID=17379 NCOMS
ncp_disco NOT RUNNING
ncp_d_helpserv RUNNING PID=17390 NCOMS
ncp_config RUNNING PID=16987 NCOMS
ncp_poller NOT RUNNING
nco_p_ncpmonitor RUNNING PID=16994 NCOMS
ncp_g_event RUNNING PID=17456 NCOMS

Tivoli Integrated Portal:
Server RUNNING PID=12887
```

If you encounter startup problems, then complete the steps in the Troubleshooting startup problems procedure.

If all processes started up without any problems, then you can now log into Network Manager.

For more information about Network Manager processes, see the *IBM Tivoli Network Manager IP Edition Administration Guide*.

Troubleshooting startup problems

Troubleshoot startup problems by examining log files for the processes that are not starting up. Use process dependencies to help identify the origin of the startup problem.

The default name of the log file is the process name followed by the domain name and then the .log or .trace file extension. Complete these steps to locate a log file for a process.

1. Run the `itnm_status` command to determine which process or processes are failing to start. The `itnm_status` output might show, for example, that the ncp_disco process is failing to start.
2. Determine the process dependencies for the failing processes. Process dependencies are listed in the link below. For example, if the ncp_disco process is failing then the following dependencies apply:
   - The ncp_disco process depends on the Helper Server, ncp_d_helpserv, and the Topology Manager, ncp_model.
   - The ncp_d_helpserv process has no dependencies.
   - The ncp_model process depends on the Active Object Class manager, ncp_class.
3. Navigate to the default location for process log and trace files, *NCHOME*`/log/precision`.
4. Locate the log and trace files that correspond to the process name. For example, an instance of the ncp_disco process running on the NCOMS domain generates the following files.

   ncp_disco.NCOMS.log

   ncp_disco.NCOMS.trace
5. View the content of the log files for the failing process and for the process dependencies. Examine these files in reverse order of startup. For example, if the ncp_disco process is failing, then examine the log files in the following order
   - ncp_disco.*DOMAIN*.log
   - ncp_d_helpserv.*DOMAIN*.log
   - ncp_class.*DOMAIN*.log

For more information about troubleshooting the Tivoli Integrated Portal, see the log files in the following location: `NCHOME/precision/profiles/TIPProfile/logs`.

For more information about troubleshooting Network Manager processes and managing process dependencies, see the *IBM Tivoli Network Manager IP Edition Administration Guide*.

Logging into Network Manager

Use your Web browser to access the Network Manager user interface.

To check the settings for local host, port, and user ID, view the application server profile.

To log into Network Manager:

1. Open a supported browser.
2. Enter the URL of the application server: `https://localhost:16311/ibm/console` (secure access).

   Where *localhost* is the fully-qualified host name or IP address of the Tivoli Integrated Portal server.

   16310 is the default nonsecure port number and 16311 is the default secure port number. If your environment was configured during installation with a port number other than the default, enter that number instead.
3. On the login screen, enter your username and password as configured during installation, and click **Log in**. Your user credentials are stored in the browser session. If you open a Tivoli Integrated Portal GUI in another window of the same browser, you can use the GUI without logging in again.
4. When you have finished working with any Tivoli Integrated Portal GUI, log out using the **Logout** link, or close all browser windows. This prevents another user from accessing the GUI using your stored user credentials.

If you specified the parameters for an initial discovery during the installation, you can now view the network topology. If you did not specify the discovery parameters then you must now perform an initial discovery.

For more information about troubleshooting login problems, see the *IBM Tivoli Network Manager IP Edition Administration Guide*.

Accepting the security certificate

When logging in, you might see a security alert with a message that says there is a problem with the security certificate. This indicates that the browser application is verifying the security certificate of the application server.

Self-signed or CA-signed certificate

The application server uses a self-signed security certificate. You might see a Security Alert when you first connect to the portal that alerts you to a problem with the security certificate. You might be warned of a possible invalid certificate and be recommended to not log in.

Although this warning appears, the certificate is valid and you can accept it. Or, if you prefer, you can install your own CA-signed certificate. For information on creating your own CA-signed certificate, go to: http://www-01.ibm.com/support/knowledgecenter/SSEQTP_7.0.0/com.ibm.websphere.base.doc/info/aes/ae/tsec_sslcreateCArequest.html?cp=SSEQTP_7.0.0%2F1-8-30-4-7

For more information about certificates, go to the IBM WebSphere® Application Server Community Edition Documentation Project at https://www-01.ibm.com/support/knowledgecenter/SSEQTP/mapfiles/product_welcome_was.html?lang=en, and search for *Managing trust* and *Managing SSL certificates*.

Getting started with discovery

Using Network Manager you can discover your network and schedule regular discoveries to ensure that your network topology stays up to date.

This is the most important task after Network Manager has been installed. Configure and verify your network discovery to produce the most complete and accurate network topology possible for the devices and technologies in your network. An accurate network topology facilitates efficient root-cause analysis of network problems. This in turn enables your operators to troubleshoot network problems faster.

The most efficient approach to discovering your network is an iterative approach. This means beginning with a simple discovery and ensuring that the results are satisfactory. It is good practice to gradually configure more complex discoveries by adding extra scope zones to discover new regions of your network.

This information guides you through configuring and launching an initial discovery, and verifying the results. The information gives you guidance on how to identify problems with the discovered topology, and how to fix the discovery configuration in order to address these problems. Once you have a satisfactory topology that covers all regions of your network, the next step is to configure an efficient production discovery. The information shows you how to configure a production discovery, and how to schedule regular production discoveries to keep your discovered topology up to date with any network changes.

As you work through this information, you will perform the following discovery tasks:

- "Configuring initial discovery settings"
- "Launch the discovery and monitor discovery progress" on page 21
- "Verifying the topology" on page 29
- "Configuring production discovery settings" on page 37
- "Keeping topology up to date" on page 39

For complete information about discoveries, see the *IBM Tivoli Network Manager IP Edition Discovery Guide*.

Configuring initial discovery settings

Use this information to set up your initial discovery.

As you work through this information, you will perform the following discovery tasks:

- "Understanding scope settings" on page 10
- "Fine tuning a subnet scope zone using prediscovery filters" on page 13
- "Enabling the Ping finder and the feedback mechanism" on page 14
- "Configuring device and subnet access using SNMP community strings" on page 16
- "Ensuring all network technologies are covered" on page 20

Understanding scope settings

You can specify the boundaries of the network to discover by creating scopes that correspond to the IP addresses and subnets that you want to manage.

Let's suppose that your managed regions are made up of the following IP addresses and subnets:

- A single IP address, 10.30.1.20/32.
- The entire Class B subnet, 10.40.0.0/16, except for the Class C subnet 10.40.2.0/24.
- The first five IP addresses only in a set of subnets, 10.30.*.1-5.

Each of these regions can be defined using *scope zone*s. You can define as many scope zones as necessary for your network discovery.

On your own: Try to identify the boundaries of your own network. How would you express these boundaries using IP addresses, subnets, or formulas similar to the managed regions above?

In the next steps you are going to define each of the three managed regions listed above using scope zones.

Setting a single IP address scope zone

Set a single IP address scope zone when one of your managed regions is made up of a single IP address only; for example, 10.30.1.20/32.

1. Click **Discovery** > **Network Discovery Configuration**. This takes you to the Discovery Configuration GUI, where you can configure settings for your discovery. Network Manager also offers advanced users the option of configuring discovery using text files. The Discovery Configuration GUI provides users less familiar with the discovery process with an easier way of configuring the discovery. When you save your settings in the Discovery Configuration GUI, the discovery configuration settings are written to the discovery text files.
2. From the **Domain** list, select the required domain.
3. Click **Scope**.
4. To add a new scope zone, click **New**  . The Scope Properties page is displayed.
5. Ensure that the **Protocol** setting is IPv4. Our single IP address scope zone, 10.30.1.20/32, is an IPv4 address, so this setting can stay as it is.
6. Ensure that **Scope By** ***Subnet** is selected, and type the IP address `10.30.1.20` in the **Subnet** field. In the adjacent field following the slash sign, **/**, type the subnet mask `32`.
7. Ensure that the **Action** setting is set to **Include**. This defines the single IP address scope zone, 10.30.1.20/32 as an *inclusion zone*, an area of the network to be included in the discovery process.
8. Ensure that **Add to Ping Seed List** is checked. Clicking this option automatically adds the device in the scope zone to the *ping seed list*. This is a list of discovery seed devices, the locations from which to begin discovering devices. Discovery seeds can be IP addresses, as in this case, or subnet addresses. Clicking this option saves you having to separately enter the same entry in the **Seeds** tab.
9. Click **OK** to add this scope zone.

10. Click **Save** to save your discovery configuration settings.

You have now configured a single IP address scope zone consisting of the IP address 10.30.1.20/32. The next step is to configure a subnet scope zone for the entire Class B subnet, 10.40.0.0/16, but excluding the Class C subnet 10.40.2.0/24.You configured this single IP address as an inclusion zone, an area of the network to be included in the discovery process. When you configured this zone, you also added the IP address to the ping seed list. This is a set of IP addresses from which the discovery starts discovering the network.

The next step is to configure a scope zone for the managed region made up of the the entire Class B subnet, 10.40.0.0/16, but excluding the Class C subnet 10.40.2.0/24.

Setting a subnet scope zone

Set a subnet scope zone when one of your managed regions is made up of a complete subnet.

A subnet scope zone can be an *inclusion zone*, an area of the network to be included in the discovery process, or an *exclusion zone*, an area of the network to be excluded from the discovery process. Let's suppose that your managed regions include the following subnet scope zones: the entire Class B subnet, 10.40.0.0/16 (inclusion zone), except for the Class C subnet 10.40.2.0/24 (exclusion zone). This can be represented graphically as follows. Only devices in the darker area are included in the discovery.

Note: The exclusion zone must be a subset of an inclusion zone otherwise you may find that the discovery has no boundaries. If you make the exclusion zone a subset of the inclusion zone then everything outside the exclusion zone becomes in scope.

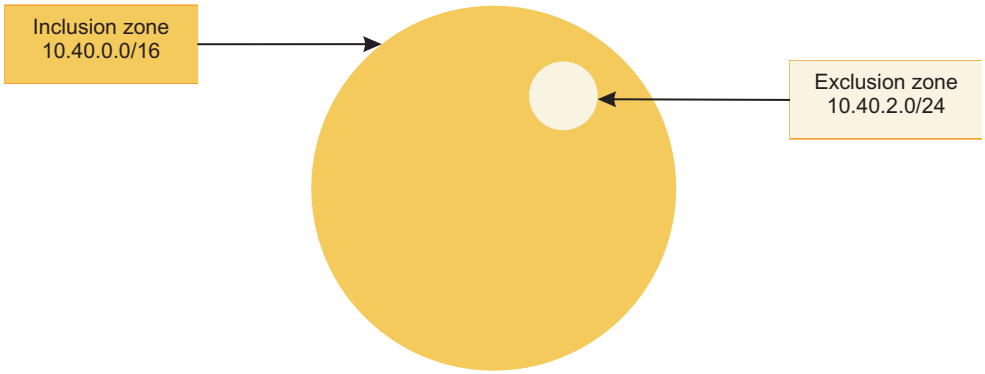

Figure 2. Exclusion zone within an inclusion zone

To create the exclusion zone within the inclusion zone, first add the Class B subnet, 10.40.0.0/16 as an inclusion zone, and then add the Class C subnet 10.40.2.0/24 as an exclusion zone.

1. Click **Scope**. This takes you back to the **Scope Configuration** section of the GUI, where you can configure the subnet scopes.
2. To add a new scope zone, click **New** . The Scope Properties page is displayed.

3. Leave the **Protocol** setting at IPv4. This is the required protocol setting as the Class B subnet inclusion zone 10.40.0.0/16 is an IPv4 address.
4. Ensure that **Scope By *Subnet** is selected, and type the IP address `10.40.0.0` in the **Subnet** field. In the adjacent field following the slash sign, /, type the subnet mask `16`.
5. Ensure that the **Action** setting is set to **Include**. This defines the Class B subnet scope zone, 10.40.0.0/16 as an inclusion zone.
6. Ensure that **Add to Ping Seed List** is checked.

   Clicking this option automatically adds all the devices in the Class B subnet scope zone to the ping seed list. The discovery process will therefore attempt to ping every single device in this scope zone. This is known as *ping sweeping*.

   Ping sweeping results in long discoveries, as the discovery process has to try every single possible IP address in the subnet. Class B subnets can take up to two to three hours to ping sweep. Class A networks can take a day or more to ping sweep. However, ping sweeping takes minimal effort to configure and is useful when you are configuring initial discoveries as it enables the system to automatically discover all devices within scope. Later, when you have successfully discovered your managed network and want to schedule more efficient production discoveries, you can generate a list of discovered IP addresses, and use this as the ping seed list rather than ping sweeping.

   Restriction: The `Add to Ping Seed List` option is not available for IPv6 scope zones. This prevents ping sweeping of IPv6 subnets, which can potentially contain billions of devices to be pinged. Ping sweeping of IPv6 subnets can therefore result in a non-terminating discovery.

   Ping sweeping relies on an active Ping finder. The discovery process uses the Ping finder to find devices specified in the ping seed list. You will enable the Ping finder as part of one of the next tasks.
7. Click **OK** to add this scope zone.
8. To add the Class C subnet 10.40.2.0/24 exclusion zone, click **New**  .
9. Ensure that **Scope By *Subnet** is selected, and type the IP address `10.40.2.0` in the **Subnet** field. In the adjacent field following the slash sign, /, type the subnet mask `24`.
10. Under **Action** click **Exclude**. This defines the Class C subnet 10.40.2.0/24 as an exclusion zone.
11. Click **OK** to add this scope zone.
12. Click Save  to save your discovery configuration settings.

You have now configured a subnet scope zone for the entire Class B subnet, 10.40.0.0/16, but excluding the Class C subnet 10.40.2.0/24. You did this by creating an inclusion zone and an exclusion zone. You configured ping sweeping of your Class B subnet inclusion zone. A ping sweep of this size of subnet will take up to two to three hours as every single possible IP address in the subnet has to be pinged. This is acceptable for initial discoveries as it enables the discovery process to identify all devices in scope. Later you will configure a more efficient production discovery that uses the results of your initial discoveries as ping seeds.

