Network Manager IP Edition
Version 4 Release 1.1

*Product Overview*
Network Manager IP Edition
Version 4 Release 1.1

Product Overview
## Contents

<table>
<thead>
<tr>
<th>About this publication</th>
<th>v</th>
</tr>
</thead>
<tbody>
<tr>
<td>Audience for this information</td>
<td>v</td>
</tr>
<tr>
<td>What this information contains</td>
<td>v</td>
</tr>
<tr>
<td>Publications</td>
<td>vi</td>
</tr>
<tr>
<td>Accessibility</td>
<td>ix</td>
</tr>
<tr>
<td>Tivoli technical training</td>
<td>ix</td>
</tr>
<tr>
<td>Support and community information</td>
<td>x</td>
</tr>
<tr>
<td>Conventions used in this publication</td>
<td>xi</td>
</tr>
</tbody>
</table>

### Chapter 1. About Network Manager . . . . 1

<table>
<thead>
<tr>
<th>What's new in this release</th>
<th>2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Network Manager architecture</td>
<td>8</td>
</tr>
<tr>
<td>Network Manager data flow</td>
<td>11</td>
</tr>
<tr>
<td>Integration with other products</td>
<td>14</td>
</tr>
<tr>
<td>Network layer</td>
<td>16</td>
</tr>
<tr>
<td>About discovery</td>
<td>16</td>
</tr>
<tr>
<td>About EMR-based discovery</td>
<td>21</td>
</tr>
<tr>
<td>About polling</td>
<td>25</td>
</tr>
<tr>
<td>Data layer</td>
<td>28</td>
</tr>
<tr>
<td>About topology storage</td>
<td>28</td>
</tr>
<tr>
<td>About root cause analysis and event enrichment</td>
<td>30</td>
</tr>
<tr>
<td>About event storage</td>
<td>36</td>
</tr>
<tr>
<td>About historical polled data collection and storage</td>
<td>38</td>
</tr>
<tr>
<td>Visualization layer</td>
<td>41</td>
</tr>
<tr>
<td>About topology visualization</td>
<td>41</td>
</tr>
<tr>
<td>About event visualization</td>
<td>43</td>
</tr>
<tr>
<td>About reporting</td>
<td>45</td>
</tr>
<tr>
<td>Tivoli Integrated Portal</td>
<td>47</td>
</tr>
<tr>
<td>Network Manager web applications</td>
<td>47</td>
</tr>
<tr>
<td>Web application architecture</td>
<td>48</td>
</tr>
<tr>
<td>Single sign-on</td>
<td>50</td>
</tr>
</tbody>
</table>

### Chapter 2. Benefits of Network Manager . . . . . 51

| Comprehensive network management | 51 |
| Flexible network visualization | 51 |
| Built-in device and interface polling capabilities | 51 |
| Built-in root-cause analysis capabilities | 51 |
| Single-click network troubleshooting | 52 |
| Rich network topology and event data | 52 |
| Increasingly bigger network discovery | 52 |
| Extensive reporting capabilities | 52 |
| Fully customizable content | 53 |
| Multiple integration options | 53 |

### Chapter 3. Deployment of Network Manager . . . . . 55

<table>
<thead>
<tr>
<th>Deployment scenarios</th>
<th>55</th>
</tr>
</thead>
<tbody>
<tr>
<td>Network and deployment comparisons</td>
<td>55</td>
</tr>
<tr>
<td>Demonstration or educational system deployment</td>
<td>60</td>
</tr>
<tr>
<td>Small customer network</td>
<td>60</td>
</tr>
<tr>
<td>Medium customer network</td>
<td>61</td>
</tr>
<tr>
<td>Large customer network</td>
<td>62</td>
</tr>
<tr>
<td>Very large customer network</td>
<td>64</td>
</tr>
<tr>
<td>Telecommunications company network</td>
<td>65</td>
</tr>
<tr>
<td>LTE 4G wireless telecommunications company network</td>
<td>66</td>
</tr>
<tr>
<td>Deployment considerations</td>
<td>68</td>
</tr>
<tr>
<td>Deployment examples</td>
<td>70</td>
</tr>
<tr>
<td>Constraints for installing and starting components</td>
<td>70</td>
</tr>
<tr>
<td>Example simple deployment architecture</td>
<td>71</td>
</tr>
<tr>
<td>Example large deployment architecture</td>
<td>73</td>
</tr>
</tbody>
</table>

### Appendix. Network Manager glossary 77

### Notices 81

### Trademarks 83

### Index 85
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 IP Edition Product Overview describes the benefits of Network Manager. It also describes the architecture and functionality of Network Manager, and provides a basic overview of the tasks of Network Manager users. The document also provides initial deployment and capacity planning guidance.

Audience for this information

This information is for users and system and network administrators. It describes the product architecture, components, and functionality. This information is also for anyone interested in IBM Tivoli Network Manager IP Edition.

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 information contains

This information contains the following sections:

• Chapter 1, “About Network Manager,” on page 1
  Provides an introduction to IBM Tivoli Network Manager IP Edition, and describes the functions and features of the product. This sections also describes the benefits of Network Manager and provides deployment best practices.

• Chapter 2, “Benefits of Network Manager,” on page 51
  Describes Network Manager’s flexible visualization capabilities and network troubleshooting capabilities, including root-cause analysis. This chapter also describes the powerful polling and reporting capabilities, as well the ability to use data from third-party sources to perform topology enrichment and event enrichment.

• Chapter 3, “Deployment of Network Manager,” on page 55
  Provides initial deployment and capacity planning guidance.
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 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:


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:
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.
- 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 Network Manager Information Center is accessibility-enabled. The accessibility features of the information center are described in Accessibility and keyboard shortcuts in the information center.

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 and community information

Use IBM Support, Service Management Connect, and Tivoli user groups to connect with IBM and get the help and information you need.

IBM Support

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

Tivoli user groups

Tivoli user groups are independent, user-run membership organizations that provide Tivoli users with information to assist them in the implementation of Tivoli Software solutions. Through these groups, members can share information and learn from the knowledge and experience of other Tivoli users. Tivoli user groups include the following members and groups:

- 23,000+ members
- 144+ groups

Access the link for the Tivoli Users Group at www.tivoli-ug.org.

Service Management Connect

Access Service Management Connect at https://www.ibm.com/developerworks/servicemanagement/ Use Service Management Connect in the following ways:

- Become involved with transparent development, an ongoing, open engagement between other users and IBM developers of Tivoli products. You can access early designs, sprint demonstrations, product roadmaps, and prerelease code.
- Connect one-on-one with the experts to collaborate and network about Tivoli and the (enter your community name here) community.
- Read blogs to benefit from the expertise and experience of others.
- Use wikis and forums to collaborate with the broader user community.
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

**Bold monospace**
- Command names, and names of macros and utilities that you can type as commands
- Environment variable names in text
- Keywords
- Parameter names in text: API structure parameters, command parameters and arguments, and configuration parameters
- Process names
- Registry variable names in text
- Script names
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.

On UNIX systems, preface environment variables with the dollar sign $. For example, on UNIX, NCHOME is $NCHOME.
Chapter 1. About Network Manager

Network Manager provides the features necessary to manage complex networks. These features include network discovery, device polling, including storage of polled SNMP and ICMP data for reporting and analysis, and topology visualization. In addition, Network Manager can display network events, perform root-cause analysis of network events, and enrich network events with topology and other network data.

Network Manager integrates with other IBM products, such as IBM Tivoli Application Dependency Discovery Manager, IBM Tivoli Business Service Manager and IBM Systems Director.

Network Manager is not suitable for monitoring the performance of networks. To monitor the performance of networks, use the IBM Tivoli® Netcool® Performance Management product. For more information, see the Tivoli Netcool Performance Management information center at http://publib.boulder.ibm.com/infocenter/tivihelp/v8r1/topic/com.ibm.tnpm.doc/welcome_tnpm_allversions.html.

Network Manager features include the following:

**Manage modern complex networks**

Network Manager discovers, polls, and visualizes complex networks, containing a wide range of network-type devices, such as routers and switches, and using network protocols and technologies, such as MPLS, BGP, and OSPF.

**View the network in multiple ways**

Different ways to visualize the network include network views, which show standard and customized device groupings such as subnets, VLANs, and VPNS, and the Network Hop View, which shows a selected device and all devices connected to it up to a configurable number of connections. You can also navigate the interfaces and other components of a device using the Structure Browser. Multi-portlet views enable you to put these views together; for example, you can select a device in the Network Hop View and instantly see the interfaces and other components of the device in an adjacent Structure Browser portlet. You can also use multiportlet views to show simultaneous topology maps and event lists.

**Apply ready-to-use device and interface polling capabilities**

Network Manager provides a set of ready-to-use device and interface polls, including ICMP polls and MIB variable threshold polls. The MIB variable threshold polls generate network events if thresholds are violated on specified MIB variables. You can customize network polling so that events are received when thresholds are violated on any MIB variable on your network devices.

**Leverage built-in root-cause analysis capabilities**

Network Manager sorts through multiple network events and uses knowledge of network topology to determine a single root-cause event. Network Manager highlights root-cause events in event lists and in topology maps so that your operators can instantly determine where to begin troubleshooting the network.
Troubleshoot network problems using right-click tools
Network Manager provides a set of ready-to-use right-click tools to enable operators to troubleshoot network devices shown in network topology maps and event lists. For example, you can perform diagnostic actions such as ping and traceroutes and you can retrieve device information such as DNS lookups or retrieve more complex protocol information such as BGP and OSPF information. You can add custom right-click tools to perform any desired action on a device.

Generate richer network visualization and event data
You can enrich the network topology by customizing the discovery to retrieve and store data about the discovered devices from third-party data sources. For example, you could retrieve customer information related to devices from a third-party inventory database, thereby enabling network operators to see the customer associated with a given device or network event.

You can also enrich network events with any topology data retrieved by the discovery process. For example, standard network events on device interfaces that originate from traps show the interface index only. You can enrich these events with interface name and description data. Operators viewing network events on device interfaces can then easily identify the interface.

Discover increasingly bigger networks
Network Manager can discover and manage increasingly bigger networks.

Run reports to retrieve essential network data
You can run reports to retrieve a wide range of network data, including data on network availability, network assets, and network technology.

Build custom multi-portlet pages
You can build pages that contain any combination of data. For example, you can combine topology maps with device structure views and event lists. You can also combine discovery status information with event lists that show custom discovery events.

Integrate with a wide range of Tivoli products
By default Network Manager integrates with a number of Tivoli products, including IBM Tivoli Business Service Manager (TBSM), IBM Tivoli Application Dependency Discovery Manager (TADDM), IBM Systems Director, and IBM Tivoli Monitoring (ITM).

What's new in this release
Network Manager V4.1.1 provides new features and enhancements that include the following areas: Long Term Evolution (LTE) network discovery and visualization, full 64-bit support, the ability to filter out certain SNMP information from discovery, entity smart search option, and enhanced polling functions.

Summary of new features
This description of the new features and enhancements is also available in the Release Notes.

Fix Pack 1
Fix pack 1
The following features and functions are introduced in fix pack 1.
Extended browser support
Network Manager 4.1.1 Fix Pack 1 introduces support for Internet Explorer 11 and Mozilla Firefox 31 Extended Support Release (ESR). Support for these new browsers requires Web GUI 7.4 Fix Pack 3 and Tivoli Integrated Portal 2.2.0.15, or later.

For more information, see the IBM Tivoli Network Manager IP Edition Installation and Configuration Guide.

Extended platform support
Network Manager 4.1.1 Fix Pack 1 introduces support for SuSE Linux Enterprise Server (SLES) 11.0 (x86-64) SP2 and SP3.

For more information, see the IBM Tivoli Network Manager IP Edition Installation and Configuration Guide.

Alcatel5620SamSoap Java Collector
This is a newly introduced Java collector for the Alcatel Lucent SAM 5620. This Java collector retrieves the same information as the Perl Alcatel5620SamSoapFindToFile collector, and additionally supports the Java Message Service (JMS), High Availability (HA) notifications, Link Aggregation Group (LAG) information, and notifications about added and removed devices. When the collector receives an HA notification from the JMS that indicates a loss of connection to the SAM 5630 primary server, the collector pauses to allow time for the backup server to become the primary server, and attempts to reconnect to the primary server. If after four reconnection attempts the primary server is still not available, the collector exits. The collector retrieves LAG containment data, which is modeled and presented in the Structure Browser. The collector listens for JMS notifications about devices that have been added or removed from the EMS. The collector pauses and then redisCOVERs the SAM 5620.

For more information, see the IBM Tivoli Network Manager IP Edition Discovery Guide.

Cisco Nexus support (support for data centre/virtual environment)
The CiscoNexusContext discovery agent discovers VRF context-sensitive information from Cisco Nexus family devices. The SNMP context is used to discover the IP address and IP routing data from non-default VRFs. The CiscoNexusVdc discovery agent discovers Virtual Device Context (VDC) information from Cisco Nexus 7000 series devices. Each VDC instance is hosted under a hypervisor entity contained by the physical device.

For more information, see the IBM Tivoli Network Manager IP Edition Discovery Guide.

Link Aggregation Group (LAG) discovery
Network Manager 4.1.1 Fix Pack 1 can discover and model link aggregation information from devices running the Alcatel 5620 SAM EMS. Aggregated links are modeled in the NCIM topology database and shown within the device containment in the Structure Browser. Additionally, a Service Affected Event (SAE) is generated if any port that participates in an aggregated link has an alert of severity 5 or higher.

For more information, see the IBM Tivoli Network Manager IP Edition Discovery Guide.

Checkpoint firewall support (support for centre/virtual environment)
The new CheckpointContext Perl agent queries CheckPoint VSX firewalls to retrieve context data. Retrieving context data allows other
context-sensitive discovery agents to retrieve interface and connectivity data from the appropriate contexts. The MACFromTDWDatabase discovery agent queries the Tivoli Data Warehouse where access parameters for the database are provided in DbLogins.cfg. The agent retrieves the mapping of MAC address to IP address for each server. The CheckpointVSX agent queries CheckPoint VSX firewalls and retrieves context-sensitive data. The agent builds interface containment from the data.

**Note:** Before using the Checkpoint agents, download the required MIB files:

2. Open the file in a text editor and copy the MIB definition, which starts SNMP-VIEW-BASED-ACM-MIB DEFINITIONS ::= BEGIN and ends END into a new file named SNMP-VIEW-BASED-ACM-MIB.mib in the $NCHOME/precision/mibs/ directory.
3. Download the chkpnt.mib from http://www.checkpoint.com/, if it is not already present on the Checkpoint device.
4. Open the file in a text editor and copy the MIB definition, which starts CHECKPOINT-MIB DEFINITIONS ::= BEGIN and ends ... END into a new file named CHECKPOINT-MIB.mib in the $NCHOME/precision/mibs/ directory.
5. Run the ncp_mib process to update the MIB data.

For more information, see the IBM Tivoli Network Manager IP Edition Discovery Guide.

**Fix pack rollback support**

You can now uninstall a Network Manager fix pack and this will roll back all of the functionality installed as part of that fix pack.

For more information, see the IBM Tivoli Network Manager IP Edition Installation and Configuration Guide.

**Binding a domain to a network interface controller (NIC)**

This new feature provides the ability to define which NIC Network Manager uses to access the network. This enables you to bind different customer domains to different NICs and thereby resolve issues with device addressing.

For more information, see the IBM Tivoli Network Manager IP Edition Installation and Configuration Guide.

**Discovery option for sub-interfaces to inherit unmanaged status**

You can now configure discovery to set sub-interfaces to unmanaged when the parent interface is marked as unmanaged by the PopulateDNCIM_ManagedStatus.stch stitcher.

For more information, see the IBM Tivoli Network Manager IP Edition Discovery Guide.

**Improved Oracle RAC connection capabilities**

You can now define a custom Oracle connection string. This connection string can be used for connecting to an Oracle RAC cluster, or for other purposes.

For more information, see the IBM Tivoli Network Manager IP Edition Installation and Configuration Guide.
LTE support added for the Discovery Library Adapter (DLA)

You can now use the DLA to collect data on LTE network resources for export to other systems, such as CCMD, TADDM, and SCCD.

For more information, see the IBM Tivoli Network Manager IP Edition Installation and Configuration Guide.

Improved polling capabilities

Polling capabilities have been enhanced, as follows:

- The command-line options for the itnm_poller.pl script have been enhanced. The command-line options for the itnm_poller.pl script have been enhanced. A new -monitors option has been provided. This option enables you to retrieve polling status of a poll policy without having to know if it is a chassis or interface poll. In addition the -status and -interface options have been enhanced to display time in human readable format.

- You now have the ability to include actual poll data value from an expression in Alert Threshold details. You can include actual poll data value from an expression in a Basic Threshold poll definition in the alert threshold details. This enables the actual value from an expression to be displayed in the event summary displayed to Operations staff and thereby helps operators to determine the extent to which the threshold was violated.

For more information, see the IBM Tivoli Network Manager IP Edition Event Management Guide and the IBM Tivoli Network Manager IP Edition Administration Guide.

Improved root-cause analysis (RCA) capabilities

RCA capabilities have been enhanced, as follows:

- The inject_fake_events.pl Perl script has been added. This script provides the ability to generate fake events in order to verify RCA.

- The RCA path tool has been added. This tool provides the ability to check whether a topological path between network devices exists for the purposes of topological correlation using the RCA path tool.

For more information, see the IBM Tivoli Network Manager IP Edition Event Management Guide and the IBM Tivoli Network Manager IP Edition Administration Guide.

New end-to-end search feature

You can now launch SmartCloud Analytics - Log Analysis from Network Views. The ability to perform this launch is not enabled by default. In order to enable this feature you must follow the instructions at the following Knowledge Center location: http://www-01.ibm.com/support/knowledgecenter/SSTPTP_1.3.0/com.ibm.netcool_ops.doc_1.3.0/soc/topo_search/task/soc_ts_configuretoposearch.html

Tivoli Common Reporting 3.1 is available for Netcool Operations Insight users

Netcool Operations Insight users can point their Network Manager server at an existing Tivoli Common Reporting 3.1 installation that is installed with Jazz for Service Management. For more information, see http://www-01.ibm.com/support/knowledgecenter/SSTPTP_1.3.0/com.ibm.netcool_ops.doc_1.3.0/omn_nmip/task/nnm_noi_installtcr.html

GA product

Long Term Evolution® (LTE) network discovery and visualization

Network Manager 4.1.1 can discover and visualize LTE networks.
Discovery of LTE networks is by means of collectors, notably the NetAct XML Interface for Configuration Management collector and the Alcatel5620SamSoapFindToFile collector. In order to visualize LTE networks and devices, you can add an LTE network view hierarchy to your network view tree. The default LTE network view hierarchy includes a drill-down into LTE devices, a geographical view of LTE devices, and user and control plane topology views.

For more information, see the following documents:
- IBM Tivoli Network Manager IP Edition Discovery Guide
- IBM Tivoli Network Manager IP Edition Network Visualization Setup Guide
- IBM Tivoli Network Manager IP Edition Topology Database Reference

**Full 64-bit support**

Network Manager 4.1.1 is a 64-bit application, and can only be installed in a 64-bit environment. However, during the installation procedure, you can install the Network Manager GUI components into an existing 32-bit Tivoli Integrated Portal and Web GUI environment.

In previous versions of Network Manager, the maximum amount of memory that could be allocated to an individual process was typically 4GB. On 64-bit Linux, the maximum memory allowed for a process is not arbitrarily limited but depends on how much memory is installed on the server.

However, if you install Network Manager GUI components into an existing 32-bit Tivoli Integrated Portal environment, then the maximum memory available to the Network Manager GUI components is typically 4GB.

**SNMP interface filtering**

You can now exclude interfaces that you are not interested in by applying SNMP interface filtering to the discovery. SNMP interface filtering can help you to discover virtual devices faster.

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

**Browser support**

Network Manager V4.1.1 introduces support for Internet Explorer 10 and Mozilla Firefox 24 Extended Support Release (ESR).

For more information, see the *IBM Tivoli Network Manager IP Edition Installation and Configuration Guide*.

**Reduced platform and database support**

Network Manager 4.1.1 is supported only on the following systems:
- Red Hat Enterprise Linux version 5.

Network Manager 4.1.1 is supported only on 64-bit hardware.

IBM DB2® version 10.1 is the default topology database for Network Manager 4.1.1. Network Manager 4.1.1 also supports IBM DB2 versions 9.7 and 10.5, and Oracle Database version 11g, Standard or Enterprise Edition.

For more information, see the *IBM Tivoli Network Manager IP Edition Installation and Configuration Guide*.
Oracle RAC support in failover setup

If you use Oracle for your NCIM topology database, you can set up a failover configuration for the database using the Oracle Real Application Clusters (RAC) feature.

For more information, see the IBM Tivoli Network Manager IP Edition Installation and Configuration Guide.

Entity smart search

Network Manager V4.1.1 includes the entity smart search feature which you can use to search for an entity across all Network Manager domains or across a custom NCIM database table in an extended NCIM database.

For more information, see the IBM Tivoli Network Manager IP Edition Network Troubleshooting Guide.

Enhanced polling functions

The polling functionality is enhanced as follows.

- The polling architecture is changed to improve the start time of\n  ncp_poller processes. The changes mean that when a poller starts, it no longer copies data for the polled entities from the NCIM database to the NCPOLLDATA database. As a result, the number of SQL WRITE operations that are associated with the startup of a poller is reduced. The NCPOLLDATA database is now updated with data from the NCIM database by the MODEL topology manager.

- Poller metrics are introduced that can be output as bar charts on the command-line interface. Use the bar charts to monitor the health of individual pollers and avoid poller failures. You can know ahead of time whether a poller is running near capacity.

- New configuration options are introduced that you can use to set thresholds and limits on the size of the queue of poll data for writing to the NCPOLLDATA database. These options help to prevent pollers running out of memory if the queue becomes too large, for example if a database outage occurs or the polling load becomes too great. If the threshold is exceeded, you are alerted by a message in the poller log file. If the limit is exceeded, you are alerted in the log and in the Active Event List (AEL). Then, the poller acts to prevent further increases by writing the queued data to a separate file and removing the data from the queue. The queued data is written as SQL INSERT statements and can be manually written to the NCPOLLDATA database.

- The messaging in the $NCHOME/log/precision/\ncp_poller.SnmpPoller.DOMAIN.trace file is improved and specifies the varbind for SNMP noSuchName messages. The varbind helps you to identify which variable or, if the polled data is from a table, which ifIndex is causing problems. The following example shows the new message format:

  ncp_poller.SnmpPoller.ITNM39.trace:2013-10-23 10:03:44 WARNING\nbatchitem.cc(2024): received message from 192.168.120.15:161 with error\n(error-status = NOSUCHNAME error-index = 23 \nvbsize = 29 vb = 1.3.6.1.2.1.2.2.1.21.37) - handling tooBig,\nso will not attempt to fix request

Where the varbind is specified in vb = 1.3.6.1.2.1.2.2.1.21.37.

For more information, see the IBM Tivoli Network Manager IP Edition Event Management Guide.
New information delivery: Knowledge Center

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 GUls from different Tivoli products, including Network Manager. The GUls represented in the following figure, including the topology visualization GUls and the event visualization GUls all run within the framework of the Tivoli Integrated Portal.
  - The topology visualization GUls 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 GUls 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.

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.

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, 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.

**Related concepts:**
- [“Network layer” on page 16](#)
  The network layer consists of network discovery and polling tools.
- [“Data layer” on page 28](#)
  The data layer consists of topology storage, event storage, performance reporting data storage, and root-cause analysis tools.
- [“Visualization layer” on page 41](#)
  This layer consists of topology visualization and event visualization tools.

**Network Manager data flow**

Use this information to understand how the components of Network Manager fit together.

The following figure shows the main areas of functionality within Network Manager and depicts how data flows between them.
Network administrators configure and run a discovery

Batch-mode discoveries can be run on demand or can be scheduled. In batch mode, the whole network can be discovered (this is known as full discovery), or just a single subnet or device (partial discovery). Essential discovery configuration information consists of device seeds, network scope, and device access details such as SNMP community strings.

Data is gathered from the network

Devices are found on the network by the Ping finder, File finder, or collector framework when interrogating Element Management Systems (EMS) such as the Alcatel 5620 SAM. Discovery agents are invoked when devices of specific types are found on the network by the Ping or File finder. The agents request connectivity information from devices that the finders have discovered. Discovery agents interrogate network devices for information using methods such as ICMP, SNMP, SSH, and TELNET.

Network topology is created

Data gathered from devices is processed and a network topology is created and stored in a discovery database.

Network devices are classified by type and the network topology is stored

Following discovery, topology data is classified according to device type,
and the topology data is stored in the NCIM topology database. Topology data is also made available to the Event Gateway, ncp_g_event, which uses this data to enrich events and to the Root-cause analysis plug-in to the Event Gateway, which uses this data to identify root-cause events.

5 Network administrators configure device polling
Network Manager has a default set of polling policies. These polling policies include simple device or interface pings and more complex threshold polls against specific MIB variables. Network administrators can configure polling policies to poll a more restricted set of devices, to change polling frequency, to change the data collected, and to make other custom changes.

6 Network Manager polls the network
Network Manager polls the network based on default and configured polls.

7 Network Manager converts relevant poll results into Tivoli Netcool/OMNIbus events and sends them to Tivoli Netcool/OMNIbus
Network Manager converts the results of relevant polls into Tivoli Netcool/OMNIbus events, and sends these events to the Tivoli Netcool/OMNIbus ObjectServer, which stores the events. Poll results that are converted into Tivoli Netcool/OMNIbus events include those polls where the response indicates a device or other network failure of some sort, such as a threshold violation or an ICMP ping fail.

8 Tivoli Netcool/OMNIbus and other event sources populate the ObjectServer
Tivoli Netcool/OMNIbus probes, and, potentially, other network event sources, populate the ObjectServer with network events.

9 Events are enriched with topology data
Events are passed to the Event Gateway, ncp_g_event, where they are enriched with topology data. Some events are passed directly back to the ObjectServer. Event Gateway plug-ins subscribe to certain types of events. Based on these subscriptions, events are passed to the Event Gateway plug-ins.

10 Event Gateway plug-ins perform root-cause analysis and other actions based on events
Event Gateway plug-ins perform various actions based on events received from the Event Gateway. For example, the RCA plug-in performs further event enrichment. The SAE plug-in generates synthetic service-affected (SAE) events based on events received. Other plug-ins take other actions based on the occurrence of certain events; for example, the Failover plug-in initiates failover based on the occurrence of Network Manager health check events. Plug-ins pass enriched events back to the ObjectServer.

11 Network Manager gathers device performance data on demand
At any time a network administrator can set up polling of specific SNMP and ICMP data on one or more network devices. Network Manager gathers this data and stores it in the NCPOLLDATA historical polled data database.

12 Topology visualization software accesses the NCIM database
The topology visualization web application, running within the Tivoli Integrated Portal application, accesses the topology within the NCIM database. Network operators can now log into the Tivoli Integrated Portal and view their network devices and components using the Network Manager topology visualization GUIs, including multi-portlet views, such as:
as the Fault-Finding View and the Network Health View, and single-portlet views, such as the Network Hop View, Network Views, and Path View.

13 Event visualization software accesses the ObjectServer
The Tivoli Netcool/OMNIbus Web GUI (known as Netcool/Webtop in versions 2.2 and below) requests the latest set of events from the ObjectServer. Any changes network operators make to these events using the Web GUI are sent back to the ObjectServer. Network operators can now log into the Tivoli Integrated Portal and view events using the Active Event List, Lightweight Event List and Event Dashboard.

14 Event information is requested
The Topology Visualization Web application requests event information from the Tivoli Netcool/OMNIbus Web GUI application.

15 Report data for performance reports is retrieved from the NCPOLLDATA historical polled data database
Network operators log into the Tivoli Integrated Portal, access Tivoli Common Reporting, and run performance data and other reports. The report data for performance reports is retrieved from the NCPOLLDATA historical polled data database.

Related concepts:
“Network layer” on page 16
The network layer consists of network discovery and polling tools.
“Data layer” on page 28
The data layer consists of topology storage, event storage, performance reporting data storage, and root-cause analysis tools.
“Visualization layer” on page 41
This layer consists of topology visualization and event visualization tools.

Integration with other products
Network Manager is designed to be installed with Tivoli Netcool/OMNIbus. You can also integrate with other Tivoli products, such as IBM Tivoli Business Service Manager, Tivoli Application Dependency Discovery Manager (TADDM), IBM Tivoli Monitoring (ITM), and IBM Systems Director.