The next step is to configure a managed region made up of the first five IP addresses only in a set of subnets, 10.30.*.1-5.

Fine tuning a subnet scope zone using prediscovery filters

You can also restrict discovery to more complex IP address ranges. For example, you can configure your managed regions to include the first five addresses only in a set of subnets 10.30.*.1-5. The *prediscovery filter* is a mechanism that allows you to fine tune your discovery scope.

One way to do this is to first create a scope zone for the Class B subnet 10.30.0.0/16. Then restrict the discovered devices to the desired range using a *prediscovery filter*. All IP addresses within the defined scope zone are pinged initially and SNMP polled to retrieve the device sysObjectId. However, any device that does not pass the prediscovery filter is dropped from the discovery, is not queried by discovery agents, and is not included in the topology.

Test:

1. Click **Scope**.
2. To add a new scope zone, click **New**  . The Scope Properties page is displayed.
3. Leave the **Protocol** setting at IPv4. This is the required protocol setting as the Class B subnet inclusion zone 10.30.0.0/16 is an IPv4 address.
4. Ensure that **Scope By *Subnet** is selected, and type the IP address `10.30.0.0` in the **Subnet** field. In the adjacent field following the slash sign, **/**, type the subnet mask `16`.
5. Ensure that the **Action** setting is set to **Include**. This defines the Class B subnet scope zone, 10.30.0.0/16 as an inclusion zone.
6. Ensure that **Add to Ping Seed List** is checked. Clicking this option ensures that all devices in Class B subnet scope zone, 10.30.0.0/16 are pinged and SNMP polled. Data from the results of these ping and poll operations, for example the device sysObjectId is therefore available for use in prediscovery filters.
7. Click **OK** to add this scope zone.
8. Click **Filters**. This is the section of the GUI where you can create prediscovery filters.
9. In the **Pre-Discovery Filter** section of the panel, click **Filter Library**.
10. In the **Pre-Discovery Filter Library** window, click **Add Filter**.
11. Create a filter row to checks any devices discovered to make sure that they match one of the following:
    - 10.30.*.1
    - 10.30.*.2
    - 10.30.*.3
    - 10.30.*.4
    - 10.30.*.5

    a. In the **Name** field, type `Restrict 10.30.0.0 subnet`. A meaningful name of this sort helps you and others when referencing the filter later.

    b. In the **Basic** tab, select the **m_UniqueAddress** field from the **Field** list.

    The fields in this list represent data retrieved from each device during the early stages of discovery by the Details *discovery agent*. Discovery agents retrieve information about devices in the network. The Details agent is the first agent to run on each device and retrieves basic information about devices whose existence has already been verified. The fields that are

presented in the **Field** list are stored in the details.returns database table. This enables you to construct filters based on a wide range of device data.

For more information on the details.returns table, see the *IBM Tivoli Network Manager IP Edition Discovery Guide*.

c. Select **not like** from the **Comparator** list.

d. Type `10\.30\..*\.([6-9]$)|[1-9][0-9].*$` in the **Value** field.

The prediscovery filter allows the use of regular expressions. The regular expression that you just typed in instructs Network Manager to exclude all the unwanted devices in the 10.30.0.0 range, namely, 10.30.*.6-9 and 10.30.*.10-255. Note that you do not need to enclose the operand in single quotes. The system will do this automatically when it constructs the SQL where clause for this filter.

Note: If the filter had been formulated using the more obvious **like** comparator, like this: `m_UniqueAddress LIKE '10\.30\..*\.[1-5]$` then no devices with management addresses outside of 10.30.*.1-5 would end up in the topology. This would therefore exclude the other scope zones that we formulated earlier, and would require the creation of extra filter rows in the filter to pass through those scopes. This example shows that it is important to design the filter logic so that you do not need to modify the prediscovery filter every time you add new scopes.

For information on basic regular expression syntax, see the *IBM Tivoli Network Manager IP Edition Language Reference*

12. Click **Save** to save the filter. The filter you defined appears in the filter list on the left of the window.
13. Click **Close**. The filter you defined appears in the **Available Filters** list within the **Pre-Discovery Filter** section of the panel.
14. Select the filter and click **>**. The filter is applied and appears in the **Selected Filters** list.
15. Click **Save**  to save your discovery configuration settings.

You have now configured your managed regions to include the first five addresses only in a set of subnets 10.30.*.1-5. You did this by creating a prediscovery filter, which uses regular expressions to filter out unwanted devices from the discovery. You created the prediscovery filter based on data collected early in the discovery by the Details agent.

The next step is to ensure that the *feedback* mechanism is switched on. Feedback is the mechanism by which data returned by discovery agents is used to find other devices. Examples of feedback data include the IP addresses of remote neighbors, or the subnet within which a local neighbor exists.

Enabling the Ping finder and the feedback mechanism

You can discover neighboring connected devices using the feedback mechanism. Feedback enables the discovery process to learn about the existence of devices as a result of querying other devices. In order for feedback to work, the Ping finder must be enabled.

By default, the Ping finder and the feedback mechanism are enabled. In this task you are going to locate these settings in the Discovery Configuration GUI and check that the settings are enabled.

1. Click **Seed**.

2. Check that **Use Ping Finder in Discovery** is checked. The Ping finder is used to enable discovery to ping devices to verify existence and is used at various points in the discovery, including the following:
   - At the early stages to discover devices in the ping seed list and to perform ping sweeps of subnets.
   - Throughout the discovery to verify remote neighbors discovered as part of the feedback mechanism.
   - To identify the active interfaces on any subnets found.
3. Click **Advanced**. The **Advanced** tab contains a large number of configuration settings that the advanced user can set to work around unconventional network behavior.
4. In the **Advanced Discovery Configuration** section of the panel, check the setting for **Enable Feedback Control**.

   By default, feedback is set to **Feedback only on Full**, and this is the desired setting. This setting ensures that feedback is active when you are performing a *full discovery*, a discovery of your entire network. The entire network is made up of the managed areas that you defined using your discovery scopes.

   The other settings for **Enable Feedback Control** are as follows:

   No feedback
   This setting is useful when you want to strictly limit discovery to a predefined list of IP addresses, and you do not want the discovery process to discover any connected devices that are not in the predefined list. You will learn more about this setting when you set up your production discovery configuration.

   Feedback
   This setting switches on feedback for both full discovery and *partial discovery*. Partial discovery is a discovery of only a part of your network such as one or more devices or subnets. Partial discovery is usually run in response to individual device changes: it can be run on demand, or scheduled for particularly volatile sections of the network. The main purpose of partial discovery is to quickly update topology data for a given device or devices and device connectivity is usually less of a priority when performing partial discovery. Device connectivity can be updated when the next scheduled full discovery is performed.

   Typically you do not need to change any settings on the **Advanced** tab. However, it is worth noting the following settings:

   Enable Ping Verification
   This setting forces discovery to create network objects only for devices that respond to a ping.

   Enable ifName/ifDescr Interface Naming
   Changes the default naming convention for discovered interfaces. If you change the default naming convention for discovered interfaces, you must change the BuildInterfaceName stitcher to specify your naming convention.

   Discovery stitchers are pieces of code written in Network Manager's proprietary *stitcher language* that perform two main tasks within the discovery process:

   Data collection stitchers
   These stitchers move data collected from network devices from

one database to another. These are system stitchers and are not available for customization by users.

Data processing stitchers
These stitchers 'stitch' together the data gathered by the discovery agents to generate the network topology, which can then be visualized and polled. Advanced users can modify these stitchers so that the discovery process will produce a custom topology. For example, as stated above, you can modify the BuildInterfaceName stitcher to specify a custom interface naming convention.

Enable VLAN modelling
If you do not need to model VLANs, then disable this option to speed up discovery.

5. Click Save  to save your discovery configuration settings.

You have now confirmed that the feedback and Ping finder are enabled.

The Ping finder is used to enable discovery to ping devices to verify existence and is used at various points in the discovery.

Feedback enables the discovery process to learn about the existence of devices as a result of querying other devices. In order for feedback to work, the Ping finder must be enabled.

Configuring device and subnet access using SNMP community strings

To enable discovery agents to access your network devices to retrieve SNMP data, you must specify SNMP community strings for the subnets and IP addresses in your network.

Community strings and Telnet access data can be *global*, which means that the discovery tries the community string for every device it encounters, or restricted to specific subnets (that is, used only on devices within a specific subnet), or even restricted to specific devices. Specifying community strings and Telnet access data by subnet results in a more efficient and faster discovery. In general, the more specific the credentials, the faster the discovery will determine the correct credentials.

Note: Speed of discovery related to community string settings in the GUI only affects the initial discoveries. Once Network Manager has identified the correct community strings, it stores this information in the NCMONITOR relational database. Subsequent discoveries access this database for SNMP community strings and other SNMP-related device access information.

When the discovery processes community strings, it always attempts to use the most specific match first. So if the discovery was attempting to query the device, 10.40.1.13, it first tries to use any community strings for that specific IP address (10.40.1.13/32). If none is available for that address, it tries to use any subnet strings specified for the Class C subnet 10.40.1.0/24 and then for the Class B subnet 10.40.0.0/16. If no subnet-specific IP addresses exists, it defaults to the global addresses in the priority order in which they are specified in the GUI.

Network Manager provides a global community string of `public`. Let's suppose that you want to keep the default `public` community string as some of your

devices might use that string, but you also want to create a global community string specific to your network of `acme`, and that you want to give this `acme` community string higher priority than the default `public` community string.

In addition, you also want to define the following specific community strings:

- Class C subnet 10.40.1.0/24: give this subnet a community string of `network2`
- Class B subnet 10.40.0.0/16: give this subnet a community string of `network1`

You want the discovery process to use SNMPv2 community strings but you also want a fallback to the equivalent SNMPv1 community string if the SNMPv2 community string does not work for a particular device. To do this, you will need to create duplicate entries for each community string, one for SNMPv2 and one for SNMPv1. You must also ensure that each SNMPv2 community string is placed in the GUI so that it has a higher priority over the corresponding SNMPv1 community string.

This means that the order in which community strings will be tried for a device in the Class C subnet 10.40.1.0/24 (assuming that only the final community string works for that device will be as follows):

1. There is no community string at the device level so try the most specific subnet community string.
2. Class C subnet 10.40.1.0/24 community string `network2` using SNMPv2
3. Class C subnet 10.40.1.0/24 community string `network2` using SNMPv1
4. Class B subnet 10.40.0.0/16 community string `network1` using SNMPv2
5. Class B subnet 10.40.0.0/16 community string `network1` using SNMPv1
6. Custom global community string `acme` using SNMPv2
7. Custom global community string `acme` using SNMPv1
8. Network Manager global community string `public` using SNMPv2
9. Network Manager global community string `public` using SNMPv1

The next step is to enter these community strings into the GUI to ensure that they are tried in this order.

1. Click **Passwords**. In the SNMP Community Strings section of the panel, you should see two rows in the table, showing the following information:

Table 1. SNMP community strings

#	IP/Subnet	Community String	SNMP Version
1	null	public	Version 2
2	null	public	Version 1

   This information indicates that by default you already have two public community strings defined, with priority order being given to the SNMPv2 of this community string. Priority order is set by placing the community string higher in the table.
2. Add the global community string `acme` for SNMPv2 and SNMPv1.
   a. To add a new SNMP community string, click **New**  . The SNMP Password Properties page is displayed.
   b. Type `acme` in the **Name** field.
   c. The **Apply To** default setting is **All Devices**. Leave this setting as is. This ensures that the `acme` community string is global.
   d. For **SNMP Version** click **V2**.

e. Click **OK** to accept the settings.

f. Now add a second custom global community string, with the following settings:

Name Type `acme`

Apply To
All Devices

SNMP Version
Click **V1**

Click **OK**. At this point you have added two custom global community strings. These appear below the default public community strings in the table.

Table 2. SNMP community strings

#	IP/Subnet	Community String	SNMP Version
1	null	public	Version 2
2	null	public	Version 1
3	null	acme	Version 2
4	null	acme	Version 1

3. Assign the `acme` community string higher priority than the default `public` community string.

a. In row 3, which contains the acme community string for SNMPv2, click **Move Up** twice to move the acme SNMPv2 to the top of the table.

b. In row 4, which contains the acme community string for SNMPv1, click **Move Up** twice to move the acme SNMPv1 to the second row in the table. The table should now appear as follows. The acme community strings appear in the first two rows in the table, and this means that the discovery process will try these strings first when attempting to gain access to devices. Also, the SNMPv2 version of acme will be tried before the SNMPv1 version.

Table 3. SNMP community strings