Standard Network Manager installation
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. The Tivoli Netcool/OMNIbus Web GUI was known as Netcool/Webtop in versions 2.2 and below. This standard installation provides the following capabilities:
• Network discovery
• Storage of network topology information using the NCIM database
• Visualization of network topology using the Network Views, Hop View, and Structure Browser GUls
• Various network diagnosis facilities using tools including the SNMP MIB Browser, SNMP MIB Grapher, and other dedicated diagnostic tools
• Active monitoring of network availability. Alerts generated based on availability issues are forwarded to Tivoli Netcool/OMNIbus
• Monitoring of SNMP MIB values and generation of alerts when predefined thresholds are crossed
• Correlation of events with network topology information to provide event enrichment and root-cause analysis features
• Visualization of events in the network visualization tools, including the Network Views, Hop View, and Structure Browser GUIs
• Ability to generate service-affected event (SAEs). An SAE is an alert that warns operators that a critical customer service has been affected by one or more events.
• Ready-to-use reports, combining event information with network topology information

**Integration with other Tivoli products**

Network Manager is integrated with a continually increasing set of Tivoli products. Network Manager can be integrated with the following Tivoli products:

**IBM Tivoli Business Service Manager (TBSM)**

TBSM provides the following functionality:

- Populates the business service model using network information discovered by several applications, including Network Manager.
- Maps events from multiple sources to the resources in IBM Tivoli Business Service Manager, including those resources discovered in Network Manager. In this case resources refers to devices, interfaces, and so on.
- Troubleshoots faults in the infrastructure by launching in context from the IBM Tivoli Business Service Manager service views to one of several Network Manager topology views.

**Tivoli Application Dependency Discovery Manager (TADDM)**

TADDM provides the following functionality:

- Imports network topology information discovered by Network Manager into TADDM to complete the TADDM view of application-to-network dependencies.
- The Network Manager inventory report is available in TADDM.
- Allows TADDM users to launch in context into Network Views to investigate which problems in the infrastructure might be affecting application performance.
- Network Manager can also launch in context into TADDM to display the TADDM Change History view and the TADDM Details view associated with the devices.

**IBM Tivoli Monitoring (ITM)**

ITM provides the following functionality:

- Monitors the health of the Network Manager application and displays key metrics and situations that help administrators monitor the health and status of Network Manager.
- Can be used to monitor resources within the Network Manager network.
- Network Manager can store performance data collected from network devices in Tivoli Data Warehouse for centralized management of warehouse data. This data can be accessed from other products such as Tivoli Performance Analyzer and ITM.
- It is possible to launch from ITM directly into Network Manager, although this is not an in-context launch.
IBM Systems Director
IBM Systems Director provides consolidated views of your managed systems and a set of tasks for system management including discovery, inventory, configuration, system health, monitoring, updates, event notification and automation across managed systems. After setting up the integration, you can open IBM Systems Director tasks from the Network Manager GUI using the right-click menu.

IBM Netcool Configuration Manager (ITNCM)
ITNCM provides extensive configuration management capabilities for network devices, as well as network policy thresholding capabilities. Netcool Configuration Manager can be integrated with Network Manager and Tivoli Netcool/OMNIbus to provide more powerful diagnostic functionality to network operators.

IBM Tivoli Common Reporting (TCR)
The integration with TCR provides ready-to-use reports, including reports on network configuration information.

Network layer
The network layer consists of network discovery and polling tools.

About discovery
Network administrators configure and run full and partial discoveries to generate a network topology.

You can keep the discovered topology up to date by scheduling regular discoveries, configuring automatic rediscovery, and manually rediscovering devices.

If new subnets or new devices are added to your network, you can use partial discovery to discover just those subnets and devices.

If you have a very large network, then you can break the discovery of your network into different network domains. Partitioning your network into domains allows you to discover your network in sections. Reasons for partitioning your network include the following:
- Scalability: Your network might be too big to be discovered in one piece.
- Geography: You might want to break the network into geographical regions, and make each region correspond to a domain.
- Logical network boundaries: You might want to discover and manage the network based on particular network boundaries.

You can also discover links between devices in different domains, and create an aggregated domain, by configuring and running cross-domain discoveries.

Network Manager can also collect network topology data from Element Management Systems (EMSs). Once data is collected from EMSs, it is integrated with other data collected during the discovery.

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

Related concepts:
Network Manager can be configured to collect topology data from Element Management Systems (EMSs) and integrate this data into the discovered topology.

**Discovery architecture**

Use this information to understand how the components of the discovery process work together to perform a full discovery of the network.

During a full discovery, the Discovery Engine, ncp_disco, detects the existence of devices on the network and queries the devices for inventory and connectivity information. This information is subsequently processed or 'stitched' together to generate a connectivity or topology model.

After a full discovery, the system classifies devices based on a predefined Active Object Class (AOC) hierarchy. The network topology is stored in the topology database and can be visualized by network operators as topology maps, customized to show specific devices or specific device groupings such as subnets and VLANs.

The following figure shows how the components of the discovery process work together to perform a full discovery of the network.

---

1. An AOC classifies devices based on vendor, type, and model family.
Seed devices are discovered
Based on discovery configuration seed settings, the Discovery engine, ncp_disco, sends finders onto the network to find seed devices. The finders discover the existence of devices but do not retrieve connectivity information.

Agents are invoked to identify devices details, connected devices, and device type information
As devices are found, discovery agents are invoked to retrieve details of discovered devices, device connectivity information, and information specific to device types. There are multiple discovery agents to support the wide variety of network devices. Discovery agents interrogate network devices for information using methods such as ICMP, SNMP, SSH, and TELNET.

Connected devices are identified and device details and device type information are retrieved
Discovery agents do not have direct interaction with the network, but instead retrieve information from network devices using the Helper Server. The Helper Server manages the helpers and stores the information that is retrieved from the network. The helpers retrieve information from the
network on behalf of the discovery agents. Helpers also translate agent queries into the appropriate network protocol and make requests to the devices.

4 **Connected devices are discovered**

Connected IP addresses identified by the discovery agents are fed back into the finders, which discover the existence of these connected devices. Discovery agents are then invoked for these connected devices (step 2) and the feedback process repeats itself until the discovery encounters the boundary delineated by the discovery scope settings.

**Note:** The Discovery communicates with the Active Object Class (AOC) manager, ncp_class, to classify all the devices in the topology based on vendor, type, and model family. It uses the sysObjectID value held in the device MIB and assigns the device a particular classification based on logic held within active object class files.

5 **Network topology is 'stitched' together**

Once all devices and device connectivity has been discovered, discovery processing stitchers are invoked. These stitchers 'stitch' together the data gathered by the agents to generate a connectivity or topology model.

6 **Discovered topology is processed by Topology manager, ncp_model**

The discovery sends the topology model to Topology manager, ncp_model. The Topology manager processes the discovered topology. For example, it processes linger time for devices and makes appropriate modifications. The topology is also stored in ncp_model.

7 **Network topology is stored**

The Topology manager, ncp_model, sends the topology to the Topology database, NCIM. The topology data in NCIM can be queried using SQL.

**About cross-domain discovery**

Cross-domain discovery can be configured to join two or more discovered domains together.

For performance or operational reasons, networks are often discovered in sections, known as discovery domains. For example, if your network is so large that discovering it in one discovery takes too long, you might choose to split network discovery into different domains.

Discovering the network in domains can be more convenient and faster. You can also choose to have different configuration options for different domains. For example, each domain has its own poll policies. However, there are disadvantages to discovering the network in pieces. If a device in domain A is connected to a device in domain B, this connection is not represented in the topology database or in the GUI. Domains must be viewed separately.

If you want to visualize multiple domains linked together in one network view, you must enable, configure, and run cross-domain discoveries. Connections between devices in different domains are found and added to the topology.

When all discovered domains have been aggregated, Network Views can be composed of devices from all domains. In the Network Hop View, searches for devices can span domains.

**Note:** Cross-domain network views can not be polled; only network views from individual domains can be polled.
Considerations for splitting the network into domains

Links between devices in different domains are not as easy to discover as links between devices within one domain.

It is important to scope your discovery domains to ensure the minimum of links between domains. For example, you would not normally split the network such that highly connected switches were in different domains. Natural splits for domains are often along geographical lines.

Restriction:

However you split your network, you must ensure that any given device appears in only one domain. That is, the discovery domains must not overlap if you want to join them together using cross-domain discovery.

Discovery tasks

Network administrators configure and run discoveries in order to generate a network topology. Network operators can rediscover specific devices and thereby refresh device data by simply right-clicking a device in a topology map.

Discovery user tasks fall into the following two categories:

- Configuring discoveries
- Running discoveries

Configuring full and partial discoveries

Network administrators configure network discoveries by specifying a wide range of discovery parameters. These parameters include the following:

- Discovery scope: the subnet ranges to include in or exclude from a discovery
- Discovery seeds: devices or subnets to discover first. You can seed a limited set of devices and subnets and use the discovery feedback mechanism to discover all devices connected to your seeds. You can also specify discovery seeds using other mechanisms such as a seed file
- Device access: SNMP community strings, and Telnet or SSH access parameters to enable Network Manager to access your network devices

Administrators can specify discovery parameters using the Discovery Configuration GUI or directly from the command line.

Running full and partial discoveries

Network administrators run discoveries to discover the entire network and have the ability to schedule regular full discoveries to keep the network topology up to date.

Administrators can also run partial discoveries to add new subnets or new devices to the topology. Administrators can run full and partial discoveries using the Discovery Status GUI or directly from the command line. Network Manager also provides tools to schedule regular discoveries.
**About EMS-based discovery**

Network Manager can be configured to collect topology data from Element Management Systems (EMSs) and integrate this data into the discovered topology.

For more information about how to configure an EMS discovery and for detailed process information about EMS discovery, see the *IBM Tivoli Network Manager IP Edition Discovery Guide*.

**Related concepts:**

"About discovery" on page 16

Network administrators configure and run full and partial discoveries to generate a network topology.

**Overview of EMS-based discovery architecture**

Use this information to understand how Network Manager collects topology data from Element Management Systems (EMSs).

The following figure shows how Network Manager EMS-based discovery process retrieves topology data from EMSs.

---

**Figure 4. EMS-based high-level architecture**

1. **Collector finder queries EMS collector for device list**
   - Using the Collector finder, the Discovery subsystem queries the collector to
obtain a list of devices managed by the EMS. In the case of a partial rediscovery, the discovery may query for a single device or subnet only.

2 **EMS is queried**
   The collector queries the EMS for the list of devices.

3 **EMS responds**
   EMS responds with list of managed devices.

4 **Collector provides list of devices to the discovery process**
   Collector provides the list of devices.

5 **Agents requests detailed information from the EMS collector**
   Using a number of specialized collector discovery agents at different times during the discovery, the Discovery subsystem queries the collector for basic and detailed information about each of the devices in the list. Detailed information requested includes inventory information, layer 1, layer 2, and layer 3 connection details, and VPN information.

6 **Collector provides detailed device information**
   Collector responds by providing basic and detailed information as this is requested.

**Details of EMS-based discovery architecture**
Use this information to understand how the components of EMS-based discovery work together.

The following figure shows how the components of the Network Manager EMS-based discovery process work together to retrieve topology data from EMSs.
Collector finder queries EMS collector for device list

The Collector finder reads the collector host seeds from a seed table in the collectorFinder database. It then queries the collectors specified in this table to get a list of devices managed by the EMSs associated with each collector.

Collector finder sends data to stitchers

The Collector finder sends the list of devices managed by the EMSs to the data collector collection stitchers.

Collector agents are invoked

The data collection stitchers invoke the Collector agents.

Collector agents retrieve information about the devices from the collectors

The Collector agents retrieve basic and detailed information about the devices on the collector. Each agent makes use of the Collector helper to retrieve this information. The Collector helper, ncp_dh_xmlrpc, enables Network Manager to communicate with the collectors using the XML-RPC interface. The Collector agents retrieve the following information:

CollectorDetails agent

Retrieves basic information about the devices on the collector, including sysObjectId, sysDescr, and naming data.
CollectorInventory agent
Retrieves local neighbor, entity and associated address data for each of the devices on the collector.

CollectorLayer1 agent
Retrieves layer 1 and microwave connectivity information for the devices on the collector.

CollectorLayer2 agent
Retrieves layer 2 connectivity information for the devices on the collector.

CollectorLayer3 agent
Retrieves layer 3 connectivity information for the devices on the collector.

CollectorLTE agent
Retrieves LTE-specific entity information for the devices on the collector.

CollectorRan agent
Retrieves radio access network (RAN) information for the devices on the collector.

CollectorVpn agent
Retrieves layer 2 and layer 3 VPN data for the devices on the collector.

Stitchers are invoked
Collector agent data is passed to the data processing stitchers.

EMS-based discovery tasks
Network administrators configure an EMS-based discovery by configuring and starting the different elements of the EMS-based discovery process. Network Manager ships with a number of ready-to-use collectors, each of which processes data from a different EMS, and with Java™ and Perl modules to support the development of custom collectors. Network Manager also provides CSV collectors, which process input data from CSV files.

These are the EMS-based discovery tasks:
• Creating custom collectors
• Configuring collectors
• Starting collectors
• Seeding the Collector finder
• Enabling collector agents

Creating custom collectors
Network Manager ships with ready-to-use collectors, each of which processes data from a different EMS, and with Java and Perl modules to support the development of custom collectors. Network administrators can develop custom collectors to process data from other EMSs using the Java or Perl modules provided, or using other languages, as documented in the Tivoli Field Guide: EMS Collector Developer Guide.
Configuring collectors

You must configure each collector so that it is able to pass data requests between Network Manager and the associated data source (for example, EMS or CSV file). Configuring the collector depends on the type of data source:

- **EMS**: specify the hostname, port, username and password of the EMS.
- **CSV file**: specify details of the CSV files and how to parse them.

You must also instruct the collector which port to listen on for XML-RPC requests from Network Manager. This is typically a one-time setup task required when a new collector is added to your Network Manager installation.

Starting collectors

Before discovery starts, all the collectors must be running.

Seeding the Collector finder

In order to enable Network Manager to find the collectors, you must seed the Collector finder. This task involves specifying for each collector, the hostname of the device on which the collector is running and the port on that device on which the collector is listening. If a collector is running on the same host as Network Manager, then you need only specify the port. This is typically a one-time setup task required when a new collector is added to your Network Manager installation.

Enabling collector agents

By default, the Collector agents are not enabled. You must enable these agents if you are running a discovery that includes collector-based discovery.

About polling

Network polling determines whether a network device is up or down, whether it has exceeded key performance parameters, and identifies inter-device link faults. If a poll fails, Network Manager generates a device alert, which operators can view in the **Active Event List (AEL)**.

Network Manager polling policies poll network devices at regular intervals, and if something does not match the polling criteria, an event is generated. For example, a polling policy might retrieve the CPU utilization of a device at a periodic interval. If the CPU utilization exceeds a predefined threshold, then an event is generated.