#	IP/Subnet	Community String	SNMP Version
1	null	acme	Version 2
2	null	acme	Version 1
3	null	public	Version 2
4	null	public	Version 1

The next step is to add the subnet-specific community strings.

4. Add community strings for the Class C subnet 10.40.1.0/24.

a. Click **New** . The SNMP Password Properties page is displayed.

b. Add a subnet-specific community string with the following settings:

Name Type `network2`

Apply To
Click ***Subnet** and type `10.40.1.0` in the ***Subnet** field. In the adjacent field following the slash sign, /, type the subnet mask `24`.

SNMP Version
Click **V2**

Click **OK**.

c. Click **New**  . The SNMP Password Properties page is displayed.

d. Add a second subnet-specific community string with the following settings:

Name Type `network2`

Apply To
Click ***Subnet** and type `10.40.1.0` in the ***Subnet** field. In the adjacent field following the slash sign, **/**, type the subnet mask `24`.

SNMP Version
Click **V1**

Click **OK**.

5. Add community strings for the Class B subnet 10.40.0.0/16.

a. Click **New**  . The SNMP Password Properties page is displayed.

b. Add a third subnet-specific community string with the following settings:

Name Type `network1`

Apply To
Click ***Subnet** and type `10.40.0.0` in the ***Subnet** field. In the adjacent field following the slash sign, **/**, type the subnet mask `16`.

SNMP Version
Click **V2**

Click **OK**.

c. Click **New**  . The SNMP Password Properties page is displayed.

d. Add a fourth subnet-specific community string with the following settings:

Name Type `network1`

Apply To
Click ***Subnet** and type `10.40.0.0` in the ***Subnet** field. In the adjacent field following the slash sign, **/**, type the subnet mask `16`.

SNMP Version
Click **V1**

Click **OK**. The **SNMP Community String** table should now appear as follows.

Table 4. **SNMP Community String** *table following configuration and prioritization of the community strings*

#	IP/Subnet	Community String	SNMP Version
1	null	acme	Version 2
2	null	acme	Version 1
3	null	public	Version 2
4	null	public	Version 1
5	10.40.1.0	network2	Version 2

Table 4. **SNMP Community String** *table following configuration and prioritization of the community strings (continued)*

#	IP/Subnet	Community String	SNMP Version
6	10.40.1.0	network2	Version 1
7	10.40.0.0	network1	Version 2
8	10.40.0.0	network1	Version 1

The subnet-specific community strings appear below the global community strings in the table. In this case the order does not matter, as the discovery process always tries specific community strings before global community strings, Also, the more specific the community string, the greater the priority, so the Class C community strings will always be tried before the Class B strings, regardless of where you place them in the table.

You have now configured SNMP community strings to enable access to all the devices in your managed regions.

You have configured subnet-specific community strings for two of the subnets in your managed regions. These strings have priority.

You have also defined global community strings, which apply to all devices.

Ensuring all network technologies are covered

Network Manager discovery agents retrieve information from devices in your network. Dedicated agents such as CiscoFrameRelay retrieve specific network technology data. Check the list of full discovery agents to ensure that all the technologies in your network are covered.

In addition to the Details discovery agent, which retrieves basic information from all network devices, the discovery process also uses a set of protocol and technology-specific discovery agents to retrieve more detailed device information. By default a subset of discovery agents is enabled, and this subset usually satisfies the needs of most initial discoveries. You can use the GUI to retrieve a description of each discovery agent, and optionally enable extra discovery agents.

1. Click **Full Discovery Agents**.
2. In the **Agents** tree, click **Agents** > **Full Layer 2 and Layer 3 Discovery** and then click the subnodes to view the full set of enabled layer 2 and layer 3 discovery agents.
3. Click an agent to see its description.

   For example, click **Agents** > **Full Layer 2 and Layer 3 Discovery** > **Layer 3** > **IpRoutingTable** to see a description of the IpRoutingTable agent. This layer 3 agent is enabled by default and learns other IP addresses and subnets to feed back into the discovery by examining each router's routing table.

   Other agents are disabled by default. For example, click **Agents** > **Entity** to see a description of the Entity agent. This agent discovers optional detailed containment information for a network entity. This agent only needs to be enabled if you want to model physical containment and perform asset management, because the agent is resource intensive and lengthens discovery time.

4. If you made any changes to the discovery configuration, then click Save  to save your changes.

You have reviewed the network protocols and technologies that will be discovered. You did this by reviewing the set of full discovery agents that are enabled. You have also seen how to use the agent tree to locate agents, how retrieve more information on each agent, and how to enable an agent.

Configuring initial discovery settings: Summary

While working through this information, you configured initial discovery settings. You will use these settings to run a discovery.

By working through this information, you learned the following concepts and skills:

- Use of the Discovery Configuration GUI to configure key discovery settings
- Use of scope zones to include and exclude regions of your network
- How to use prediscovery filters to restrict discovery to more complex IP address ranges
- Use of ping sweeping to identify all devices in your managed regions
- How to activate the Ping finder and the feedback mechanism to enable ping sweeping
- How to configure device and subnet access using SNMP community strings
- Use of discovery agents to ensure all network technologies are covered

Launch the discovery and monitor discovery progress

Use the Discovery Status GUI to launch the discovery and to monitor discovery progress.

As you work through this information, you will perform the following discovery tasks:

- "Launching discovery" on page 22
- "Monitoring overall discovery progress" on page 23
- "Monitoring Ping finder status" on page 24
- "Monitoring discovery agent status" on page 25
- "Troubleshooting discovery issues" on page 28

Understanding discovery phases

A discovery runs through distinct phases. Use this information to understand each of the phases and to help you to monitor the discovery.

A running discovery passes through the following phases. It is possible for phases to overlap; for example, the Resolving Addresses phase (Phase 2) can start before the Interrogating Devices phase (Phase 1) has completed.

Interrogating Devices

In earlier versions of Network Manager, this phase was known as Phase 1. During this phase the discovery has found the first seed device and is finding other devices within scope. As devices are found, discovery agents proceed to interrogate the devices and retrieve device details, associated device addresses, and device connectivity.

Resolving Addresses

In earlier versions of Network Manager, this phase was known as Phase 2. During this phase the discovery maps devices in layers two and three of the OSI model.

Downloading Connections
In earlier versions of Network Manager, this phase was known as Phase 3. During this phase the discovery uses information retrieved from network switches to discover and verify device connectivity.

Correlating Connectivity
In earlier versions of Network Manager, this phase was known as Phase -1. During this phase the discovery process builds the network topology using network device information collected during the earlier phases. The work of building the topology is performed by discovery stitchers.

Launching discovery

Now that you have configured your discovery settings, the next step is to manually start your initial discoveries using the Discovery Status GUI.

You can launch a discovery in any of the following ways:

- Manually launching a discovery.
- Scheduling discovery to launch automatically on a regular basis.

Once you are satisfied with the results of your discovery, then you will probably want to schedule regular discoveries. For the moment, we are still running initial discoveries and tuning the discovery configuration based on the results, so the next step is to launch discovery manually.

1. Click **Discovery** > **Network Discovery Status**. This takes you to the Discovery Status GUI, where you can launch your discovery and monitor progress. The **Monitoring** section of the Discovery Status GUI displays a table with the four discovery phases:
   - Interrogating Devices
   - Resolving Addresses
   - Downloading Connections
   - Correlating Connections

   The **Discovery Type** label at the top right informs you that discovery is not running.

   Note: If you access this GUI while a discovery is running, for example, a regularly scheduled discovery, even if you did not start the discovery yourself, you would be able to monitor the running discovery.
2. From the **Domain** list, select the required domain.
3. Click **Start Discovery** 
4. Check that discovery starts okay. To do this, check the following:
   - The **Discovery Type** label shows the text **Discovery starting** and displays the running icon  .
   - The **Last status received** label shows the text **Discovery starting**.
   - After a short time, the **Interrogating Devices** phase displays  running status and the **Elapsed Time Current** column shows a time counter for this phase.

You have now manually launched discovery. The next step is to monitor discovery progress.

Monitoring overall discovery progress

As discovery progresses, use the Discovery Status GUI to monitor discovery, to provide detailed information on the progress of discovery agents, and to view details of the last discovery.

You have now launched the discovery and the first phase, Interrogating Devices, has started. You can monitor the overall progress of the discovery by viewing the progress of each of the discovery phases in the Discovery Status GUI. You can sort the columns in this screen to make the viewing of data easier.

1. Check that the **Interrogating Devices** phase is progressing without errors. As the phase progresses, you should see the following:
   - **Status** is **Running** .
   - Under the **Elapsed Time** column, the **Current** value increases at regular intervals.
   - Under the **Work Completed** column, the **Current** value shows the number of IP addresses that have been found so far by the Ping finder. Based on the managed areas configured earlier as scope zones, the expectation is that the **Current Work Completed** value for **Interrogating Devices** shows the actual number of IP addresses found during the discovery. For our discovery configuration, the value will be anything from 16,000 to 65,000 IP addresses. This is based on the following estimates of the number of IP addresses in each managed area:
     - Single IP address, 10.30.1.20/32: number of IP addresses is 1.
     - The entire Class B subnet, 10.40.0.0/16, except for the Class C subnet 10.40.2.0/24. Maximum number of IP addresses is 65,536 - 255, which is just over 65,000. Assuming a sparsely populated Class B subnet, a minimum number of IP addresses might be 15,000.
     - The first five IP addresses only in a set of subnets, 10.30.*.1-5. Maximum number of IP addresses is 255 x 5 = 1275.

   You can compare the values of **Elapsed Time** and **Work Completed** in this discovery to the values in the previous discovery and this provides an extra level of verification that the discovery is running OK.
2. As the discovery moves from the **Interrogating Devices** phase into the subsequent phases, monitor the **Current Work Completed** for an idea of how far the phase has completed. In the **Resolving Addresses** and **Downloading Connections** phases, the percentage of work completed is calculated based on the number of IP addresses each agent has completed processing divided by the number of IP addresses the agents still have to process. Use this figure to obtain an idea of how close the phase is to completion.

You now have an idea of how the overall discovery progress is progressing on a phase by phase basis. During the Interrogating Devices phase, the Ping finder pings each device within the configured scope zones The next step is to monitor the progress of the Ping finder as it pings the various IP addresses and subnets within each scope zone.

Monitoring Ping finder status

During the Interrogating Devices phase, the Ping finder pings each device within the configured scope zones. You can use the Discovery Status GUI to track the progress of the Ping finder through the subnets of each scope zone.

In the Discovery Status GUI, click **Ping Finder Status**. The **Ping Finder Status** section of the GUI shows the **Ping Finder Status** table. This table lists all the subnets that make up the scope zones that you configured earlier.
The **Ping Finder Status** table contains the following information:

Address
A list of IPs and subnets discovered to this point.

Netmask
For each subnet, this column indicates the netmask value.

Last Pinged
The last IP address pinged in this subnet.

Status Indicates whether the Ping finder is still pinging this device or subnet or whether it has completed pinging.

For example, based on the scope zones that you configured, the **Ping Finder Status** table might look something like this:

Table 5. Example of data in **Ping Finder Status** *table*

Address	Netmask	Last Pinged	Status
10.30.1.20	-	-	
10.30.0.0	255.255.0.0	10.30.255.5	
10.40.0.0	255.255.0.0	10.40.39.3	

This example shows that the pinging of your configured scope zones is proceeding. In particular:

- The single IP address scope zone 10.30.1.20 has been successfully pinged.
- All of the routers in the subnet 10.30.0.0 have been successfully pinged. The managed area here was made up of a complex IP address range, and to define this range you had to configure a prediscovery filter to exclude all IP addresses in the Class B subnet 10.30.0.0 outside of the range defined by 10.30.*.1-5. As you can see from the table, the last IP address to be pinged was 10.30.255.5, which is the very last IP address in this complex range.
- The Class B subnet 10.40.0.0 is still in the process of being pinged. Class B subnets can take up to two to three hours to ping sweep, because the discovery process has to try pinging every single possible IP address in the subnet. For a Class B subnet, that is 65,536 attempted IP address pings. As you can see from the table, the discovery has just pinged the IP address 10.40.39.3, so it still has a lot of pinging to do.

Note: If the Class B subnet 10.40.0.0 is a sparsely populated subnet then, when the Interrogating Devices phase completes, the Ping finder might not yet have completed pinging the subnet. By default, if, following the end of Phase 1, the Interrogating Devices phase, 90 seconds passes without the Ping finder finding any more devices, then the discovery enters what is known as the *blackout state*. During the blackout state the rest of the discovery phases progress normally but the Ping finder continues to ping sweep the sparsely populated Class B subnet 10.40.0.0,

and any new IP addresses that are found are held in a special database table until the discovery phases complete for the IP addresses discovered up to the moment when the blackout state began. Once those discovery phases are complete, then the discovery process resumes for these addresses discovered during the blackout state.

You have now spent some time monitoring Ping finder status and you have concluded that pinging of subnets is progressing satisfactorily. Once a device has been pinged by the Ping finder and its existence has therefore been verified, the discovery process passes the device details to the discovery agents for information retrieval from the device. The next step is therefore to monitor the progress of the discovery agents.

Monitoring discovery agent status

As the discovery progresses, different discovery agents are called to retrieve device and connectivity data from discovered devices. This data will later be used to build the network topology. You can use the Discovery Status GUI to track overall and detailed status of the discovery agents.

Discovery agents run in the following order. Let's assume that the Ping finder has just verified the existence of the device 10.40.230.1, in our Class B subnet.

- After the Ping finder has verified the existence of IP address 10.40.230.1, the Details agent is called to retrieve basic information from this device.
- Once the Details agent has retrieved information from the IP address 10.40.230.1, the AssocAddress agent is called to retrieve all the IP addresses associated with 10.40.230.1. If the associated IP addresses have not yet been discovered and are in scope, then the IP address is passed to the Ping finder so that the device existence can be verified.
- Meanwhile other agents are called to interrogate the IP address 10.40.230.1. For example, this IP address is a router, so the IpRoutingTable agent is called to retrieve information from the routing table of router 10.40.230.1 and to feed back connected devices to the Ping finder. More precisely, 10.40.230.1 is a Cisco router within a BGP network, so the CiscoBGPTelnet agent is called to retrieve BGP-related data from the device.

This process is repeated for all devices until all relevant data has been retrieved from the devices in scope.

Note: The Details agent and the AssocAddress agent are the only discovery agents that interrogate every device in scope. The IP address counts for these agents are therefore always higher than those of the other agents.

As you monitor discovery agent progress, some of the key questions to ask are the following?

- Are all agents running okay? Have any agents crashed? If so, which IP address might have caused the agent to crash.
- Which agents are taking a long time to complete and which IP address appears to be causing this delay?
- Which agents are holding up a discovery phase and preventing the phase from completing? For example, which agents are holding up the Interrogating Devices phase?
- How many IP addresses does a particular agent, for example, the Details agent, still have to work on? How far is the agent through its work?
- Which IP addresses were found late in the discovery?

In this task we will use the Discovery Status GUI **Agents Status** table to answer these questions.

1. In the Discovery Status GUI, click **Agents Status**. The **Agents Status** section of the GUI shows the discovery agents that are currently running and provides status information on each agent. Agents are sorted in order of state, and for each state, are sorted alphabetically.
2. Check that all agents are running okay.
   a. Check the **State** column of the Agents Status table. If any of the agents have terminated unexpectedly, then that agent will appear at the top of the list. The default sort order is descending order of state. Each agent state is assigned a number as follows:

Table 6. Agent states

State	Value	Icon	Description
Died	5		The agent has terminated unexpectedly. This is a potential discovery problem.
Finished	4		The agent is still running but has finished processing of all the IP addresses in its queue. The agent is still available to process any further agents placed in the queue.
Running	3		The agent is currently processing IP addresses.
Starting	2		The agent is starting up.
Not running	1		The Agent is not running.

   Let's assume that the CiscoSwitchTelnet agent has terminated unexpectedly.
   b. Click the CiscoSwitchTelnet agent cell in the **Agents Status** table. The **IP Address Status** table, which is the table below the **Agents Status** table, now displays all IP addresses for this agent.
   c. Set the radio button above the table to **All**. This ensures that the table shows all IP addresses for this agent, including IP addresses that have been processed by this agent, that are currently being processed, and that are queued for processing by the agent.
   d. Sort the **IP Address Status** by **Return Time**. Table rows with empty **Return Time** cells move to the top of the table. Look for IP addresses where the row has a value in the **Despatch Time** cell but the **Return Time** cell is empty. These IP addresses might have caused the agent to terminate unexpectedly.

      Note: Further investigation is required to determine why this IP address caused the agent to crash.
3. Focus on an agent that is taking a long time to complete and to determine which IP address might be causing this delay. Let's assume that the IpForwardingTable agent is taking a long time to complete.
   a. Click the IpForwardingTable agent cell in the **Agents Status** table. The **IP Address Status** table, which is the table below the **Agents Status** table, now displays all IP addresses for this agent.
   b. Set the radio button above the table to **All**. This ensures that the table shows IP addresses that have been processed by this agent and IP addresses that are queued for processing by the agent.
   c. Sort the **IP Address Status** by **Return Time**. Table rows with empty **Return Time** cells move to the top of the table. Look for IP addresses where the

row has a value in the **Despatch Time** cell but the **Return Time** cell is empty. These IP are still being worked on by the agent and might be causing the agent to take a long time to complete.

4. Determine which agents are holding up a discovery phase; for example, which agents are holding up the Interrogating Devices phase.
   a. From the **Filter Agents by Phase** list just above the **Agents Status** table, select **Interrogating Devices**. The **Agents Status** table now shows only the agents that complete in the Interrogating Devices phase.
   b. Sort the **Agents Status** table by ascending order of **State**. This brings the running agents to the top of the table, and enables you to see which agents are still running.
   c. Sort the **Agents Status** table by descending order of state of **Outstanding IP Addresses**. This brings the agents that are still processing IP addresses to the top of the table, with the agents that have the most IP addresses to process at the very top.
5. Determine how far an agent, let's say the Details agent, is through its work.
   a. Sort the **Agents Status** table by alphabetical order of **Agent**.
   b. Find the Details agent.
   c. In the Details agent row record the values for **Outstanding IP Addresses** and **Total IP Addresses**. You can determine a percentage of work complete using the following formula.

      Percentage work complete for an agent = (Outstanding IP Addresses / Total IP Addresses) * 100
6. Determine which IP addresses were found late in the discovery.
   a. Sort the **Agents Status** table by alphabetical order of **Agent**.
   b. Find the Details agent.
   c. Click the Details agent cell in the **Agents Status** table. The **IP Address Status** table now displays all IP addresses queued for the Details agent. These are the IP addresses that are in the agents despatch queue; however the agent has not yet started work on these devices.
   d. Sort the **IP Address Status** table by descending order of **Despatch Time**. In general, the later the agent was found during the discovery, the later its despatch time to the Details agent. This means that the agents that now appear at the top of the **IP Address Status** table were found latest during the discovery.

You have now used the **Agents Status** and **IP Address Status** tables within the Discovery Status GUI to monitor the status of discovery agents. In particular, you have used the tables to find answers to the following queries:

- Are all agents running okay? Have any agents crashed? If so, which IP address might have caused the agent to crash.
- Which agents are taking a long time to complete and which IP address appears to be causing this delay?
- Which agents are holding up a discovery phase and preventing the phase from completing? For example, which agents are holding up the Interrogating Devices phase?
- How many IP addresses does a particular agent, for example, the Details agent, still have to work on? How far is the agent through its work?
- Which IP addresses were found late in the discovery?

Troubleshooting discovery issues

Discovery issues might arise for a number of reasons often due to malformed or inconsistent data in the network causing discovery agents to hang or crash. Before calling IBM Support, you can run a series of checks to try to narrow down the problem.

General discovery troubleshooting checks include the following:

- Check for any rogue processes running on UNIX servers.
- Check that you have sufficient memory for the Discovery process.
- Check for any discovery core files.

1. On UNIX systems check for any rogue processes running on the server.
   a. Stop all Network Manager processes. Use the following command:`itnm_stop ncp`
   b. Check for any Network Manager processes processes that have not stopped. Use the following command:`ps -ef | grep ncp`
   c. Kill any of these rogue ncp process using the Unix `kill` command.
   d. Restart Network Manager. Use the following command:`itnm_start ncp`
   e. Launch discovery.
2. Check that you have sufficient memory for the Discovery process.
3. Check for any discovery core files.
   a. Issue the following command to recursively list out all directories that contain core files: `ls -lR $NCHOME/precision/PD/cores/`.
   b. Call IBM Support for help with debugging these core files.

At this point you have run a series of checks to try to narrow down any possible discovery problems.

This task has covered the following activities and concepts:

- Checking for rogue processes running on the server.
- Checking that you have sufficient memory for the Discovery process.
- Checking for any discovery core files.

Launch the discovery and monitor discovery progress: Summary

While working through this information, you launched and monitored key elements of a discovery, including discovery phases, agents, and the Ping finder.

By working through this information, you learned the following concepts and skills:

- Discovery phases and what occurs in each phase
- How to use the Discovery Status GUI to launch and monitor discoveries
- How to compare the current discovery with the previous discovery
- How to determine which subnets and IP addresses the Ping finder is currently processing
- Why a discovery might enter the blackout state and consequently require more than one discovery cycle
- The order in which discovery agents and what information each discovery agent gathers
- How to interpret the Agents Status table in the Discovery Status GUI in order to answer key questions about the progress of the discovery
- General information about discovery troubleshoting

Verifying the topology

Use reports and topology views to check how well the discovery has modeled your network.

As you work through this information, you will perform the following discovery tasks:

- "Checking device access"
- "Checking for unclassified devices" on page 31
- "Checking connectivity" on page 35
- "Checking for unmanaged interfaces" on page 35

Checking device access

Use reports and SQL queries to check that all devices responded to SNMP requests during the discovery.

Let's suppose that you want to check that the discovery was able to access all the devices in your configured scope zones. You can run the Devices with no SNMP Access report. This report lists the devices that were found but for some reason the discovery was unable to access the device using SNMP.

Once you have identified the devices that the discovery was unable to access using SNMP, you can begin to investigate why SNMP access failed. Reasons why SNMP access might fail include the following:

Device not reachable
A firewall configuration might be blocking SNMP access. Make sure that SNMP access to the device is possible across your network's firewalls.

Device not responding
When Network Manager issues a request to a device, if the device does not respond, the request times out after a configurable time-out period and number of retries. Reasons for the timeout might include any of the following:

- If any of the devices in the configured scope zones is down at the time of discovery, then the device will not be found by the Ping finder and will not appear in the discovered network topology. You will also not be able to see these devices in the Devices with no SNMP Access report.
- The SNMP agent on the device is not running. Check the device and make sure that the SNMP agent is running.
- The SNMP agent is using a non-standard port. SNMP uses UDP protocol to communicate with the agents normally on port 161. You might need to reconfigure this on the device.
- The SNMP agent on the device is configured with a different community string than the one that you specified in Network Manager.
- If access control lists (ACLs) are being used for SNMP security, then check that the management device is on the SNMP agent ACL.

Let's suppose that you have narrowed down the reason for SNMP access failure to the last item in the list: the SNMP agent on the device is configured with a different community string than the one that specified in Network Manager. In this task you will learn how to fix this.

1. Click **Reporting** > **Common Reporting** > **Network Manager**. Then click **Troubleshooting Reports**.

2. Select the **Devices with no SNMP Access** report. The report displays a list of devices (the entity name and the IP address for each device) that the discovery found but the discovery agents were unable to access using SNMP.

   For more information on the **Devices with no SNMP Access** and other discovery troubleshooting reports, see the *IBM Tivoli Network Manager IP Edition Administration Guide*.

3. To check which community string Network Manager used during discovery for a given IP address, run the following SQL query on the NCMONITOR database to retrieve the SNMP access attributes for that device.

   ```
   SELECT v1.community, sa.version, sa.timeout, sa.retries
   FROM ncmonitor.snmpTarget st
   INNER JOIN ncmonitor.snmpV1Sec v1 on v1.accessid = st.readaccessid
   INNER JOIN ncmonitor.snmpAccess sa on sa.accessId = v1.accessid
   WHERE st.netaddr = 'ip_address';
   ```

   Where *ip_address* is one of the IP addresses listed in the **Devices with no SNMP Access** report.

   This query retrieves the following information:

   community
   Community string to use.

   version
   SNMP version to be used. Possible value are:
   - 0: SNMPv1
   - 1: SNMPv2
   - 3: SNMPv3

   timeout
   Number of milliseconds before retrying the SNMP request .

   retries Number of retries before giving up.

   For more information on the NCMONITOR database, see the *IBM Tivoli Network Manager IP Edition Event Management Guide*.

4. If the community string is encrypted, you can decrypt it using the `ncp_crypt` utility. For example, the following command-line decrypts an encrypted password.

   ```
   ncp_crypt -password @44:G5IhL1i2obPcXDu6uiMcse+U0qdRPojK0o6erxrfk/Y=@ -decrypt
   ncp_crypt ( IBM Tivoli Network Manager Password Encryption/Decryption Tool )
   Copyright (C) 1997 - 2008 By IBM Corporation. All Rights Reserved.
   See product license for details.