A set of polling policies are enabled by default. A policy specifies:

- A set of devices to poll
- Poll definitions, including threshold triggers
- Frequency of the polling
- Whether to store the data for historical reporting

Chapter 1. About Network Manager 25
**Polling architecture**

Use this information to understand how the components of the Network Manager polling process work together to poll network devices.

The Polling engine, ncp_poller, is the component that controls network polling. The Polling engine uses active polling operations to gather data. This data is used to trigger alerts if appropriate, and can optionally be stored in the NCPOLLDATA database for later analysis in the performance reports.

**Note:** By default, there is a single instance of ncp_poller running on the Network Manager server, with the service name of ncp_poller(default). If you set up multiple pollers, then there will be multiple instances of ncp_poller running on the Network Manager server, each with a different service name; for example Ping Poller and snmpPoller.

After you create polls and poll definitions, they are saved to the NCMONITOR schema within the NCIM database. The ncp_poller process reads the poll definitions from the NCMONITOR schema. The following figure shows how ncp_poller interacts with other Network Manager components and the ObjectServer. Note that the figure does not include the historical data storage feature.

![Polling data flow diagram](image-url)
Polling targets are retrieved
Network Manager has a default set of polling policies. These polling policies include simple device or interface pings and more complex threshold polls against specific MIB variables and are stored in the NCIM database. The Polling engine, ncp_poller, retrieves polling policies from the NCIM database and determines the target devices to poll, and how to poll them. Network administrators can log into the Polling GUI to configure these polling policies. For example, they can configure the system to poll a more restricted set of devices, to change polling frequency, to change the data collected.

Network devices are polled
The Polling engine, ncp_poller, polls network devices based on the polling policies defined in Network Manager. Devices are polled based on polling frequency and based on the set of devices specified in the polling policy.

Relevant poll results are converted to Tivoli Netcool/OMNibus events
Poll results that are converted into Tivoli Netcool/OMNibus events include only those polls where the response indicates a device or other network failure of some sort, such as a threshold violation or an ICMP ping fail. The Polling engine, ncp_poller, sends these poll results to the Probe for Tivoli Netcool/OMNibus. This probe maps the poll results into Tivoli Netcool/OMNibus event format.

Optionally polled data is stored in the NCPOLLLDATA database
Depending on poll policy configuration settings, ncp_poller sends collected data to the NCPOLLLDATA database, where it is stored and can be retrieved using reports. The NCPOLLLDATA database can be hosted on the NCIM database, or, as shown in the figure, can be placed on a remote machine and hosted in a separate data storage system. An example of a data storage system that can be used for this purpose is the Tivoli Data Warehouse (TDW).

Events sent to ObjectServer
Probe for Tivoli Netcool/OMNibus sends the converted event data to the Tivoli Netcool/OMNibus ObjectServer. Tivoli Netcool/OMNibus and other probes also sent events to the ObjectServer.

Polling tasks
Network Manager has a default set of polling policies. These polling policies include simple device or interface pings and more complex threshold polls against specific MIB variables. Network administrators can configure polling policies to poll a more restricted set of devices, to change polling frequency, to change the data collected, and to make other custom changes. Administrators can also enable and disable polling policies. They can also set a device or component to an unmanaged state to suspend it from Network Manager network polling.

Polling tasks fall into the following categories:
- Configuring polling policies
- Configuring the multiple poller feature
- Suspending polling of specified devices

Configuring polling policies
Network administrators can customize how Network Manager polls discovered devices by modifying polling policy parameters. These parameters include the following:
• Polling frequency
• Whether to store historical data for the poll to be used in performance reporting
• The set of devices or interfaces to poll. Devices can be selected based on device class. In addition, the network administrator can create an SQL filter to poll a more restricted set of devices or interfaces.
• Which data to collect from the polled devices. The data to poll for is defined using poll definitions.

Network administrators configure ping, link-state, and threshold polls using poll definitions. The poll definition specifies whether the polling policy should simply ping a device or interface, or should apply a threshold value to a MIB variable within the device MIB. A threshold violation causes Network Manager to generate a network event, which network operators can view in the Active Event List (AEL).

Configuring multiple polling engines

Network Manager provides a mechanism to help distribute the load due to polling, by configuring multiple polling engines. If the default Polling engine cannot handle the polling demands for your network, then the network administrator can configure multiple polling engines on the Network Manager server. Administration tasks here include the following:
• Monitoring the health of the Polling engine using the Network Manager Health views in IBM Tivoli Monitoring
• Deploying additional pollers on the Network Manager server
• Removing pollers if these are no longer being used to poll the network

Suspending polling of specified devices

Network administrators can suspend polling of devices and specific interface types. Administrators can specify files containing lists of devices for which polling is to be suspended. Network operators can suspend polling of individual devices.

Data layer

The data layer consists of topology storage, event storage, performance reporting data storage, and root-cause analysis tools.

About topology storage

Topology data is stored in the Network Connectivity and Inventory Model (NCIM) database. Network Manager reports also use the NCIM database.

The NCIM database is a relational database that consolidates topology data from Network Manager (OSI layers 1, 2 and 3).
**Topology storage architecture**

Use this information to understand how the NCIM database interacts with other components of Network Manager.

The following figure shows how data flows through the NCIM database.

---

**Figure 7. Topology storage data flow**

1. **Topology is transferred from the Topology manager, ncp_model**
   Following a discovery, the Topology manager, ncp_model, sends the topology to the Network Connectivity and Inventory (NCIM) database.

2. **Network administrators query the NCIM database**
   Network administrators query the NCIM database using SQL queries to retrieve detailed topology data.

3. **Topology visualization web applications access the topology**
   The topology visualization web applications, running within the Tivoli Integrated Portal, access the topology. This enables the topology visualization GUIs, Hop View, Network Views, and Structure Browser, to display network topology and device structure.
**Topology storage tasks**

Network administrators query the NCIM database to programmatically retrieve topology information. Administrators can support topology enrichment from third-party data sources by adding tables and fields to the topology database.

Network administrators can also access visual representations of the network topology by viewing topology maps in the Hop View and Network Views, and by using the Structure Browser to explore details about a device.

**Querying the database**

Network administrators can write SQL queries to retrieve topology information from the NCIM database. The administrator can retrieve any topology information, including the following:

- Network domain information; for example, list all devices in a domain
- Device information; for example, list all devices with corresponding class name
- Containment information; for example, list all components in a device
- Port and interface information; for example, list all interfaces that have specific attributes
- Connectivity information; for example, identify all connections between routers
- Hosted services; for example, list all chassis devices hosting OSPF services
- Collection information; for example, list all devices in a specified VPN
- Enumeration information; for example, identify all hardware manufacturers listed in the database

**Modifying the database to support topology enrichment**

It is possible to customize discovery to retrieve and store data about the discovered devices from third-party data sources. This is known as topology enrichment. For example, a discovery stitcher could be created to retrieve customer information related to devices from a third-party inventory database. This would enable network operators to see the customer associated with a given device or network event. Network administrators can support topology enrichment from third-party data sources by adding tables and fields to the topology database.

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

**About root cause analysis and event enrichment**

Root cause analysis (RCA) and event enrichment are conceptually distinct topics; however, within the Network Manager architecture, RCA is a plug-in to the Event Gateway, the process that manages event enrichment.
About event enrichment
The more information that is contained within a network event, the easier it is to
determine what caused the event and who it is impacting. Event enrichment is the
process of adding topology information to the event.

Network Manager gathers a lot of information about devices and topology in the
network. You can use this information to enrich events with data, such as the
system location, contact information, and product serial number.

The event enrichment process is known as the Event Gateway, ncp_g_event. The
Event Gateway has a set of associated plug-ins, which are modular processes that
receive enriched events from the Event Gateway and perform further event
enrichment or take other action on these events. One of the Event Gateway
plug-ins is the RCA plug-in.

Event enrichment tasks
Network Manager adds a default set of topology information to events. Network
administrators can configure event enrichment to enrich events with extra topology
data.

Event Gateway configuration
Examples of event enrichment include the following
• Enriching an event with main node location: administrators can configure event
  enrichment so that the location of the main node associated with an event is
  added to a field in the event.
• Enriching an event with interface name: administrators can configure event
  enrichment so that for all interface events, the name of the interface on which
  the event occurred is added to a field in the event.

Other examples of useful information that can be used to enrich events include the
following:
• System contact
• Interface description
• Interface alias

To configure event enrichment, network administrators must first create new fields
to hold the extra topology data in the Tivoli Netcool/OMNIbus ObjectServer. The
next step is to configure the Event Gateway config database to pass the relevant
topology data to the ObjectServer.

For information on how to add new columns to the Tivoli Netcool/OMNIbus
ObjectServer alerts.status column, see the IBM Tivoli Netcool/OMNIbus
Administration Guide.

Event Gateway plug-in configuration
Event Gateway plug-ins are modular processes that receive enriched events from
the Event Gateway and perform further event enrichment or take other action on
these events. One of the Event Gateway plug-ins is the RCA plug-in. Event
Gateway plug-in configuration tasks include the following:
• Enabling and disabling plugins
• Modifying event map subscriptions This changes the type of events that each
  plug-in acts on.
• Setting plug-in configuration parameters You can set optional configuration parameters for the Event Gateway plug-ins.

For more information on Event Gateway and Event Gateway plug-in configuration, see the IBM Tivoli Network Manager IP Edition Event Management Guide.

Default event enrichment:

The default set of topology information added to events includes information about whether this is a root-cause event and the managed status of the entity on which the event occurred.

By default, events are enriched with the following topology information.

Table 1. Default set of topology information added to events

<table>
<thead>
<tr>
<th>Field in ObjectServer alerts.status table</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>NmosSerial</td>
<td>The serial number of a suppressing event. Note: The suppressing event is not necessarily a root cause event. There could be a chain of suppressing events from the perspective of the suppressed event.</td>
</tr>
<tr>
<td>NmosObjInst</td>
<td>An ID representing the device to which the event corresponds.</td>
</tr>
<tr>
<td>NmosCauseType</td>
<td>Indicates whether this is a root-cause or symptom alert.</td>
</tr>
<tr>
<td>NmosDomainName</td>
<td>The name of the Network Manager domain that is managing the event. The Event Gateway sets this field value based on a topology lookup.</td>
</tr>
<tr>
<td>NmosEntityId</td>
<td>A unique numerical ID that identifies the Network Manager topology entity with which the event is associated. This column is similar to the NmosObjInst column, but is more granular. For example, the NmosEntityId value can represent the ID of an interface within a device.</td>
</tr>
<tr>
<td>NmosEventMap</td>
<td>Specifies the event map that is used to process the event. The event map defines how the RCA plugin processes the event. For example, you can configure which Event Gateway stitcher is called for which event map.</td>
</tr>
<tr>
<td>NmosManagedStatus</td>
<td>The managed status of the network entity for which the event was raised. Can apply to events from Network Manager and from any probe. You can use this column to filter out events from devices, interfaces, and other entities that are not considered relevant.</td>
</tr>
</tbody>
</table>
About root cause analysis

Within Network Manager, the term root cause analysis is used to refer to topological root cause analysis. Topological root-cause analysis means that, in a situation where there are multiple events, Network Manager uses knowledge of the network topology to establish a point of failure and to identify those events that are symptomatic.

A failure on the network usually generates multiple alerts. This is because a failure condition on one device may render other devices inaccessible. Alerts are generated indicating that all of these devices are inaccessible.

Network Manager performs root-cause analysis by correlating event information with topology information, and thereby determining which devices are temporarily inaccessible due to other network failures.

Alerts on devices which are temporarily inaccessible are suppressed, that is, shown as symptoms of the original, root cause alert. Root-cause alerts are shown in alert lists and topology maps with the highest severity so that operators can easily identify them.

Root cause is implemented using a series of root cause analysis rules. These rules are enabled by default. No manual configuration is required.

Root-cause analysis tasks

Network administrators can perform administrative tasks, such as enabling and disabling RCA, and changing the event maps handled by RCA in order to modify which types of events are handled by RCA. Network operators use RCA to investigate the root-cause of events.

Administering RCA

Network administrators can perform various administrative tasks, including the following:

- Enabling and disabling RCA.
- Modifying which types of events are handled by RCA by changing the event maps handled by RCA.
- Changing the relative importance of different event types. When there are multiple events on the same entity, the event with the highest precedence value on the entity is used to suppress other events.
- Configuring features of RCA behavior, such as the maximum age difference between events that pass through the RCA plug-in. Events that have a difference in age greater than this specified value cannot suppress each other. The default value is 5 minutes.

For more information on RCA administrative tasks, see the IBM Tivoli Network Manager IP Edition Event Management Guide.

Investigating root-cause

Network operators use RCA to investigate the root-cause of events. They can determine which events are root cause and which events are results of that root cause (symptom events) and this enables them to quickly focus on the events that are causing network problems.
For more information on investigating the root-cause of network events, see the
IBM Tivoli Network Manager IP Edition Network Troubleshooting Guide.

**Event Gateway and Event Gateway plug-in architecture**

Use this information to understand how the Event Gateway, ncp_g_event and
Event Gateway plug-ins interact with other components of Network Manager.

The following figure shows how data flows through ncp_g_event and the Event
Gateway plug-ins.

![Event enrichment data flow](image)

**Figure 8. Event enrichment data flow**

1. **Event received from ObjectServer**
   
   Events are received from the Object Server at startup, upon receiving a
   SIGHUP command to the Event Gateway to change the configuration of
   the Event Gateway, and subsequently over the IDUC channel at a
   configurable 5 second interval. An incoming event filter rejects any events
   that do not have an associated LocalNodeAlias field (this field usually
   contains data that points to the main node device). An incoming field filter
   filters out alerts.status fields that do not participate in the Event Gateway
   processing.
**Event map is selected**

Based on the value of the event ID field in the event, the Event Gateway determines an event map to use to handle the event.

*Note:* The event map can also be set directly within the event by configuring the associated probe rules file or the Netcool/OMNibus Knowledge Library to populate the NmosEventMap alerts.status field directly with an event map value. In this case the event arrives with one of its fields containing a preset event map value. However, any event map settings configured in the Event Gateway override the event map settings configured in either the probe rules files or in the Netcool/OMNibus Knowledge Library. This enables you to locally override any event map settings configured in the network.

**Event map points to a topology lookup stitcher**

Optionally, the selected event map contains a pointer to a topology lookup stitcher. This step and the next step are optional. Some events do not call a topology lookup stitcher. For example, a Network Manager health check event passes through the Event Gateway purely in order to trigger the Failover Event Gateway plugin, and does not perform a topology lookup.