   IBM Tivoli Network Manager Version 3.9 created by fblucher at
   03:38:26 Wed Nov 19 GMT 2010

   @44:G5IhL1i2obPcXDu6uiMcse+U0qdRPojK0o6erxrfk/Y=@ public
   ```

   For more information on the `ncp_crypt` utility, see the *IBM Tivoli Network Manager IP Edition Administration Guide*.

5. Compare the SNMP community string stored in the NCMONITOR database for each device with the actual SNMP community for that device.
6. Correct the SNMP community string information for each device in the Discovery Configuration GUI.

You have now checked that discovery was able to access devices and have corrected the SNMP community strings for any devices which could not be accessed. The next step is to check for any unclassified devices, that is, devices that do not have a device type classification within Network Manager.

Related tasks:

"Configuring device and subnet access using SNMP community strings" on page 16

To enable discovery agents to access your network devices to retrieve SNMP data, you must specify SNMP community strings for the subnets and IP addresses in your network.

Checking for unclassified devices

Use reports to check that all devices found have a device type classification within Network Manager. Device type classifications are based on the MIB variable sysObjectId retrieved from the device during discovery.

Let's suppose that you want to check whether the discovery encountered any unclassified devices, that is, devices that do not have a device type classification within Network Manager. You can run the following reports to determine this:

- Devices with Unclassified SNMP Object IDs report
- Devices with Unknown SNMP Object IDs report

For any unclassified devices, you can take the following actions:

- Contact IBM Support with a list of these device types. IBM issues new device support several times a year and the latest Network Manager FixPack might include these device types. If not submit them for inclusion in a future FixPack.
- In the meantime add the sysObjectId information and mappings to Network Manager so that future discoveries are able to classify the devices. By doing this you will also enable the device to be correctly visualized in topology maps. The device class will also be available for inclusion in poll policies.

Note: The Network Manager team updates device support throughout the year. Contact IBM Support to find out when your unclassified devices will be added. In the meantime, this task describes how you can configure new device classifications.

In this task you will learn how to add the sysObjectId information and mappings to Network Manager.

1. Click **Reporting** > **Common Reporting** > **Network Manager**.
2. Click **Troubleshooting Reports**.
3. Select the **Devices with Unclassified SNMP Object IDs** report. The report displays a list of devices with sysObjectId values that are unrecognized by Network Manager. The data in the report is grouped by sysObjectId. Let's assume you see data in the report similar to the following, under the sysObjectId 1.3.6.1.4.1977.1.6.1279.1.

Table 7. Unclassified device data for the sysObjectId 1.3.6.1.4.1977.1.6.1279.1

Entity Name	IP Address	System Description	CLASSNAME
group-1-b2.class.example.org	10.40.15.113	Hardware: x86 Family 15 Model 2 Stepping 8 AT/AT COMPATIBLE - Software; Microsoft Windows 2000 Version 5.0 (Build 2195 Uniprocessor Free)	NetworkDevice

The report shows that the device with IP address 10.40.15.113 has the generic classification of NetworkDevice. The device has been assigned this generic

classification because the system does not recognize the sysObjectId. Network Manager uses a class hierarchy to model and organize network devices. The *Network Device* class is a superclass for all device types, and contains a hierarchy of subclasses such as Cisco and Juniper that group network devices by manufacturer and then by device type and model.

For more information on the class hierarchy, see the *IBM Tivoli Network Manager IP Edition Language Reference*.

In order to correctly classify the device with IP address 10.40.15.113, the first step is to determine the manufacturer of the device. You can determine the device manufacturer by identifying the manufacturer associated with the sysObjectId. The sysObjectId is an SNMP MIB variable, and includes information within it that identifies the device manufacturer.

4. Click **Availability** > **Network Availability** > **SNMP MIB Browser** to open the SNMP MIB Browser. The SNMP MIB Browser enables you to browse the SNMP MIB tree. Each SNMP MIB variable, such as the sysObjectId, corresponds to a node on the tree. The SNMP MIB Browser opens with the MIB tree in the top left panel. By default, the MIB tree is open to the **iso** > **org** > **dod** > **internet** node.
5. Click the **internet** node in the MIB tree. The **MIB Variable Information** panel at the bottom left now displays the OID value for the internet node as **1.3.6.1**. You are now going to "walk" the MIB tree until you get to the sysObjectId value of 1.3.6.1.4.1.1977, which is the sysObjectId that contains the manufacturer of the unclassified device from the **Devices with Unclassified SNMP Object IDs** report.
6. Click the **private** node in the MIB tree. The **MIB Variable Information** panel at the bottom left now displays the OID value for the internet node as **1.3.6.1.4**. The number 4 at the end of this sysObjectId value refers to the fourth subnode (**private**) within the **internet** node.
7. Expand the **private** node. The **private** node expands to display a single **enterprises** node.
8. Click the **enterprises** node. The **MIB Variable Information** panel at the bottom left now displays the OID value for the internet node as **1.3.6.1.4.1**. The number 1at the end of this sysObjectId value refers to the first (and only) subnode (**enterprises**) within the **private** node.
9. Expand the **enterprises** node. The **enterprises** node expands to display a list of device manufacturers.
10. Click each manufacturer node in turn. Review the resulting OID value in the **MIB Variable Information** panel. You get results similar to the information in the following table:

Table 8. Enterprises and their sysObjectIds

enterprise	OID (sysObjectId)
synernetics	1.3.6.1.4.1.114
bicc	1.3.6.1.4.1.170
wellfleet	1.3.6.1.4.1.18
alteon	1.3.6.1.4.1.1872
extremenetworks	1.3.6.1.4.1.1916
networkharmoni	1.3.6.1.4.1.1977
foundry	1.3.6.1.4.1.1991
alliedTelesyn	1.3.6.1.4.1.207

The enterprise corresponding to the sysObjectId `1.3.6.1.4.1977` is Network

Harmoni. This means that the manufacturer of the unclassified device from the **Devices with Unclassified SNMP Object IDs** report is Network Harmoni.

Device classifications are created using active object class (AOC) files. The next step is to check whether any AOC files exist for NetworkHarmoni devices.

For more information on AOC files, see the *IBM Tivoli Network Manager IP Edition Language Reference*.

11. Go to the directory that contains the AOC files:

```
cd $NCHOME/precision/aoc/
ls Net*
```

A listing of the files in this directory with filenames starting with the letters `Net` shows no AOC files for NetworkHarmoni devices.

12. Run the following command to see if the NetworkHarmoni enterprise number is used in any of the AOC files:

```
grep 1977 *.aoc
```

This search retrieves two files: `EndNode.aoc` and `EndNode.NCOMS.aoc`.

Note: The `EndNode.NCOMS.aoc` file is a domain-specific version of the `EndNode.aoc` file. The `EndNode.NCOMS.aoc` file starts off as an exact copy of `EndNode.aoc`. The domain-specific version is created to enable domain-specific class hierarchy settings. Network Manager always looks for a domain-specific version of the file first. If it can't find a domain-specific version, then it uses the generic version.

13. Let's assume that we ran our discovery in the NCOMS domain. Open the NCOMS domain-specific AOC file `EndNode.NCOMS.aoc` .
14. Search for the text `1977` in the file. This retrieves a line that reads: `EntityOID = '1.3.6.1.4.1.1977'` A review of the code around that line shows the following:

```
1]    active object 'EndNode'
2]    {
3]    super_class = 'Core';
4]    instantiate_rule = "EntityOID like '1 \.3\.6\.1\.4\.1\.2021\.' OR
5]                        EntityOID = '1.3.6.1.4.1.2021' OR
6]                        EntityOID = '1.3.6.1.4.1.1575' OR
7]                        EntityOID like '1 \.3\.6\.1\.4\.1\.11\.2\.3\.9\.' OR
8]                        EntityOID = '1.3 .6.1.4.1.11.2.3.9' OR
9]                        (EntityType = 1 AND EntityOID IS NULL)OR
10]                       ... OR
11]                       EntityOID = '1.3.6.1.4.1.1977' OR
12]                       EntityOID like '1\.3\.6\.1\.4\.1\.2136\.' OR
13]                       ...
```

The following table explains this code.

Table 9. Description of the query

Line numbers	Description
1	Name of the class is EndNode
3	The parent of this class is the Core class.
4-13	The instantiate_rule performs a series of matches for each device it encounters. If the relevant device MIB data (in this case each match is attempted with the EntityOID, which is the same as the sysObjectId) matches any of these lines, then the device is assigned to the EndNode class.

Line 11 shows that this AOC file is looking for an *exact* match to the sysObjectId 1.3.6.1.4.1.1977, which is the sysObjectId for the Network Harmoni enterprise. However, this does not match our original unclassified device, because that device has a sysObjectId of 1.3.6.1.4.1977.1.6.1279.1.

This is an error in the regular expression syntax in this AOC file. Line 11 should read:

```
EntityOID like '1\.3\.6\.1\.4\.1\.11\.2\.3\.9\.'
```

This regular would ensure that any devices that has a sysObjectId that begins with 1.3.6.1.4.1.1977 would be classified as an EndNode device. Instead of doing this, you can create a new AOC file that is specific to devices with sysObjectId 1.3.6.1.4.1977.1.6.1279.1 and that classifies this device type as Network Harmoni end node device.

15. Create a new AOC file in the $NCHOME/precision/aoc/ directory. Name this file EndNodeNetHarmoni.aoc.
16. Add the following text to the EndNodeNetHarmoni.aoc file.

```
//************************************************************
//
// File : EndNodeNetHarmoni.aoc
//
//************************************************************
active object 'EndNodeNetHarmoni'
{
  super_class ='EndNode';

  instantiate_rule = "EntityOID like '1 \.3\.6\.1\.4\.1\.1977\.1\.6\.1279\.'";

  extension for Fault = {
  rules = [] ,
  poll_list = [] };

  visual_icon = 'EndNode';
}
```

For more information on classifying network devices, see the *IBM Tivoli Network Manager IP Edition Discovery Guide*.

17. Save the EndNodeNetHarmoni.aoc file.
18. Restart Network Manager using the following commands. This forces the AOC file to be read into the system.