**Topology lookup performed**

The topology lookup stitcher performs a topology lookup to retrieve topology data associated with this event. Topology data is retrieved from NCIM cache.

For more information on these topics, see the following documentation:

- For information on Event Gateway stitchers, see the *IBM Tivoli Network Manager IP Edition Event Management Guide*.
- For information on stitcher rules used in the Event Gateway stitchers, see the *IBM Tivoli Network Manager IP Edition Language Reference*.
- For information on NCIM cache, see the *IBM Tivoli Network Manager IP Edition Product Overview*.

**Outgoing filter applied to event**

The outgoing filter only passes the fields enriched by the Event Gateway.

**Enriched fields are placed on the Event Gateway queue**

The outgoing Event Gateway queue receives enriched events from the Event Gateway stitchers (main event enrichment) and from the plug-ins.

**Based on a return value in the stitcher, the Event Gateway determines whether to send the enriched event to the plug-ins**

The Event Gateway determines whether a plug-in is interested in the event based on the state and event maps subscribed to by the plug-ins, and forward the event to the relevant plug-ins.

For information on plug-in subscriptions, and how to modify them, and on the gwPluginEventMaps and gwPluginEventStates tables that hold information on plug-in subscriptions, see the *IBM Tivoli Network Manager IP Edition Event Management Guide*.

**Plug-ins perform further event enrichment or take other action**

Plug-ins process the events to which they subscribe, and either perform further event enrichment, such as the RCA plug-in, which sets a field in the event to indicate whether this is a root-cause or symptom event, or take further action, such as the Disco plugin, which can initiate partial rediscovery based on an event.
Event data sent back to ObjectServer

Event data is sent back to the ObjectServer at a configurable interval of 5 seconds.

For more information on the outgoing Event Gateway queue and how to configure it, see the IBM Tivoli Network Manager IP Edition Event Management Guide.

For more information on the event enrichment dataflow, see the IBM Tivoli Network Manager IP Edition Event Management Guide.

About event storage

Event and alert data is stored in the Tivoli Netcool/OMNibus ObjectServer. The ObjectServer is a high-speed, in-memory event database at the center of Tivoli Netcool/OMNibus.

Network Manager is one among many possible applications that feed events to the Tivoli Netcool/OMNibus ObjectServer. Each of these applications is called an event source. Other event sources can include Tivoli Netcool/OMNibus probes and monitors, Tivoli Business Service Manager, and other event management systems.

The ObjectServer receives and stores events from event sources. It eliminates duplicate events. The ObjectServer also correlates events by removing, for example, matching pairs of problem and resolution events.

For more information on Tivoli Netcool/OMNibus, refer to the publications described in “Publications” on page vi.

Event storage architecture

Use this information to understand how the ObjectServer interacts with components of Network Manager.

The following figure shows how data flows through the ObjectServer.
Events generated by Network Manager polls are sent to the ObjectServer
Probe for Tivoli Netcool/OMNIbus sends the converted event data to the Tivoli Netcool/OMNIbus ObjectServer.

2 Tivoli Netcool/OMNIbus probes and other event sources populate the ObjectServer
Tivoli Netcool/OMNIbus probes, and, potentially, other network event sources, populate the ObjectServer with network events.

3 Event correlation and deduplication
The ObjectServer performs event correlation and deduplication on all the events that it stores.

4 Event enrichment and root cause analysis
The Event Gateway requests a filtered subset of events from the ObjectServer. Enriched events are returned to the ObjectServer. These events are enriched with a default set of topology information. The returned events are also modified with root cause analysis and symptom information.

5 The Tivoli Netcool/OMNIbus Web GUI accesses the event data
The Tivoli Netcool/OMNIbus Web GUI was known as Netcool/Webtop in versions 2.2 and below. The Web GUI requests the latest set of events from
the ObjectServer. Any changes that the user makes to events using the Web
GUI are sent back to the ObjectServer.

**Event storage tasks**

Network administrators can add fields to the ObjectServer to support event
enrichment and topology enrichment activities. In addition, network administrators
can perform the full range of Tivoli Netcool/OMNIbus tasks.

As part of event enrichment activity, network administrators might need to extend
the ObjectServer to hold extra topology information added by the Event Gateway,
ncp_g_event. For example, network administrators might add fields to the
ObjectServer alerts.status table to store the following information:

- System location
- System contact

Network operators displaying information for an event in an event list would then
see the standard event information, together with location information, such as the
city or building in which the affected device is located, and the name of the
administrator responsible for that device.

For more information on Tivoli Netcool/OMNIbus, refer to the publications
described in “Publications” on page vi.

**About historical polled data collection and 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. Operators can then use the Tivoli
Common Reporting viewer to run performance reports to interpret the data. Use
this information to learn about historical polled data collection and storage
architecture and some of the tasks that network administrators and network
operators perform that relate to historical polled data.

**Historical polled data collection and storage architecture**

By default, Network Manager implements the NCPOLLDATA historical polled
data database (hereinafter referred to as the NCPOLLDATA database) using a
database schema within the NCIM topology database. You can optionally integrate
Network Manager with IBM Tivoli Monitoring, 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.

The Tivoli Data Warehouse option also provides a data summarization capability.
This enables data to be presented, for example in summary form, as follows:

- Last 24 hours (raw data)
- Last 7 days (hourly data)
- Last 30 days (daily data)
- Last 6 months (weekly data)

Tivoli Data Warehouse also supports advanced data pruning, archiving, and
management functionality.

The following figure shows how historical polled data flows through the
NCPOLLDATA database.
Historical polled data option is configured
A network administrator or operator configures collection of historical polled data on a specific device.

Network devices are polled for SNMP and ICMP data
The Polling engine, ncp_poller, polls the specified network device for the data specified. Data is polled at regular intervals and is stored in the NCPOLLDATA database.

Network administrators tune the NCPOLLDATA database
Network administrators manage the storage limit for historical polled data to ensure optimum report response times.

Historical polled data is retrieved from the NCPOLLDATA database
Network operators run performance data reports. The historical polled data for these reports is retrieved from the NCPOLLDATA database.
Historical polled data collection and storage tasks

Use this information to learn about some of the tasks that network administrators and network operators perform that relate to historical polled data collection and storage.

Configuring a collection of historical polled data

Network administrators who engage in a carefully thought-out historical polling data collection and storage strategy can provide a better understanding of behavior trends associated with throughput rates, device CPU and memory resources, interface usage, errors, discards, and so on. For example, such a strategy could allow network administrators to closely monitor:

- Problematic or key network devices after a maintenance period
- An area of the network where there are suspected problems

Network administrators use the Network Configuration GUI panels to configure a collection of historical polled data, which typically consists of the following tasks:

- Defining the specific SNMP and ICMP data to collect, including threshold triggers for alerts
- Defining the scope and time interval for polling
- Determining what SNMP and ICMP data to store
- Starting the data collection

For more information about tasks associated with polling the network and administering historical polled data, see the IBM Tivoli Network Manager IP Edition Event Management Guide.

Running performance reports using historical polled data

Network administrators and operators can access historical polled data reports by accessing the Tivoli Common Reporting section of Network Manager and running performance reports, or by issuing a right-click command on a selected device in a topology map.

Specifically, network administrators and operators can use the Tivoli Common Reporting section of Network Manager to:

- View sets of defined reports detailing trends and analysis based on SNMP and ICMP short term historical data collections for a subset of the collected data
- View generic Trend and TopN graphs of ad hoc collections of stored data

The TopN reports can help compare and focus on the right network devices and drill down to see patterns over time. Summarization reports (available with Tivoli Data Warehouse) can help extend the time period for which to compare performance results. The reports offered out of the box can be used as examples to create custom reports to meet specific needs.

For information on the tasks associated with reports, see the IBM Tivoli Network Manager IP Edition Administration Guide.

Managing storage limits for historical polled data

Network administrators manage the storage limit for historical polled data to ensure optimum report response times and to allow the performance data reports to display a greater amount of historical data. Network administrators can also set
automatic pruning of the NCPOLLDATA historical polled data database to a specified number of database rows. By default, the Network Manager poller maintains a pruning policy to maintain the latest 5 million database rows. Network administrators can increase this limit if their network environments are achieving satisfactory performance results when generating performance reports.

For more information on storage capacity limits for historical polled data, see the IBM Tivoli Network Manager IP Edition Event Management Guide.

Network administrators can reset the storage limit for historical polled data in the NCHOME/etc/precision/NcPollerSchema.cfg file for the local cache.

For more information on increasing the storage limit for historical polled data, see the IBM Tivoli Network Manager IP Edition Event Management Guide.

Visualization layer

This layer consists of topology visualization and event visualization tools.

About topology visualization

Network Manager provides two types of topology views for network visualization. It is also possible to explore the structure of network devices using the Structure Browser.

The topology visualization GUIs include single-portlet views, such as the Network 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.

Network Hop View

The Network Hop View shows a selected device and all devices connected to it up to a configurable number of hops, or connections. You can use the Network Hop View to search the network for a specific device and display the network around that device. This view is useful for viewing the impacted area of an outage.

The Network Hop View provides Layer 1, Layer 2, and Layer 3 views, and converged topology views based on available layer 1 to layer 3 topology information.

Network Views

Topology maps can be customized to show specific devices or to show specific device groupings such as subnets and VLANs.

Network administrators and operators can monitor distinct sections of the network by creating and visualizing partitioned network views using filters on any device or component attribute. For example, you can display network views based on location, technology, more complex filtered combinations of attributes.

Path Views

Path Views can display system paths such as MPLS TE paths and Virtual Circuits, as well as user-defined paths. You can trace IP paths on an ad hoc basis to view a
snapshot of a network path at a specific moment. Once saved, you can monitor these paths regularly.

**MIB Grapher**

Graphing a MIB variable is useful for fault analysis and resolution of network problems. By graphing a MIB, operators and administrators can see a real-time graph of specific MIB variables for a network device. The MIB variable is polled at a user-defined interval and displayed in a graph over time. Optionally, you can display historical data for the MIB variable.

**Multiportlet views**

Multiportlet views enable you to put single-portlet views, such as the Network Hop View and the Structure Browser together; for example, you can select a device in the Network Hop View and instantly see the interfaces and other components of the device in an adjacent Structure Browser portlet. You can also use multiportlet views to show simultaneous topology maps and event lists.

**Topology visualization architecture**

Use this information to understand how the topology visualization web applications interact with other components of Network Manager.

The following figure shows how data flows through the topology visualization web applications.

![Topology visualization data flow](image)

1. **Topology visualization web application accesses the topology**
   The topology visualization web application accesses data in the NCIM database.
Event information requested
The topology visualization web application requests event information from the Tivoli Netcool/OMNibus Web GUI.

Network operators visualize network topology
A network operator uses a web client to connect to the topology visualization web application and display Network Hop View, network views, and structure browser views. Multiple users can connect to the topology visualization web application.

Topology visualization tasks
Network administrators and operators can use topology visualization tools, such as the Network Hop View, Network Views, and the Structure Browser, to view the topology following a discovery. They can also perform diagnostic tasks on the network devices and components by launching in-context tools from topology maps.

Network administrators and operators can use topology visualization tools to perform a wide range of diagnostic and information retrieval tasks, including the following:
- Identifying network problems
- Troubleshooting network problems by running troubleshooting tools within the Network Hop View and network views, as listed below
- Drilling into network devices to see faulty components
- Performing SNMP MIB queries on devices for diagnosis purposes
- Investigating network routes by issuing ping and traceroute commands
- Retrieving device information, such as domain registration information, DNS lookups, and retrieving specialized protocol information from Cisco and Juniper devices

Operators can also 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.

About event visualization
In Network Manager events are viewed using the Tivoli Netcool/OMNibus Web GUI event lists. Event lists can be filtered to monitor the health of specific areas of the network. Event severity is displayed to facilitate rapid identification of the more serious events. Right click tools provide immediate access to related topology views, stored information about affected devices, and real-time SNMP tools for problem diagnosis.

The Tivoli Netcool/OMNibus Web GUI was known as Netcool/Webtop in versions 2.2 and below. The Web GUI event lists provide a number of event display and event handling features, including the following:
- Event filtering: filters can be created to monitor different sections of the network. Multiple filters can be created and can be assigned to different operators based on operator responsibilities.
- Event severity: events are highlighted in the event list by severity. Events can be sorted and filtered by severity to enable rapid identification of more severe alerts.
- Right-click tools: a right-click takes the operator to any topology view in context, which displays the relationship of the affected device within the network. From the topology views, user can access a wide range of stored information about
the affected device and run diagnostic tools in real time. Users can also use right-click tools to acknowledge events and perform other event management tasks.

**Event visualization architecture**

Use this information to understand how the Tivoli Netcool/OMNIbus Web GUI interacts with components of Network Manager.

The Tivoli Netcool/OMNIbus Web GUI was known as Netcool/Webtop in versions 2.2 and below.

The following figure shows how data flows through the Web GUI.

![Event Visualization Flow Diagram](image)

**Figure 12. Event visualization data flow**

1. **Events are requested from the ObjectServer**
   The Web GUI requests the latest set of events from the ObjectServer. Any changes that the user makes to events using the Web GUI are sent back to the ObjectServer.

2. **Network operators monitor events in event lists**
   A network operator uses a Web browser to connect to the Web GUI and display event lists. Multiple clients can connect to the Web GUI. Network operators take action on the events.
**Event visualization tasks**

Operators can view events lists and use event severity ratings to quickly identify high-priority device events.

Operators can switch from alert views 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. Alerts may be generated by the Network Manager polling mechanism, or may be received from other network management systems.

Network operators use event visualization tools to perform the following tasks:

- Viewing event lists and using event severity ratings to quickly identify high priority device alerts.
- Switching from event lists to topology views (Network Hop Views and network views) to see which devices are affected by specific events, and to explore the network in context for related issues
- Identifying root-cause events and listing symptom alerts that contribute to the root cause.
- Identifying service-affected events, which are indicators that a critical customer service, such as a VPN, has a fault, and listing device events that contributed to a service-affected event

**About 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.

**Reporting architecture**

**Restriction:** Tivoli Common Reporting and the Tivoli Common Reporting reports must be installed in order to be able to use the Reports feature.

The following figure shows how data flows through Network Manager reporting.
A network operator uses a runs a report.

Data is requested from the NCIM and the NCPOLLDATA databases. If the operator ran a performance report, then the latest set of historical polled data is requested from the NCPOLLDATA database.

### Reporting tasks

Network Manager has a default set of reports under the following categories:

- Asset reports
- Current® Status reports
- Monitoring reports
- Network technology reports
- Network View Reports
- Path View Reports
- Performance reports
- Summary reports
- Troubleshooting reports

*Figure 13. Reporting data flow*

1. **Network operator runs a report**
   - A network operator uses a runs a report.

2. **Data is requested from the relevant database**
   - Data is requested from the NCIM and the NCPOLLDATA databases. If the operator ran a performance report, then the latest set of historical polled data is requested from the NCPOLLDATA database.
Utility reports

Network administrators create new reports and run existing reports. Network operators run reports.

**Tivoli Integrated Portal**

Tivoli Integrated Portal is the application that runs GUIs from different Tivoli products.

Tivoli Integrated Portal provides a single point for authentication and authorization of multiple web applications within that Tivoli Integrated Portal environment. Tivoli Integrated Portal also provides consolidated user management, and a single point of access for different applications. The Tivoli Integrated Portal also provides the ability to create customized pages and administer access to content by user, role, or group.

Tivoli Integrated Portal is installed automatically with the first Tivoli Integrated Portal-enabled product. Support for additional Tivoli Integrated Portal-enabled products can be added if installed in the same Tivoli Integrated Portal environment.

**Network Manager web applications**

Network Manager runs a number of web applications within Tivoli Integrated Portal.

**Multiple web clients**

Multiple web clients interact with the network topology, and distribute topology maps on demand. The topology maps include device status information, that is, the severity of the highest alert that affects a device, which is calculated from ObjectServer events. You can also use the Structure Browser to view device structure information.

The following web-client views are provided:

- Hop View
- Network Views

**SNMP MIB browser**

The MIB browser enables on-demand SNMP queries to network device MIBs.

**MIB graphing**

The MIB graphing function provides a real-time graph of specific MIB variables and MIB expressions on a network device. This graph is useful for fault analysis and resolution of network problems.

**Path Views**

Path Views display discovered system paths and paths through the network that are specified by the user.

**Reporting**

The Reporting section provides a set of default Business Intelligence and Reporting Tools (BIRT) and Tivoli Common Reporting reports, including performance reports. Reports are formatted for .html, .pdf, and .csv output, and can be formatted to PostScript. Ensure that at a minimum a reader is installed on your computer for the file format to which you output reports. For example, to output reports in .pdf format, install a PDF reader.
Network Discovery GUI
The Network Discovery GUI enables web-based configuration of network discovery.

Management Database Access page
The Management Database Access page enables web-based querying of Network Manager databases.

Network Polling GUI
Provides the front end for managing the Network Manager poll policies and definitions.

Web application architecture
To visualize the network, multiple Web clients (both Hop View and Network Views) connect to a single Tivoli Integrated Portal server. The Tivoli Netcool/OMNIbus Web GUI also connects to the Tivoli Integrated Portal server to enable Web-based viewing and interacting with alerts held in the ObjectServer.

The Tivoli Netcool/OMNIbus Web GUI was known as Netcool/Webtop in versions 2.2 and below.

The following figure shows how topology and event information is displayed in Network Manager.
1. **Topology transferred to NCIM**
   The Topology manager, ncp_model sends the topology to NCIM.

2. **The Topology Visualization Web application accesses the topology**
   The topology is accessed by the Topology Visualization Web application running within Tivoli Integrated Portal.

3. **Topology lookup performed by the Event Gateway**
   Topology data from NCIM cache is made available to the Event gateway, ncp_g_event.

4. **Event enrichment and root cause analysis**
   Using topology data from NCIM cache, the Event Gateway performs relevant topology lookup operations to enrich events from the ObjectServer. The Event Gateway then sends relevant event to the RCA plug-in to perform root cause analysis on events, as configured.

5. **Events requested from the ObjectServer**
   The Web GUI requests the latest set of events from the ObjectServer. Any changes that the user makes to events using the Web GUI are sent back to the ObjectServer.

*Figure 14. Visualization architecture*
Multiple client/server connections
Each Web GUI server can have multiple Web GUI clients connected to it.

Event information requested
The Topology Visualization Web application requests event information from the Web GUI application.

Events sent to topology visualization Web clients
The Tivoli Integrated Portal server sends topology maps to Network Hop View and Network Views clients on demand. The topology maps include device status information, that is, the severity of the highest alert affecting a given device, calculated from ObjectServer events.

Single sign-on
The single sign-on (SSO) capability in Tivoli products means that you can log on to one Tivoli application and then launch to other Tivoli Web-based or Web-enabled applications without having to re-enter your user credentials.

The repository for the user IDs can be the Tivoli Netcool/OMNIbus ObjectServer or a Lightweight Directory Access Protocol (LDAP) registry. A user logs on to one of the participating applications, at which time their credentials are authenticated at a central repository. With the credentials authenticated to a central location, the user can then launch from one application to another to view related data or perform actions. Single sign-on can be achieved between applications deployed to Tivoli Integrated Portal servers on multiple machines.

Single sign-on capabilities require that the participating products use Lightweight Third Party Authentication (LTPA) as the authentication mechanism. When SSO is enabled, a cookie is created containing the LTPA token and inserted into the HTTP response. When the user accesses other Web resources (portlets) in any other application server process in the same Domain Name Service (DNS) domain, the cookie is sent with the request. The LTPA token is then extracted from the cookie and validated. If the request is between different cells of application servers, you must share the LTPA keys and the user registry between the cells for SSO to work. The realm names on each system in the SSO domain are case sensitive and must match exactly. See Managing LTPA keys from multiple WebSphere Application Server cells on the WebSphere® Application Server Information Center.
Chapter 2. Benefits of Network Manager

Network Manager is a highly flexible system. Discovery, polling, and other parts of Network Manager can be extensively customized. Network Manager is also scalable; this means that it can discover increasingly bigger networks.

Comprehensive network management

Network Manager discovers, polls, and visualizes complex networks, containing a wide range of network-type devices, such as routers and switches, and using network protocols and technologies, such as MPLS, BGP, and OSPF.

Network Manager provides SNMP v1, v2, and v3 capabilities, and uses these capabilities to interrogate and poll network devices. Network Manager also provides the capability to discover and poll IPv6 devices.

Flexible network visualization

Network Manager provides different ways to visualize the network include network views, which show standard and customized device groupings such as subnets, VLANs, and VPNs, and the Network Hop View, which show a selected device and all devices connected to it up to a configurable number of connections.

Users can also navigate the interfaces and other components of a device using the Structure Browser. Multiportlet views enable you to put these views together; for example, you can select a device in a Network Hop View and instantly see the interfaces and other components of the device in an adjacent Structure Browser portlet. You can also use multiportlet views to show simultaneous topology maps and event lists.

Built-in device and interface polling capabilities

Network Manager provides a set of ready-to-use device and interface polls, including ping polls and MIB variable threshold polls. The MIB variable threshold polls generate network events if thresholds are violated on specified MIB variables.

You can customize network polling to so that events are received when thresholds are violated on any MIB variable on your network devices.

Built-in root-cause analysis capabilities

Network Manager sorts through multiple network events and uses knowledge of network topology to determine a single root-cause event. Network Manager highlights root-cause events in event lists and in topology maps so that your operators can instantly determine where to begin troubleshooting the network.
**Single-click network troubleshooting**

Network Manager provides a set of ready-to-use right-click tools to perform diagnostic and information retrieval actions on network devices shown in network topology maps. For example, you can perform diagnostic actions such as ping and traceroutes and you can retrieve device information such as DNS lookups or retrieve more complex protocol information such as BGP and OSPF information. You can add right-click tools to perform any desired action on a device.

**Rich network topology and event data**

You can enrich network topology using data from third-party sources, and you can enrich event data with topology data.

**Topology enrichment**

You can customize discovery to retrieve and store data about the discovered devices from third-party data sources. For example, you could retrieve customer information related to devices from a third-party inventory database. This would enable network operators to see the customer associated with a given device or network event.

**Event enrichment**

You can enrich network events with any topology data retrieved by the discovery process. Standard network events on device interfaces that originate from traps show the interface index only. You can enrich these events with interface name and description data. Operators viewing network events on device interfaces can then easily identify the interface. Another example of event enrichment is where you enrich events with topology information specifying the location of the network entity affected and a contact name for that network entity. Network operators can then use this information to support problem resolution, either by directly contacting the device administrator, or by including the contact information in a trouble ticket.

**Increasingly bigger network discovery**

Network Manager can discover and manage increasingly bigger networks.

**Related concepts:**

*Network and deployment comparisons* on page 55

Use this information to compare the example customer networks and to compare the Network Manager deployments for each of the example customer networks.

**Extensive reporting capabilities**

Run reports to retrieve a wide range of network data, including network performance, network assets, and network technology.

**Restriction:** Tivoli Common Reporting and the Tivoli Common Reporting reports must be installed in order to be able to use the Reports feature.
**Fully customizable content**

You can build pages that contain any combination of data. For example, you can combine topology maps with device structure views and event lists. You can also combine discovery status information with event lists that show custom discovery events.

**Multiple integration options**

By default Network Manager integrates with a number of Tivoli products, including Tivoli Business Service Manager (TBSM), Tivoli Application Dependency Discovery Manager (TADDM), IBM Systems Director, and IBM Tivoli Monitoring (ITM).
Chapter 3. Deployment of Network Manager

Use this information for guidance on how to configure the physical deployment of your Network Manager installation.

Deployment scenarios

How you deploy Network Manager depends on your environment, including factors such as the size and complexity of your network and the number of operations staff who require system access.

The following are typical Network Manager deployment scenarios:
- Small demonstration or educational system deployment
- Small customer network
- Medium customer network
- Large customer network
- Very large customer network

A further deployment scenario is the following: Telecommunications company or service provider network.

**Note:** Failover can be applied to each of these Network Manager deployments.

This section provides general guidance to assist you in deciding how to deploy Network Manager. For more detailed information, see the *IBM Tivoli Network Manager IP Edition Installation and Configuration Guide* and the *IBM Tivoli Network Manager IP Edition Release Notes*.

Network and deployment comparisons

Use this information to compare the example customer networks and to compare the Network Manager deployments for each of the example customer networks.

**Customer networks compared**

Use this information to compare the example customer networks and to identify which example most closely matches your network.

The following table lists typical features for each of the example customer networks. These values are example values only. Your specific network values might vary. In particular, you should note the following:

- With regard to the values for *Average number of interfaces per device* specified in this table, the actual interface counts can vary considerably from the average interface count. An example of this is found in MPLS networks, where the number of interfaces per device is very high in the core network, but might be as low as 2 to 3 interfaces per device for the edge devices.
- With regards to the number of devices for a telecommunications company, the value specified (15,000) is an average value. A national telecommunications company will have a far larger number of devices, a small local telecommunications company will have far fewer.
Table 2. Example customer networks compared

<table>
<thead>
<tr>
<th>Feature</th>
<th>Demo</th>
<th>Enterprise</th>
<th>Telco</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Small</td>
<td>Medium</td>
<td>Large</td>
</tr>
<tr>
<td>Number of devices</td>
<td>25</td>
<td>150 to 300</td>
<td>250 to 5,000</td>
</tr>
<tr>
<td>Average number of interfaces per device</td>
<td>1-2</td>
<td>3-5</td>
<td>20-30</td>
</tr>
<tr>
<td>Network locations</td>
<td>Single location</td>
<td>Single location</td>
<td>Distributed</td>
</tr>
<tr>
<td>Network architecture</td>
<td>Flat</td>
<td>Flat</td>
<td>Flat</td>
</tr>
<tr>
<td>Number of active GUI clients</td>
<td>1 to 3</td>
<td>3</td>
<td>5 to 20</td>
</tr>
<tr>
<td>Chassis ping polling examples</td>
<td>Values set for demonstration purposes</td>
<td>2-minute intervals</td>
<td>2 - 5 minutes</td>
</tr>
<tr>
<td>SNMP polling examples</td>
<td>Values set for demonstration purposes</td>
<td>3 to 6 values at 30 minute intervals</td>
<td>5 to 15 minute intervals</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SNMP v1, 2c, or 3 polling in any of the environments listed</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Device and interface polls in any of the environments listed.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Tivoli product integrations</td>
<td>None</td>
<td>None</td>
<td>ITM with TDW</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>TBSM</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>TADDM</td>
</tr>
<tr>
<td>Performance data collection period</td>
<td>1 to 5 days</td>
<td>31 days</td>
<td>31 days</td>
</tr>
</tbody>
</table>

**Network Manager deployments compared**

Use this information to compare the Network Manager deployments for each of the example customer networks.

The following table lists the settings required for the Network Manager deployments for each of the example customer networks. These values are example values only. The values that are appropriate for your specific deployment might vary.

**Note:** With regard to the values for Deployment specified in this table, these values do not take failover servers into account.
Table 3. Example Network Manager deployments compared

<table>
<thead>
<tr>
<th>Settings</th>
<th>Demo</th>
<th>Enterprise</th>
<th>Telco</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Small</td>
<td>Medium</td>
</tr>
<tr>
<td>Platform</td>
<td>Linux x86</td>
<td>Any supported platform</td>
<td>Any supported platform</td>
</tr>
<tr>
<td>Deployment</td>
<td>Single server</td>
<td>Single server</td>
<td>1-2 servers</td>
</tr>
<tr>
<td>Client system</td>
<td>Single processor</td>
<td>2 GB DRAM minimum, or 4 GB DRAM for large networks</td>
<td>Supported JRE and Internet browser</td>
</tr>
<tr>
<td>Topology database</td>
<td>Default database</td>
<td>Default database</td>
<td>Any supported RDBMS</td>
</tr>
<tr>
<td>Number of network domains</td>
<td>1</td>
<td>1</td>
<td>1 - 2</td>
</tr>
<tr>
<td>Number of polling engines based on network size</td>
<td>1</td>
<td>1</td>
<td>Consider more than one poller</td>
</tr>
</tbody>
</table>

Reasons for multiple domains

There are a number of reasons why you might need to partition your network into multiple domains.

You might need to partition your network into multiple domains for one of the following reasons:

- Your network exceeds a certain size. See the section Guidelines for number of network domains to determine whether your network requires multiple domains.
- Discovery takes a very long time. You can shorten your discovery times by partitioning your network into multiple domains.
- Operational boundaries dictate the need for multiple domains. Examples of operational boundaries include geographical boundaries and security boundaries.
- Your network contains overlapping IP addresses.

Tip:

If you are intending to run cross-domain discoveries in order to visualize links between the different domains, you must consider how the boundaries between domains affects cross-domain links. In particular, ensure that you choose domain boundaries with a minimum of cross-domain links.