```
itnm_stop ncp
itnm_start ncp
```

19. Run the following command to check that the Active Object Class manager, ncp_class, has restarted correctly. itnm_status The Active Object Class manager manages the AOCs and distributes them to any Network Manager process that needs them.
    - If ncp_class started OK, then it means that new AOC file was set up correctly.
    - If ncp_class does not start, check the following log file for any errors: $NCHOME/log/precision/ncp_class.NCOMS.log.
20. Specify entries in the NCIM topology database deviceFunction and mappings ncim database tables to provide the vendor, model, and function for the Network Harmoni end node device classification. See these two files for the appropriate data and syntax.
    - $NCHOME/precision/scripts/sql/data/populateDeviceFunction.sql
    - $NCHOME/precision/scripts/sql/data/populateMappings.sql

For more information on these NCIM topology database tables, see the *IBM Tivoli Network Manager IP Edition Topology Database Reference*.

You have now checked for unclassified devices and made the necessary modifications to the AOC files. that discovery was able to access devices and have corrected the SNMP community strings for any devices which could not be accessed. The next step is to check for device connectivity.

Checking connectivity

Use reports and topology views to check for missing connections between devices.

1. Click **Reporting** > **Common Reporting**.
2. Click **Reporting** > **Common Reporting** > **Network Manager**. Then click **Troubleshooting Reports**.
3. Select the **Devices with no connections** report. The report displays a list of devices for which connectivity was not discovered properly.

   For more information on the **Devices with no connections** and other discovery troubleshooting reports, see the *IBM Tivoli Network Manager IP Edition Administration Guide*.
4. For each device in the list, try to resolve the problem by doing one of the following:
   - Ensure that Network Manager has been added to the access control list for the device.
   - Check that all the appropriate discovery agents were configured.
   - Check whether the device that you expected to be connected is out of scope.

   Another possible reason for missing connectivity is that the device that you expected to be connected was not discovered due to incorrect seeding of the discovery. In the case of the discovery configuration that was configured earlier, that should not be a problem because you configured ping sweeping on all of the scopes, so each device in scope was individually pinged. However, this is another area to investigate if you configured individual device seeds.

You have now checked for device connectivity and taken action to fix the discovery configuration. The next step is to check for unmanaged interfaces.

Related tasks:

"Ensuring all network technologies are covered" on page 20
Network Manager discovery agents retrieve information from devices in your network. Dedicated agents such as CiscoFrameRelay retrieve specific network technology data. Check the list of full discovery agents to ensure that all the technologies in your network are covered.

Checking for unmanaged interfaces

Use SQL queries to check that devices marked as permanently unmanaged are really meant to be marked as permanently unmanaged.

During discovery, the TagManagedEntities discovery stitcher is used to mark specific interfaces as permanently unmanaged for the purposes of monitoring. This means that the specified interface types are not polled by Network Manager. Interfaces are set to permanently unmanaged on an interface type basis, based on the value of various interface attributes, including the ifDescr attribute. You can check which types of interfaces are marked permanently unmanaged by default and you can change the settings if necessary. By default, interfaces which are virtual or dial up are marked permanently unmanaged, because it costs money to poll these interfaces.

In addition to being marked unmanaged, any events generated on these interfaces will be tagged so that they can be filtered out of event lists by network operators.

1. Use the following SQL command to retrieve a list of interfaces that have been marked as permanently unmanaged by the discovery.

```
SELECT     mnode.entityName AS DeviceName,
           i.entityName AS Interface,
           i.ipAddress AS IPAddress,
           i.ifDescr AS ifDescr,
           m.status AS Status
FROM       ncim.interfaces i
INNER JOIN ncim.entity mnode on mnode.entityId = i.mainNodeEntityId
INNER JOIN ncim.managedStatus m on m.entityId = i.entityId and m.status = 2;
```

2. Check each IP address returned by the query and note any IP addresses that you want to be able to poll and monitor. Let's assume that the results of the query show a number of interfaces that need to be monitored. Let's also assume that each of these IP addresses has the text `Vlan` in their ifDescr MIB variable, and you want to monitor them.
3. Edit the `$NCHOME/precision/disco/stitchers/TagManagedEntities.stch` file.
4. Remove the VLAN interface type from the filter at the end of the file. To do this you must remove the following two lines from the filter:

```
OR
ExtraInfo->m_IfDescr like 'Vlan'
```

5. Save the file.

You have now checked for unmanaged interfaces. You noted that VLAN interfaces had been set to permanently unmanaged. This was not a desired setting, and so you modified the settings in the `TagManagedEntities.stch` stitcher file so that VLAN interfaces are discovered as managed interfaces.

You have performed a number of topology checks. You can perform more topology checks by running other discovery troubleshooting reports.

Assuming that you are now satisfied with the changes that you have made to the discovery configuration, you can run another discovery to check the results of your changes.

Related tasks:

"Launching discovery" on page 22
Now that you have configured your discovery settings, the next step is to manually start your initial discoveries using the Discovery Status GUI.

Verifying the topology: Summary

While working through this information, you used reports, topology views, and SQL queries to check how well the discovery modeled your network. You used the results of these verification activities to adjust the discovery configuration settings.

By working through this information, you learned the following concepts and skills:

- An understanding of why discovery was unable to access devices using SNMP
- How to correct SNMP community string discovery configuration settings for individual devices
- The use of active object class (AOC) files to classify network devices
- How to modify AOC files in order to fix device classification problems
- An understanding of why the topology might be missing connections between devices

- How to identify permanently unmanaged interfaces
- How to modify the assignment of permanently unmanaged interfaces

Configuring production discovery settings

Configure an efficient production discovery by generating a list of the discovered devices and using this list to seed discoveries.

As you work through this information, you will perform the following discovery tasks:

- "Generating a list of discovered devices"
- "Specifying IP addresses to find using the File finder"

Generating a list of discovered devices

After successfully discovering your managed network, generating a list of discovered IP addresses can make subsequent discoveries more productive.

Your initial discoveries used ping sweeping to try every single possible IP address in the subnet. This method is useful when you are configuring initial discoveries as it takes minimal effort to configure and it enables the system to automatically discover all devices within scope. Now, however, you have successfully discovered your managed network and want to schedule more efficient production discoveries. You can do this by generating a list of discovered IP addresses, and using this list as the ping seed list rather than ping sweeping.

Run the `BuildSeedList.pl` Perl script to write the IP addresses discovered by your previous discovery to a file. Issue the following command to do this:

```
$NCHOME/precision/bin/ncp_perl $NCHOME/precision/scripts/perl/scripts/
BuildSeedList.pl -domain NCOMS -outFile seedfile.txt
```

This command builds a device seed list based on the IP addresses discovered by your previous discovery. The command looks in the NCOMS domain, which is the domain in which the previous discovery was run. The file that is output by this script is stored in the following location: `$NCHOME/etc/precision/seedfile.txt`.

You have now generated a device seed list that contains all the devices (and only the devices) in your topology. You will now use this consolidated device seed list to configure a more efficient production discovery.

Specifying IP addresses to find using the File finder

Use the File finder with feedback to configure an efficient discovery that can be used on an ongoing basis in production.

You have now created a seed list containing just the devices in your discovered topology. You are now going to configure the File finder to use this file as your seed list. You must also make the following changes to your discovery configuration to ensure that you have an efficient production discovery that completes more quickly than your initial discoveries:

- Switch off ping sweeping of your scope zones.
- Ensure that the Ping finder is still enabled. You need the Ping finder to verify the existence of devices in your File finder seed list and to ping newly discovered devices connected to your seed devices.
- Ensure that feedback is switched on. You need feedback on in order to discover new devices connected to your seed devices.

1. Click **Discovery** > **Network Discovery Configuration**. This takes you to back to the Discovery Configuration GUI.
2. From the **Domain** list, select the required domain.
3. Switch off ping sweeping of your scope zones.
   a. Click **Scope**.
   b. For each include scope in the **Scope Configuration** table, click on the **Address** field.
   c. In the Scope Properties window, uncheck the **Add to Ping Seed List** option.
   d. Repeat this for each include scope in the **Scope Configuration** table.
4. Ensure that the Ping finder is still enabled.
   a. Click **Seed**.
   b. Check that **Use Ping Finder in Discovery** is checked.
5. Configure the File finder.
   a. Check the **Use File Finder in Discovery** option.
   b. In the File finder section of the GUI. click **New** 
   c. In the **Filename** field of the File Seed Properties window, type the path to the device seed list that you generated. The path to this file is:
      `$NCHOME/etc/precision/seedfile.txt`
   d. In the **Delimiter** field, type `[\t]+`.
   e. Ensure that the **IP Column** is set to 1 and that the **Name Column** is set to 2.
   f. Click **OK**.
6. Ensure that feedback is switched on.
   a. Click **Advanced**.
   b. In the **Advanced Discovery Configuration** section of the panel, check the setting for **Enable Feedback Control**.
      By default, feedback is set to **Feedback only on Full**, and this is the desired setting. This setting ensures that feedback is active when you are performing a *full discovery*, a discovery of your entire network. The entire network is made up of the managed areas that you defined using your discovery scopes.
7. Click Save  to save your discovery configuration settings.

Now that you have configured a more efficient discovery using the File finder, you can schedule regular discoveries to run with these settings.

Configuring production discovery settings: Summary

While working through this information, you configured production discovery settings. You will use these settings to schedule future production discoveries.

By working through this information, you learned the following concepts and skills:

- Use of a ping seed list, instead of ping sweeping, to configure an efficient discovery
- How to use the `BuildSeedList.pl` Perl script to capture IP addresses discovered by your previous discovery
- How to use the File finder to use file containing a list of IP addresses to seed a discovery

Keeping topology up to date

Keep the discovered topology up to date by configuring a discovery schedule for your entire network.

As you work through this information, you will perform the following discovery task: “Scheduling discovery”

Scheduling discovery

Schedule ongoing production discoveries to keep your discovered topology up to date.

Now that you have configured an efficient discovery using the File finder, you can schedule discovery to run on a regular basis. This ensures that any new devices or devices changes are identified and added to the topology.

Let's suppose you want to set up a scheduled discovery to run every night at 3:00 AM.

Run the `scheduleDiscovery.pl` Perl script to schedule a full discovery. Issue the following command to do this:

```
$NCHOME/precision/bin/ncp_perl $NCHOME/precision/bin/scheduleDiscovery.pl
-domain NCOMS -time 03:00
```

This command instructs Network Manager to run a full discovery on the NCOMS domain every night at 3 AM.
In addition to using the `scheduleDiscovery.pl` Perl script to schedule a daily full discovery, you can also use this script to perform other tasks related to discovery scheduling:

- Display the current discovery schedule.
- Schedule a weekly or monthly discovery.
- Schedule discovery to occur at a specified interval; for example, every 48 hours.

For more information on the `scheduleDiscovery.pl` Perl script, see the *IBM Tivoli Network Manager IP Edition Administration Guide*.
The `scheduleDiscovery.pl` Perl script uses the scheduling parameters configured in the command line and uses these to update the `FullDiscovery.stch` discovery stitcher. The discovery daemon checks for changed discovery stitchers every 60 seconds. When it sees that the `FullDiscovery.stch` stitcher has been modified, it sets up the next scheduled discovery.
It is also possible to configure a discovery schedule by editing the `FullDiscovery.stch` stitcher directly. The stitcher is located at:

```
$NCHOME/precision/disco/stitchers/
```

For more information on the `FullDiscovery.stch` stitcher, see the *IBM Tivoli Network Manager IP Edition Discovery Guide*.

Keeping topology up to date: Summary

While working through this information, you configured a discovery schedule for your entire network. You will use these settings on an ongoing basis to keep the discovered topology up to date.

By working through this information, you learned the following concepts and skills:

- How to use the `scheduleDiscovery.pl` Perl script to schedule regular discoveries
- Use of stitchers, pieces of code written in Network Manager's proprietary stitcher language and that perform tasks within the discovery process
- The role of the FullDiscovery.stch in scheduling discoveries.

Viewing the network

Following an initial network discovery, use the following options to view the discovered network: browse the network using the Network Views or search for specific network devices using the Network Hop View.

Browsing the network

Browse the network using the Network Views to visualize the network based on geographical or other groupings. For example, you can browse discovered subnets or device classes. Network Views also highlights discovery issues by showing you devices that the discovery was unable to access or unable to classify.

Before you can work with network views, the first network discovery must have successfully completed, either as part of the installation or immediately following installation.

Restriction: The Administrator user, itnmadmin, has network views assigned by default, so that when you log in with this user, you can view the topology. By default, itnmuser has OOB network views assigned, but other user profiles may not have any network views assigned by default.

1. Click **Availability** > **Network Availability** > **Network Views**.
2. In the **Network Views** tree on the left of the portlet, browse the network by expanding network view nodes of interest. Here are some examples:
   - To browse subnets, click the + symbol next to the Subnets node.
   - To browse VLANs, click the + symbol next to the Global VLANs node.
   - To browse device classes and see devices grouped into categories such as Linux, Sun, and Cisco, click the + symbol next to the Device Classes node.

     Note: Devices which the discovery process could not access using SNMP appear in the NoSNMPAccess sub-node, under the Device Classes node.
3. Click a network view. The network map displays subnets and devices in that network view. Faulty devices are displayed with an associated event icon.

Searching for network devices

You can search for a specific device in the discovered network topology using the Network Hop View.

1. Click **Network Availability** > **Network Hop View**.
2. Select a network domain from the **Domain** list.
3. Click **Search for Seed Device**  to specify the device to search for.
4. In the Entity Search window, ensure that the Basic tab is selected and complete the search criteria fields.

   Domain
   Select the domain in which you want to search.

   IP Address
   Specify the IP address of the device. You can specify all of the address, or only the first part of the address. You can also use the percent character (%) or the asterisk (*) as wildcards.

   Device Name
   Specify the name of the device. You can specify all of the name, or only the first part of the name. You can also use the percent character (%) or the asterisk (*) as wildcards. Device names are not case-sensitive. If you specify both an IP address and a device name, the IP address takes precedence.
5. Click **Find**. The **Results** list box displays the devices resulting from your search, as a listing of IP addresses or entity names.
6. Select the device you want from the **Results** list box, and click **Select & Close** to return to the Network Hop View main window. The **Seed device** field in the Network Hop View toolbar is populated with the seed device IP address or host name.

   Tip: If you know the entity ID of the device, you can also type it into the **Seed** field. Do not type device IP addresses or hostnames into the **Seed** field.
7. Select the maximum number of hops displayed from the seed device from the **Hops** list. This setting shows more or less devices connected to the seed device.
8. Specify how to display connectivity:

   Layer 2
   Displays all switched connections between devices in the topology. A layer 2 view typically shows switch and hub connections.

   Layer 3
   Shows routers and the connections between routers. Switches are not normally displayed.

   Note: If switches have active connections involving layer 3 interfaces, they are included in this layout.

   The connections between devices are displayed as follows:
   - Connections between two layer 3 interfaces are shown as normal.
   - Connections between a layer 3 and a layer 2 interface are shown as being between the layer 3 interface and the subnet to which the layer 2 interface belongs.
   - Connections between two layer 2 interfaces are not shown.

IP Subnets
Shows all devices within a subnet connected to a subnet cloud. This layout helps to simplify the network map and also helps to make subnet membership clear. If you want to see all connections, select **Layer 3** to show all routers and connections between them, or **Layer 2** for data link connections.

PIM Shows all devices that belong to Protocol Independent Multicast (PIM) groups.

IPMRoute
Shows all devices that belong to Internet Protocol Multicast (IPM) routes.

9. Click **Apply Changes**  . The topology you selected is displayed in the network map. Faulty devices are displayed with an associated event icon.

   Fix Pack 4 To see which domain a device is in, hover the cursor over the device. If a cross-domain discovery is configured, the Network Hop View results can include devices that are in a different domain than the domain that contains the seed device.

Network map icons and symbols

The Network Hop View and the Network Views show icons representing discovered devices and subnets and the event status associated with a device.

Devices and subnets

The following table describes the device and device connectivity icons used in the network map. Within the network map a solid line indicates a connection between devices and a pale dashed line indicates membership; for example, membership of a subnet or of a BGP autonomous system.

Table 10. Icons used in general network maps

Icon	Name	Description
	Router icon	Represents a router.
	Switch icon	Represents a switch.
	End node icon	Represents end-node devices, including Windows, Linux, and Solaris workstations and printers.
	Unknown device icon	System cannot determine the correct icon to use for this device. The most likely reason is failed SNMP access to the device.
		Represents a subnet

Table 10. Icons used in general network maps (continued)

Icon	Name	Description
[4]	Number of connections indicator	This indicates either of the following: • In the case of a connection relationship between two devices, which is indicated by a solid line, this number indicates the number of interfaces participating in the connection between the devices. • In the case of a membership relationship, which is indicated by a pale dashed line, this number indicates the number of interfaces participating in membership, for example, of a subnet or OSPF area.
	Completely unmanaged device	The entire device, including all its interfaces, is unmanaged.
	Partially unmanaged device	Only certain components of this device are unmanaged.

Event status

The following table shows the default event status icons.

Table 11. Default event status icons

Severity or meaning	Color in the Active Event List (AEL)	Default icon in the Network Views
5 (critical)	Red	
4 (major)	Orange	
3 (minor)	Yellow	
2 (warning)	Blue	
1 (indeterminate)	Purple	
0 (clear)	Green	
No status has been retrieved for this device. If this persists, there may be an error.	Not displayed in the **AEL**	
There are no events for this device. This icon appears in the Network Views tree only.	Not displayed in the **AEL**	

Table 11. Default event status icons (continued)

Severity or meaning	Color in the Active Event List (AEL)	Default icon in the Network Views
This icon appears next to unmanaged devices in the Network Views and Hop View network map. This icon also appears next to the unmanaged components in the Structure Browser.	Not displayed in the **AEL**	
This icon appears in the Network Views and Hop View network map next to devices that contain unmanaged components.	Not displayed in the **AEL**	

For more information about visualizing the network, see the *IBM Tivoli Network Manager IP Edition Network Visualization Setup Guide*.

Creating user profiles for Network Operators

Create user profiles for your Network Operators and assign them to the Network_Manager_User group. This automatically assigns all the required roles to the user.

1. Click **Create** to create a new user.
2. In the **User ID** field, type a unique name to identify the user. This user ID is added to the user registry and is also used as the login account name. For example, you might type `dlucas`
3. Click **Group Membership** and then follow the steps in "Assigning user profiles to the Network_Manager_User group" on page 45 to add the user as a member of one or more existing groups.
4. In the **First name** field, type the given or first name of the user. For example, you might type `Diana`.
5. In the **Last name** field, type the family or last name of the user. For example, you might type `Lucas`.
6. Optional: In the **E-mail** field, type an e-mail address for the user. For example, you might type `dlucas@tivoli.com`.
7. In the **Password** field, type a unique password. For example, you might type `d4lucas`.
8. In the **Confirm password** field, type the same password again.
9. Click **Create**. If successful, a message displays that indicates that the user has been created. Also, the user ID and other user information is added to the user registry, and a new login account is created for the user.
10. To create another user, click **Create Another**.
11. Repeat the process until you have created all the new users.

Assigning user profiles to the Network_Manager_User group

Automatically assigns all the required roles to a Network Operator by assigning a user to the Network_Manager_User group.

The Network_Manager_User group provides all the necessary Network Manager roles for a network operator.

1. During the process of "Creating user profiles for Network Operators" on page 44, click **Group Membership**.
2. In the **Search by** field, select the attribute from the list that you want to use to search for one or more users. For example, select **Group name**.
3. In the **Search for** field, either type the string that you want to search for to limit the set of groups, or use the wildcard character (*) to search for all groups. Whether the search is case sensitive or case insensitive depends on the user registry that you are using.
4. In the **Maximum results** field, specify the maximum number of search results that you want to display.
5. Click **Search**. After the search completes, the results are displayed in two lists: one list is for groups that matched the search criteria and one list, named **Current Groups**, is for groups that the user is already a member of.
6. To add the user to one or more groups, highlight the groups from the matching groups list to select them. For example, to assign users to the Network_Manager_User group, highlight `Network_Manager_User` and then click **< Add**.
7. Optional: To undo or remove the user as a member, highlight the groups from the **Current Groups** list and then click **Remove >**.
8. Return to the process of "Creating user profiles for Network Operators" on page 44 to complete the steps.

Roles assigned to the Network_Manager_User group

The Network_Manager_User group provides all the necessary Network Manager roles for a network operator.

Table 12. Roles for the Network_Manager_User

Role	Description
ncp_hopview	Allows the user to access the Hop View.
ncp_networkview	Allows the user to access the Network Views and to display any of the following views: • User Views: Network views created by the user. • Group Views: Views assigned to the group or groups that this user belongs to. • Global Views: Views accessible to all users regardless of the group to which they belong.
ncp_networkview_admin_user	Allows the user to create, edit, partition, and delete their own set of network views. This role also allows the user to perform move operations on network views within a user view.
ncp_mibbrowser	Allows the user to access the MIB Browser.

Table 12. Roles for the Network_Manager_User (continued)

Role	Description
ncp_mibbrowser_config	Allows the user to access the MIB Browser for configuration purposes.
ncp_structurebrowser	Allows the user to use the Structure Browser.
ncp_webtools	Allows the user to use the Web Tools.

For more information about creating user profiles, see the *IBM Tivoli Network Manager IP Edition Administration Guide*.

Making the network topology visible to Network Operators

To enable Network Operators to view the discovered topology, add network views to the view collections that are associated with the users or user groups of the Operators.

A network discovery must have run, so that it is possible to view the generated topology. Also, the required users must have been created and assigned to the Network_Manager_User user group.

Unlike the itnmadmin user, newly-created users, for example Network Operators, do not have any access to the network topology through the Network Views. You must associate views of the network with the Operators' users or with their user groups. You do this in one of the following ways:

Creating new network views
Assign a new network view to a user or user group.

Copying existing network views
Copy a network view from the itnmadmin user and assign it to a user or user group.

The itnmadminuser has access to the network views that are assigned to each user and user group. For example, to access the network views of the user group “Public”, select **Public Views.** from the list. To access the views of a user called “dlucas ”, select **dlucas Views**.

- To copy a network view from the itnmadmin user to other user or groups:
  1. Click **Availability** > **Network Availability** > **Network Views**.
  2. From the list, select **itnmadmin Views**.
  3. Click **Copy or Move View**  .
  4. Select **Copy**.
  5. From the **To** list, select the network view collection to which to add the network view.

     Tip: To make the network view available to all users, select **Global**.
- To create an initial network view that displays all the subnets on a domain:
  1. Click **Availability** > **Network Availability** > **Network Views**.
  2. From the list, select the network view collection for the required user or user group and click **New View**  .

Tip: To make the network view available to all users, select **Global**.

3. Type a name for the network view.
4. In the **Parent** list, select `NONE`.
5. In the **Type** list, select `Dynamic Views - Subnet`.
6. In the remaining fields, specify the layout and style, and the icons that represent the network view. The map icon is used to represent the network view in the right display panel, whereas the tree icon is used to represent the network view in the left navigation panel.
7. Click **Filter**.
8. Select the domain.
9. In the **Subnet Classes** field, select **A & B** or **A, B & C**. If you select **A, B & C**, a large number of subnets is created, because most networks contain large numbers of class C subnets.
10. Click **OK**.

- To create an initial network view that displays all the entities on the network for a domain:
  1. Click **Availability** > **Network Availability** > **Network Views**.
  2. From the list, select the network view collection for the required user or user group and click **New View**  .

     Tip: To make the network view available to all users, select **Global**.
  3. Type a name for the network view.
  4. In the **Parent** list, select `NONE`.
  5. In the **Type** list, select `Dynamic Views - Template`.
  6. In the remaining fields, specify the layout and style, and the icons that represent the network view. The map icon is used to represent the network view in the right display panel, whereas the tree icon is used to represent the network view in the left navigation panel.
  7. Click **Filter**.
  8. Select the domain.
  9. In the **Template** field, select `IP Default`.
  10. Click **OK**.

When Network Operators log in with their users and open the Network Views, they can click the network views that you created or copied for them to begin visualizing the network.

To verify that the Operators can now display network views, select from the list a user or user group for which you have created or copied network views. For example, select **Global** if you created or copied network views for all users. The navigation tree now contains network views.

Alternatively, log out of Network Manager, and log back in as a user that now has network views assigned.

Now that the Operators have access to the topology, and can work, you can create additional network views that are designed around the access concept that you want to develop for the network. If you then want to revise Operators' access you can delete these initial network views.

For more information on administering network views, see the *IBM Tivoli Network Manager IP Edition Administration Guide*. For more information about visualizing the network, see the *IBM Tivoli Network Manager IP Edition Network Visualization Setup Guide*.

Viewing network events

When you have discovered your network and given access to your operators, you can configure what events are raised on the network, and then view those events in a simple list or in the context of the network topology.

About polling the network

To poll the network, Network Manager periodically sends queries to the devices on the network. These queries determine the behavior of the devices, for example operational status, or the data in the Management Information Base (MIB) variables of the devices.

Network polling is controlled by poll policies. Poll policies consist of the following:

- Poll definitions, which define the data to retrieve.
- Poll scope, consisting of the devices to poll. The scope can also be modified at a poll definition level to filter based on device class and interface.
- Polling interval and other poll properties.

Network Manager uses the IBM Tivoli Netcool/OMNIbus SNMP trap probe and the Syslog probe to monitor the network. To run Tivoli Netcool/OMNIbus probes, use Tivoli Netcool/OMNIbus process control.

For more information about how to use Tivoli Netcool/OMNIbus process control, see the *IBM Tivoli Netcool/OMNIbus Administration Guide*.

The polling process is controlled by the ncp_poller process. The ncp_poller process stores SNMP information in the ncmonitor database; other data is stored in-memory.

Network Manager has a multiple poller mechanism to distribute the load. If the default poller cannot handle the polling demands for your network, you might need to use the multiple poller feature.

Enabling polls

To generate network events, you must enable some of the default poll policies.

By default, only a few poll policies are enabled. Enabling poll policies creates network traffic. Only enable those poll policies that will give you useful information about your network.

Which poll policies to enable depends on what network technologies you are using, what device types are in your network, and what kind of information you want to monitor. As an example, we will enable some poll policies and later we will view events generated by these policies.

1. Click **Administration** > **Network** > **Network Polling**.
2. Select the check box next to the **Default Interface Ping** poll policy. This poll policy uses the Default Interface Ping poll definition to ping all interfaces on every main node with a valid IP address. Because in a typical network many

interfaces might be in an administratively down state, this policy is likely to generate significant traffic and some network events.

3. Select the check box next to the **cpuBusyPoll** poll policy. This poll policy uses the cpuBusyPoll poll definition to generate an event when the average CPU usage of a Cisco device exceeds 80%.
4. Select the check box next to the **ciscoEnvMonTemperatureState** poll policy. This poll policy uses the ciscoEnvMonTemperatureState poll definition to generate an event when the temperature of a Cisco device is reported as anything other than normal.
5. Click **Enable Selected Policies**  to enable these policies.
6. Click **OK**.

Each poll runs at certain intervals. After half an hour or so, check whether any events can be seen in the network views and **Active Event List (AEL)**.

Viewing events in the network views

You can use the network views to check that certain sections of the network or certain kinds of devices are free of problems.

To perform this task you must be logged in as the administrator, or the administrator must have given you access to the relevant network views.

In the Enabling Polls task, you enabled some poll policies that monitor Cisco routers for various error conditions. Now you can use the network views to see if any of the routers on your network have events associated with them.

1. Click **Availability** > **Network Availability** > **Network Views**.
2. In the navigation tree on the left of the portlet, click the + symbol to expand the **All Routers** network view nodes. All routers that have been discovered are displayed.
3. Any routers with any events against them are displayed with an event status icon next to them.
4. Right-click on any device to see the tools available for working with the device.

Viewing events in the Active Event List (AEL)

You can use the **Active Event List (AEL)** to see all network events.

In the Enabling Polls task, you enabled some poll policies that poll device interfaces, and others that monitor Cisco routers. Additionally, all devices in the network (except end nodes such as printers) are pinged by default. Now you can use the **AEL** to see if any events have been raised on the network. You can also see the network context in which an event appears using the Network Hop View.

1. Click **Availability** > **Network Availability** > **Fault-Finding View**. The **Fault-Finding View** page appears with the **Active Event List (AEL)** portlet above and the Network Hop View portlet below.

   Note: When you first open the **Fault-Finding View** page, the **AEL** portlet displays all events in the ObjectServer and the Network Hop View portlet is empty.
2. Select an event of interest in the **AEL**, or right-click an event and then click **Broadcast Topology Context**. The Network Hop View portlet now displays the network topology related to the selected event.

Restriction: Results vary if you select multiple events in the **AEL**.

- If all the selected events occurred on the same network device, then the Network Hop View portlet only displays the network topology related to that device.
- If the selected events occurred on different devices, then the Network Hop View portlet does not display any network topology .

Now you can investigate the events. You can investigate the root cause of events, view the structure of devices related to an event, and perform other tasks by right-clicking on an event.

Appendix. Network Manager glossary

Use this information to understand terminology relevant to the Network Manager product.

The following list provides explanations for Network Manager terminology.

AOC files
Files used by the Active Object Class manager, `ncp_class` to classify network devices following a discovery. Device classification is defined in AOC files by using a set of filters on the object ID and other device MIB parameters.

active object class (AOC)
An element in the predefined hierarchical topology of network devices used by the Active Object Class manager, `ncp_class`, to classify discovered devices following a discovery.

agent See, discovery agent.

class hierarchy
Predefined hierarchical topology of network devices used by the Active Object Class manager, `ncp_class`, to classify discovered devices following a discovery.

configuration files
Each Network Manager process has one or more configuration files used to control process behaviour by setting values in the process databases. Configuration files can also be made domain-specific.

discovery agent
Piece of code that runs during a discovery and retrieves detailed information from discovered devices.

Discovery Configuration GUI
GUI used to configure discovery parameters.

Discovery engine (ncp_disco)
Network Manager process that performs network discovery.

discovery phase
A network discovery is divided into four phases: Interrogating devices, Resolving addresses, Downloading connections, and Correlating connectivity.

discovery seed
One or more devices from which the discovery starts.

discovery scope
The boundaries of a discovery, expressed as one or more subnets and netmasks.

Discovery Status GUI
GUI used to launch and monitor a running discovery.

discovery stitcher
Piece of code used during the discovery process. There are various discovery stitchers, and they can be grouped into two types: data collection stitchers, which transfer data between databases during the data collection

phases of a discovery, and data processing stitchers, which build the network topology during the data processing phase.

domain
See, network domain.

entity A topology database concept. All devices and device components discovered by Network Manager are entities. Also device collections such as VPNs and VLANs, as well as pieces of topology that form a complex connection, are entities.

event enrichment
The process of adding topology information to the event.

Event Gateway (ncp_g_event)
Network Manager process that performs event enrichment.

Event Gateway stitcher
Stitchers that perform topology lookup as part of the event enrichment process.

failover
In your Network Manager environment, a failover architecture can be used to configure your system for high availability, minimizing the impact of computer or network failure.

Failover plug-in
Receives Network Manager health check events from the Event Gateway and passes these events to the Virtual Domain process, which decides whether or not to initiate failover based on the event.

Fault Finding View
Composite GUI view consisting of an **Active Event List (AEL)** portlet above and a Network Hop View portlet below. Use the Fault Finding View to monitor network events.

full discovery
A discovery run with a large scope, intended to discover all of the network devices that you want to manage. Full discoveries are usually just called discoveries, unless they are being contrasted with partial discoveries. See also, partial discovery.

message broker
Component that manages communication between Network Manager processes. The message broker used byNetwork Manager is called Really Small Message Broker. To ensure correct operation of Network Manager, Really Small Message Broker must be running at all times.

NCIM database
Relational database that stores topology data, as well as administrative data such as data associated with poll policies and definitions, and performance data from devices.

ncp_disco
See, Discovery engine.

ncp_g_event
See, Event Gateway.

ncp_model
See, Topology manager.

ncp_poller
See, Polling engine.

network domain
A collection of network entities to be discovered and managed. A single Network Manager installation can manage multiple network domains.

Network Health View
Composite GUI view consisting of a Network Views portlet above and an **Active Event List (AEL)** portlet below. Use the Network Health View to display events on network devices.

Network Hop View
Network visualization GUI. Use the Network Hop View to search the network for a specific device and display a specified network device. You can also use the Network Hop View as a starting point for network troubleshooting. Formerly known as the Hop View.

Network Polling GUI
Administrator GUI. Enables definition of poll policies and poll definitions.

Network Views
Network visualization GUI that shows hierarchically organized views of a discovered network. Use the Network Views to view the results of a discovery and to troubleshoot network problems.

OQL databases
Network Manager processes store configuration, management and operational information in OQL databases.

OQL language
Version of the Structured Query Language (SQL) that has been designed for use in Network Manager. Network Manager processes create and interact with their databases using OQL.

partial discovery
A subsequent rediscovery of a section of the previously discovered network. The section of the network is usually defined using a discovery scope consisting of either an address range, a single device, or a group of devices. A partial discovery relies on the results of the last full discovery, and can only be run if the Discovery engine, `ncp_disco`, has not been stopped since the last full discovery. See also, full discovery.

Path Views
Network visualization GUI that displays devices and links that make up a network path between two selected devices. Create new path views or change existing path views to help network operators visualize network paths.

performance data
Performance data can be gathered using performance reports. These reports allow you to view any historical performance data that has been collected by the monitoring system for diagnostic purposes.

Polling engine (ncp_poller)
Network Manager process that polls target devices and interfaces. The Polling engine also collects performance data from polled devices.

poll definition
Defines how to poll a network device or interface and further filter the target devices or interfaces.

poll policy
Defines which devices to poll. Also defines other attributes of a poll such as poll frequency.

Probe for Tivoli Netcool/OMNIbus (nco_p_ncpmonitor)
Acquires and processes the events that are generated by Network Manager polls and processes, and forwards these events to the ObjectServer.

RCA plug-in
Based on data in the event and based on the discovered topology, attempts to identify events that are caused by or cause other events using rules coded in RCA stitchers.

RCA stitcher
Stitchers that process a trigger event as it passes through the RCA plug-in.

root-cause analysis (RCA)
The process of determining the root cause of one or more device alerts.

SNMP MIB Browser
GUI that retrieves MIB variable information from network devices to support diagnosis of network problems.

SNMP MIB Grapher
GUI that displays a real-time graph of MIB variables for a device and usse the graph for fault analysis and resolution of network problems.

stitcher
Code used in the following processes: discovery, event enrichment, and root-cause analysis. See also, discovery stitcher, Event Gateway stitcher, and RCA stitcher.

Structure Browser
GUI that enables you to investigate the health of device components in order to isolate faults within a network device.

Topology Manager (ncp_model)
Stores the topology data following a discovery and sends the topology data to the NCIM topology database where it can be queried using SQL.

WebTools
Specialized data retrieval tools that retrieve data from network devices and can be launched from the network visualization GUIs, Network Views and Network Hop View, or by specifying a URL in a web browser.

Notices

This information applies to the PDF documentation set for IBM Tivoli Network Manager IP Edition 3.9.

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

For license inquiries regarding double-byte character set (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan, Ltd.
19-21, Nihonbashi-Hakozakicho, Chuo-ku
Tokyo 103-8510, Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web

sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation
958/NH04
IBM Centre, St Leonards
601 Pacific Hwy
St Leonards, NSW, 2069
Australia
IBM Corporation
896471/H128B
76 Upper Ground
London
SE1 9PZ
United Kingdom
IBM Corporation
JBF1/SOM1 294
Route 100
Somers, NY, 10589-0100
United States of America

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the

names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs.

Trademarks

The terms in Table 13 are trademarks of International Business Machines Corporation in the United States, other countries, or both:

Table 13. IBM trademarks

AIX	iSeries	RDN
ClearQuest	Lotus	SecureWay
Cognos	Netcool	solidDB
Current	NetView	System z
DB2	Notes	Tivoli
developerWorks	OMEGAMON	WebSphere
Enterprise Storage Server	PowerVM	z/OS
IBM	PR/SM	z/VM
Informix	pSeries	zSeries

Intel, Intel logo, Intel Inside, Intel Inside logo, Intel Centrino, Intel Centrino logo, Celeron, Intel Xeon, Intel SpeedStep, Itanium, and Pentium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

Java™ and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

Privacy policy considerations

IBM Software products, including software as a service solutions, ("Software Offerings") may use cookies or other technologies to collect product usage information, to help improve the end user experience, to tailor interactions with the end user or for other purposes. In many cases no personally identifiable information is collected by the Software Offerings. Some of our Software Offerings can help enable you to collect personally identifiable information. If this Software Offering uses cookies to collect personally identifiable information, specific information about this offering's use of cookies is set forth below.

This Software Offering does not use cookies or other technologies to collect personally identifiable information.

For more information about the use of various technologies, including cookies, for these purposes, see IBM's privacy policy at http://www.ibm.com/privacy.

Index

Printed in the Republic of Ireland