Related concepts:

"Guidelines for number of network domains" on page 58

If your network exceeds a certain size, you might need to break up the network into multiple domains. Use this information to work out the number of network domains needed for your deployment.
Guidelines for number of network domains

If your network exceeds a certain size, you might need to break up the network into multiple domains. Use this information to work out the number of network domains needed for your deployment.

Depending on the operating system, a single Network Manager domain can support approximately 1,000,000 network entities that are created during a discovery operation. Network entities include ports, interfaces (including logical interface elements), cards, slots, and chassis. It is theoretically possible to include more network entities in a single domain, but discovery might take a long time to complete.

In previous versions of Network Manager, the maximum amount of memory that could be allocated to an individual process was typically 4GB. On 64-bit Linux, the maximum memory allowed for a process is not arbitrarily limited but depends on how much memory is installed on the server.

Generally, the discovery process (ncp_disco) and the topology model process (ncp_model) use the most memory.

The number of network entities that a discovery operation creates is dependent on a number of factors that might require you to create and configure extra network domains. These factors include the following:

- Device types — For example, a Cisco NEXUS or Juniper router with virtual router instances can contribute hundreds or thousands of network entities (ports, interfaces, cards, slots, and so on) per chassis.
- Network type — For example, a discovery operation performed on a local area network (LAN) typically contributes more network entities than a comparable size wide area network (WAN).
- Type of discovery agents enabled — For example, the Entity and JuniperBoxAnatomy discovery agents are inventory based discovery agents that typically create extra network entities that other agents do not create.
- Routed or switched network — For example, switched networks tend to generate more network entities than routed networks because they contain VLANs, which contain multiple entities.

The size of a Network Manager domain might be driven by business requirements. For example, a customer might require a network discovery to complete within defined daily maintenance periods. In this scenario, although a single Network Manager domain running on UNIX systems can support approximately 1,000,000 network entities, the length of time to complete a discovery of this size might not fit within the daily maintenance period. Consequently, two scoped domains, each supporting approximately 500,000 network entities, are required to support this business requirement.

Use the following procedure to determine the number of required domains. For information on how to create and configure extra network domains, see the IBM Tivoli Network Manager IP Edition Installation and Configuration Guide.

Note: The calculations presented here provide approximate figures only. The actual number of domains required varies, depending on various factors, including the factors described previously.

1. Gather the following data:
   - Number of devices in the network
• Average number of interfaces per device

Note: The actual interface counts on a given device can vary considerably from the average interface count. An example of this is found in MPLS networks, where the number of interfaces per device is very high in the core network, but might be as low as 2 to 3 interfaces per device for the edge devices.

2. Apply the following equation to determine an approximate number of network entities:
   Number of network entities = Number of devices * Average interface count * multiplier

   Where:
   • multiplier = 2 for a routed network
   • multiplier = 3.5 for a switched network

Note: Switched networks tend to generate more network entities because they contain VLANs, which contain multiple entities.

3. Apply the following equation to determine the suggested number of network domains:
   Number of domains required = (Number of network entities) / 1,000,000

Note: The suggested maximum number of network entities is only a rough guideline for domain sizing. The actual number of network entities per domain varies depending on various factors, including the factors described previously.

Router-centric customer

The data for this customer is as follows:
• Number of devices in the network: 60,000
• Average number of interfaces per device: 20

This customer network will produce approximately 2,400,000 network entities:

Number of network entities = 60,000 * 20 * 2 = 2,400,000

Based on the following calculation, this network requires three network domains:

Number of domains required = 2,400,000 / 1,000,000 > 2

Switch-centric customer

The data for this customer is as follows:
• Number of devices in the network: 1,000
• Average number of interfaces per device: 24

This customer network will produce approximately 84,000 network entities:

Number of network entities = 1,000 * 24 * 3.5 = 84,000

Based on the following calculation, this network requires one network domain:

Number of domains required = 84,000 / 1,000,000 < 1
Demonstration or educational system deployment

This is a small installation for use as a demonstration system or for training and educational purposes.

The following sections describe this network in greater detail and provide suggestions for a Network Manager deployment to meet the needs of this network.

Description

This environment consists of about 25 network devices and key servers combined. All devices are in one location, on the same network subnet as the devices to be managed. There is one local GUI client session supported by the same machine that hosts the Network Manager product components. There might be one or two GUI client sessions on other machines. The network devices come from multiple vendors. The network architecture is flat. All devices are attached to a LAN and have Fast Ethernet connections. For demonstration purposes only, a number of network devices have SNMPv3, and a number of workstations have IPv6.

Within this environment the following example conditions apply:
- 1 to 3 active GUI clients.
- Chassis ping polling and some SNMP polling activity is required.
- No major Tivoli products are integrated with the system, other than the required Tivoli Netcool/OMNibus.
- Performance reports are required for short data collection periods (typically 1 to 5 days) to match the length of the training course.

Network Manager deployment

A single-server deployment is sufficient for this type of environment. In addition to the single-server deployment description provided elsewhere, the following deployment settings are appropriate for this type of environment.
- System is an entry workstation class machine, with 8 GB of memory, dual-core processor preferred, single-core acceptable, reasonable current processor speed, and Fast Ethernet capability.
- Default database used for the NCIM database.
- Client system: single processor, 3 GB of memory, supported JRE and Internet browser
- IPv6 dual stack support is required if workstations or network devices have IPv6.

Small customer network

This customer is a company with a network consisting of about 150-300 network devices and key servers. The purpose of this installation is to manage this customer network by alerting the operations staff to major failures.

The following sections describe this network in greater detail and provide suggestions for a Network Manager deployment to meet the needs of this network.

Description

The primary users of the product are the networking operations staff. All devices are in one location and managed by a small operations group of a few people. Network devices come from multiple vendors. A mixture of layer 2 and layer 3
network devices are present. Approximately 20 to 30 VLANs are defined. The network architecture is fairly flat and simple. All devices to be managed are located in the same network as the Network Manager system and have Fast Ethernet connections. Internet connections are passed through a firewall and access to the systems within the protected network is available through a company VPN. The network operations staff have clients attached by means of one of the following: a local LAN, WiFi connections, or by means of a VPN established by a telecommunications service provider. Network changes are made once a month and a new discovery is anticipated at this time.

Within this environment the following example conditions apply:

- 3 active GUI clients.
- Chassis ping polling at two-minute intervals. SNMP polling at 30 minute intervals. Typically three to 6 SNMP MIB values require polling.
- No major Tivoli products are integrated with the system, other than the required Tivoli Netcool/OMNIbus.
- Performance reports are required for data collection periods on the order of 31 days.

**Network Manager deployment**

A single-server deployment is sufficient for this type of environment. In addition to the single-server deployment description provided elsewhere, the following deployment settings are appropriate for this type of environment.

- A single network domain is sufficient for this size of network.
- System can be any of the supported platforms. System requires 12 GB of memory, dual-core processor, and multiple physical disks in RAID 5 configuration.
- Client system: single processor, 3 GB of memory, supported JRE and Internet browser
- Default database used for the NCIM database.
- A single ncp_poller polling engine is sufficient for this environment.
- IPv6 dual stack support is required if workstations or network devices have IPv6.

**Medium customer network**

This customer is a company with a central major data center and connections to several remote sites. The purpose of this installation is to manage this customer network by alerting the operations staff to major failures.

The following sections describe this network in greater detail and provide suggestions for a Network Manager deployment to meet the needs of this network.

**Description**

This network has between 250 and 5,000 network devices and key servers of interest. Workstations, while numbering in the thousands, are not managed. Network devices come from multiple vendors. All devices in the central location have Fast Ethernet or Gigabit Ethernet connections. Remote sites are connected by WAN connections. The devices and servers to be managed are distributed among the central and remote sites.

Within this environment the following example conditions apply:
• There are 5 to 20 active GUI clients.
• Chassis ping polling at two to five-minute intervals. SNMP polling at five to
  15-minute intervals.
• Other major Tivoli products integrated with the system, other than the required
  Tivoli Netcool/OMNibus: IBM Tivoli Monitoring with Tivoli Data Warehouse
  running DB2 to support performance reporting.
• Performance reports are required for data collection periods on the order of 31
days.

Network Manager deployment

Each customer environment with this kind of network is different. The key to
success is adequate memory and a careful understanding of the polling targets,
combined polling rates, and the event rates. The following deployment settings are
appropriate for this type of environment.
• One or more network domains are required, depending on the size of network.
• Single server deployment (up to 250 network devices and 5 to 10 concurrent
  users)
  Four processors
  8 to 10 GB memory
  Multiple physical disks in RAID 5 configuration
• Two-server deployment (up to 5,000 network devices and 10 to 20 concurrent
  users)
  Four processors for system with Network Manager
  Four processors for system with Tivoli Netcool/OMNibus and Tivoli
  Integrated Portal
  8-16 GB memory for each server
  Multiple physical disks in RAID 5 configuration
• System may be any of the supported platforms.
• Client system: single processor, 3 GB of memory, supported JRE and Internet
  browser
• Any supported RDBMS used for the NCIM database.
• Number of polling engines:
  Single-server deployment: 1
  Two-server deployment: One poller for chassis pings, one or more pollers for
  SNMP polls
• IPv6 dual stack support is required if workstations or network devices have
IPv6.

Large customer network

This customer is a large enterprise company with a globally deployed network.
The purpose of this installation is to manage this customer network by alerting the
operations staff to major failures and to support the latest network devices and
network architecture.

The following sections describe this network in greater detail and provide
suggestions for a Network Manager deployment to meet the needs of this network.
Description

The architecture of the network is complex and contains the most up to date technology. For example, the network contains MPLS core networks. The network device count ranges from 5,000 to 15,000 devices, and the complexity of the network is reflected in the fact that there are 30 or more ports per device on average. Network operations are done from a central location with operations staff constantly monitoring the core network. Network devices come from multiple vendors.

Within this environment the following example conditions apply:

- There are typically 5 to 20 concurrently active GUI clients.
- Polling:
  - Chassis ping polling at two to 5 minute intervals.
  - SNMP polling at 10-15 minutes.
  - SNMPv3 polling of key network devices
  - SNMPv1 polling for real time graphing as well as storage for performance reports.
- Other major Tivoli products integrated with the system, other than the required Tivoli Netcool/OMNIbus:
  - IBM Tivoli Monitoring (ITM) with Tivoli Data Warehouse (TDW) running DB2 to support performance reporting.
  - IBM Tivoli Business Service Manager (TBSM)
  - IBM Tivoli Application Dependency Discovery Manager (TADDM)
- Performance reports are required for data collection periods on the order of 31 days.

Network Manager deployment

Deployment choices vary depending on the size of the network. For the 5000 device network in this customer range, the choice ranges from a single-server to a two-server deployment. Key factors for success include the network response time for the targets (given that this is a county or global distribution of target devices), memory availability on the supporting servers, the polling selected and the rates of polling.

For the top end of the network (approximately 15,000 devices), a distributed, multiple domain deployment is required. In addition to the multiple-server deployment description provided elsewhere, the following deployment settings are appropriate for this type of environment.

- Deploy two domains.
- Deployment of a dedicated database server.
- Each of the servers requires the following:
  - Four processors.
  - Up to 64 GB of memory.
  - 3 disk, RAID 5 multiple disk array
- For the systems used, deploy as follows:
  - Server 1: Network Manager with 36 GB of memory.
  - Server 2: Tivoli Netcool/OMNIbus and Tivoli Integrated Portal with up to 12 GB of memory
System 3 (optional): a customer-selected RDBMS with up to 16 GB of memory

- Systems to be deployed on Linux or UNIX platform.
- Any supported RDBMS used for the NCIM database.
- Two polling engines:
  - Use the default ncp_poller process for chassis ping.
  - Create a separate ncp_poller for the SNMP polls.
- Client system: single processor, 3 GB of memory, supported JRE and Internet browser
- IPv6 dual stack support is required if workstations or network devices have IPv6.

Very large customer network

This customer is a very large global enterprise company with a simple network architecture but very large numbers of devices. The purpose of this installation is to manage this customer network by alerting the operations staff to major failures and to support short-term capacity planning.

The following sections describe this network in greater detail and provide suggestions for a Network Manager deployment to meet the needs of this network.

Description

Network management is done from a central location and from regional locations. The network is very large and contains over 15,000 network devices and critical servers. Network devices come from multiple vendors. The devices fall into two categories:

- Network device infrastructure with interface counts in the range of 30 or more per device.
- Managed devices with 1-2 interfaces per device.

The majority of the devices are in the second category, managed devices. To manage a network of this size, the network is partitioned for management on a geographical basis.

Within this environment the following example conditions apply:

- There are 5 to 20 active GUI clients.
- Polling:
  - Chassis ping polling at two to 5 minute intervals.
  - SNMP polling at 15 minutes or longer.
  - SNMPv1 data collection
- Other major Tivoli products integrated with the system, other than the required Tivoli Netcool/OMNibus:
  - IBM Tivoli Monitoring (ITM) with Tivoli Data Warehouse (TDW) running DB2 to support performance reporting.
  - IBM Tivoli Business Service Manager (TBSM)
  - IBM Tivoli Application Dependency Discovery Manager (TADDM)
- Performance reports are required for data collection periods on the order of 31 days.
Network Manager deployment

Assistance from an experienced IBM services group or qualified IBM business partner is highly advisable for a successful deployment. Multiple domains are needed, supported by a collection of individual servers, or running together on a very large system. After completing a survey of the network to be managed, break the network up into sections that can be appropriately managed, and then assign each of these sections be to a domain. In addition to the multiple-server deployment description provided elsewhere, the following deployment settings are appropriate for this type of environment.

- Multiple network domains.
- Platform selections: Linux and UNIX.
- Large systems (many processors and very large amounts of memory) can host multiple domains as long as the memory allocations and processor counts are acceptable.
  - Memory: 32-64 GB per domain
  - Processors: 4-8 per domain depending on workloads
- Any supported RDBMS used for the NCIM database.
- Two polling engines for each domain:
  - Use the default ncp_poller process for chassis ping.
  - Create a separate ncp_poller for the SNMP polls.
- Individual process memory limitations are a factor in this environment.
- Client system: single processor, 3 GB of memory, supported JRE and Internet browser
- IPv6 dual stack support is required if workstations or network devices have IPv6.

Telecommunications company network

This customer is a telecommunications company and internet services provider. The purpose of this installation is to manage this customer network by alerting 24x7 network operations center staff to major failures.

The following sections describe this network in greater detail and provide suggestions for a Network Manager deployment to meet the needs of this network.

Description

The network to be managed has about 600 network devices; with an average interface count per device of 500. This is an MPLS network, and consequently the network devices are “large” in terms of their interface counts and complexity. Network devices come from multiple vendors. All devices are in one or more locations and are managed by a small network operations group. All devices to be managed are connected via Fast Ethernet or Gigabit Ethernet.

Within this environment the following example conditions apply:

- Number of simultaneous active clients: 5-20.
- Polling requirements: chassis pings at two to 5-minute intervals; SNMP polling of 5 values at 5 minute intervals.
- Some SNMPv3 polling is in place.
- Other major Tivoli products integrated with the system, other than the required Tivoli Netcool/OMNibus:
IBM Tivoli Monitoring (ITM) with Tivoli Data Warehouse (TDW) running DB2 to support performance reporting.
IBM Tivoli Business Service Manager (TBSM)
IBM Tivoli Application Dependency Discovery Manager (TADDM)

- Performance reports done once a day for key devices, used to assemble weekly capacity reports.

**Network Manager deployment**

A three-server deployment is needed for this type of environment. In addition to the multiple-server deployment description provided elsewhere, the following deployment settings are appropriate for this type of environment.

- One to two domains.
- A three-server deployment is advised.
- System specifications:
  - System 1 (where Network Manager is installed): four processors, 32-64 GB of memory, two or more disks. Note that beyond four processors or processor cores, the core clock speed and on-chip cache can be more important than additional cores. The general rule is as follows: select the fastest 4 cores before additional cores.
  - System 2: (where the Tivoli Integrated Portal and Tivoli Netcool/OMNIibus are installed) four processors, 16 GB of memory, two or more disks
  - System 3: database server; 4 processors, 16 GB of memory
- Any supported RDBMS used for the NCIM database.
- Two polling engines:
  - Use the default ncp_poller process for chassis ping.
  - Create a separate ncp_poller for the SNMP polls.
- Client system: single processor, 3 GB of memory, supported JRE and Internet browser
- IPv6 dual stack support is required if workstations or network devices have IPv6.

**LTE 4G wireless telecommunications company network**

This customer is a large wireless telecommunications company providing 4G wireless telephony services using their LTE (Long term Evolution) infrastructure. The purpose of this installation is to manage this customer network by alerting 24x7 network operations center staff to major failures.

The following sections provide suggestions for a Network Manager deployment to meet the needs of this network.

**Description**

The network to be managed has about 5000 eNodeBs, together with a further 3000 devices, which are either in the associated Evolved Packet Core (EPC) network or functioning as mobile backhaul router devices. The Network Manager entity count discovered for this network would typically be as follows:

- For the eNodeB devices: on the order of 20 to 25 entities per eNodeB.
- For the EPC network and backhaul equipment devices: on the order of 10 to 15 entities per device.
The LTE eNodeB and associated equipment is provided by multiple vendors which can vary in complexity and scale. These devices are distributed across many locations, as is typical in a 4G mobile network. All devices to be managed are connected using Fast Ethernet or Gigabit Ethernet. The topology and network inventory data discovered by Network Manager is received from vendor-specific EMS systems, and is processed by corresponding vendor-specific Network Manager collectors.

Within this environment the following conditions typically apply:
- Number of simultaneous active GUI clients: 5 to 20.
- Polling requirements: none, since topology data is received from the EMS, hence no polling of devices is assumed.

**Network Manager deployment**

A three-part deployment is needed for this type of environment.
1. Network Manager core components.
2. Tivoli Integrated Portal and Tivoli Common Reporting.
3. NCIM topology database.

These three parts can be deployed on servers (single or multiple) or on virtual machines (single or multiple). A single domain is sufficient for this type of environment. Furthermore, the following deployment settings are appropriate:
- Network Manager core components
  - 4 to 6 processors.
  - Approximately 20 GB of memory.
  - 50 GB of disk space.
- Tivoli Integrated Portal and Tivoli Common Reporting
  - 4 to 6 processors.
  - 6 to 8 GB of memory.
  - 20 GB of disk space.
- NCIM topology database
  - 4 processors.
  - 10 GB of memory.
  - 50 GB disk space minimum configured in a suitable RAID configuration, to provide required level of fault tolerance, reliability, and performance.
  - DB2, or other supported database platform.

In addition, the following GUI client system is appropriate:
- Single processor.
- 3 GB of memory.
- Supported JRE and Internet browser.
Deployment considerations

You can deploy your entire Network Manager installation on a single server or as a distributed installation.

During a Network Manager installation, you install the following four Network Manager components.

**Network Manager core**
This component consists of the core Network Manager processes: network discovery, polling, root cause analysis and event enrichment.

**NCIM database**
This database stores topology data.

**Tivoli Netcool/OMNIbus**
This component consists of the Tivoli Netcool/OMNIbus event management software. Many customers choose to have a trouble-ticketing system integrated with Tivoli Netcool/OMNIbus.

**Tivoli Integrated Portal**
This component consists of the Tivoli Integrated Portal user interface framework, together with the web applications.

The objective of the installation is to place these components on one or more servers.

The following are typical Network Manager deployment configurations:

- Single-server deployment
- Distributed deployment: two servers or more

The factors that require an increased number of servers in a distributed deployment include the following:

- Active event rates
- Amount and rate of stored polling data
- Device status polling rates and number of polling targets
- Network response times for polled targets
- Discovery frequency and
- Size of the network to be discovered (for each domain, where there are multiple domains)

**Note:** These deployment configurations do not take into consideration requirements for other product integrations.

In addition, you must consider deployment of appropriate systems to support GUI client sessions.

Also, IPv6 dual stack support is required if workstations or network devices have IPv6.

**Single-server deployment**

Single-server deployments are appropriate for small demonstration or educational systems, and for systems to support small to medium customer networks.
Distributed deployment: two servers or more

In distributed deployments, Network Manager components are distributed across multiple servers, that is, two servers or more. Here are some guidelines for distributed deployments:

- Two-server deployments are appropriate for the top end of the range of medium customer networks.
- Deployments might require three servers or more in situations where there are multiple network domains.
- Three-server deployments might also be deployed where it is determined that a separate server is required to support a relational database product that provides topology data storage. In addition, a separate database server enables the relational database to support multiple applications, in addition to Network Manager.

Two-server deployment

An example of a two-server deployment consists of the following allocation of host workstations:

- **Server 1**: Network Manager core components and the NCIM database. The core components are the network discovery, polling, root cause analysis and event enrichment components.
- **Server 2**: Tivoli Integrated Portal with associated Network Manager web applications.

Three-server deployment

An example of a three-server deployment consists of the following allocation of host workstations:

- **Server 1**: Network Manager core components.
- **Server 2**: Tivoli Netcool/OMNIbus
- **Server 3**: Tivoli Integrated Portal with associated Network Manager web applications, together with the NCIM database.

Client systems

You must consider deployment of appropriate systems to support GUI client sessions.

The following system specification provides support for a wide range of end-user activities on GUI client sessions:

**Note**: The web application clients, notably the Tivoli Netcool/OMNIbus Web GUI Active Event List and the Network Manager Network Views, Hop View, and Structure Browser, are Java-based and therefore are dependent on the performance of the client system. Consequently, the more memory and CPU performance on the client system, the better.

- Larger display supporting comfortable viewing at higher resolution, such as 1280x1024
- Current speed single or dual core processor
- 3 GB of memory
- Supported JRE and Internet browser
- Fast Ethernet.
• Processor specification:
  
  For normal topology displays or event displays
  Single processor with the following speeds: 1 GHz or better, as found on many laptops, 2.4 GHz, as found in many workstations

  Enhanced time to display larger or complex topology maps and enhanced display of MIB graphs
  A very current processor (3.0 GHz or better) typically available in the latest workstation class systems.

Deployment examples

Use these examples of Network Manager to help you plan your deployment architecture.

Constraints for installing and starting components

Some components must be installed and started before others. Use this information as well as the installation examples to understand the order in which you must install and start components.

Topology database constraints

You must install a topology database before you install the Network Manager core components, or as part of the same installation process.

You must install a topology database before you install the Network Manager Web applications (including the Tivoli Integrated Portal), or as part of the same installation process.

You must create database tables only during the first installation of the Network Manager core components or the Network Manager Web applications (including the Tivoli Integrated Portal), and not during subsequent installations.

Tivoli Netcool/OMNIbus constraints

You must install Tivoli Netcool/OMNIbus before you install the Network Manager Web applications (including the Tivoli Integrated Portal), or as part of the same installation process.

Web application constraints

You must install the Network Manager core components before you install the Network Manager Web applications, or as part of the same installation process.

If you are using ObjectServer authentication for the Network Manager Web applications, Tivoli Netcool/OMNIbus must be running during the installation of the Network Manager Web applications.

Starting components in the right order

Do not start the Network Manager core components until the installation of the Network Manager Web applications is complete.

Ensure that both Tivoli Netcool/OMNIbus and the topology database are running before starting the Network Manager core components.
Ensure that Tivoli Netcool/OMNIbus, the topology database, and the Network Manager core components are running before using the Network Manager Web applications.

**Example simple deployment architecture**

Use this example to familiarize yourself with the architecture of a simple Network Manager deployment.

**Components**

This example simple deployment consists of the following components:

- One ObjectServer virtual pair.
- One Tivoli Integrated Portal server.
- One Network Manager installation running one domain with failover.
- One instance of the NCIM topology database.

The following figure shows the architecture for this deployment.

![Diagram of Network Manager deployment architecture](image)

**Figure 15. Simple deployment architecture**

**Allocation of host workstations**

The following figure shows an example allocation of host workstations for this deployment.

**Note:** If you have a particularly large topology, you might want to install the topology database on its own server. This decision depends on the specification of your machines and how you want to spread the load between them.
Steps to install a simple deployment

The following steps provide an overview of the tasks required for this deployment, and help plan for a similar deployment.

To install the deployment described above, perform the following steps:

1. Install the topology database on host machine 3, create the necessary tables, and start the database.

   **Note:** The topology database must be installed and started before you start the Network Manager core components so that discovery data can be saved.

2. Install the following ObjectServers and related components:
   a. Install the primary ObjectServer and the Bi-directional Gateway on host machine 1.
   b. Install the backup ObjectServer on host machine 2.

3. Configure and run the ObjectServers.

   **Note:** The ObjectServers must be running before the Network Manager core components are started.

4. Install the primary Network Manager core components on host machine 2.

5. Install the backup Network Manager core components on host machine 1.

6. Install the Network Manager Web applications on host machine 3 (part of the GUI components category in the installation wizard).

   The Tivoli Integrated Portal server is automatically installed with the installation of the Network Manager Web applications.

   **Tip:** If you install the Tivoli Integrated Portal on a machine with no other products, performance is likely to be better than if you install it on a machine with other products.

   When you install the Network Manager web applications, the Tivoli Netcool/OMNIbus Web GUI is installed and automatically configured on host machine 3 if it is not already installed there. The Tivoli Netcool/OMNIbus Web GUI was known as Netcool/Webtop in versions 2.2 and below.
Note: The Network Manager core components must be installed before the Web applications.

7. Configure the primary Network Manager for failover and start it.
8. Configure the backup Network Manager for failover and start it.

Example large deployment architecture

Use this example to familiarize yourself with the architecture of a large Network Manager deployment.

Components

This example deployment consists of:

- One ObjectServer and one Network Manager installation in London. The London domain sends events and topology to San Francisco.
- One ObjectServer and one Network Manager installation in New York. The New York domain also sends events and topology to San Francisco.
- One ObjectServer and one Tivoli Integrated Portal installation in San Francisco. The ObjectServer in San Francisco consolidates the events from London and New York. The Tivoli Integrated Portal server in San Francisco can access topology from both London and New York, but does not consolidate the topologies. Clients anywhere in the world can connect to the Tivoli Integrated Portal server, and view topology from London and New York.

The following figure shows the architecture for this deployment.
Allocation of host workstations

The following figure shows an example allocation of servers for this deployment.
Steps to install a large deployment

The following steps provide an overview of the tasks required for this deployment, and help plan for a similar deployment.

**Note:** If you are installing distributed Network Manager core component and Web application servers with different time zones, then you should set the same time zone on all servers. This ensures that Network Manager is able to perform accurate timestamp comparisons from processes on different servers. You should also advise end users, such as network operators, that the system might display times that are different to the time in their location.

To install this deployment, perform the following steps:

1. Install the topology database on San Francisco host machine 3, and create the necessary database tables.
   **Note:** The topology database must be installed and started before you start the Network Manager core components so that discovery data can be saved.

2. Install the following ObjectServers and related components:
   - Install the ObjectServer on San Francisco host machine 2.
   - Install the ObjectServer and the uni-directional gateway on London host machine 2.
   - Install the ObjectServer and the uni-directional gateway on New York host machine 2.

3. Configure and run the ObjectServers.
Note: The ObjectServers must be running before the Network Manager core components are started.

4. Install the Network Manager core components on London host machine 1.

   Note: The Network Manager core components must be installed before the Web applications.

5. Install the Network Manager core components on New York host machine 1.

6. If a version of Tivoli Netcool/OMNIbus Web GUI earlier than version 7.3.1 is already present on host machine 3, then upgrade this to the Tivoli Netcool/OMNIbus Web GUI version 7.3.1. Network Manager is not compatible with earlier versions of the Tivoli Netcool/OMNIbus Web GUI.

7. Install the Network Manager Web applications on host machine 3 (part of the GUI components category in the installation wizard).

   The Tivoli Integrated Portal server is automatically installed with the installation of the Network Manager Web applications.

   Tip: If you install the Tivoli Integrated Portal on a machine with no other products, performance is likely to be better than if you install it on a machine with other products.

   When you install the Network Manager web applications, Tivoli Netcool/OMNIbus Web GUI version 7.4 is installed and automatically configured on host machine 3 if it is not already installed there.
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](#)

**bookmark**
See, [network view bookmark](#)

**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.

dNCIM database
The dNCIM is a relational database embedded into the Discovery engine,
ncp_disco, and it stores the containment model that is derived from the
fullTopology database (and created by stitchers). This is the version of the
topology that is sent to the Topology manager, ncp_model. The dNCIM
database performs the same function as the scratchTopology database did
in previous versions of Network Manager.

domain
See, network domain.

domain
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 by Network 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.

network view bookmark
Network view bookmarks group together just those network views that
you or your team need to monitor. Create new bookmarks or change
existing bookmarks to help network operators visualize just those devices
that they need to monitor.

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**
Stitches 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 uses 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.

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-Hakozaki cho, 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.
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 4 are trademarks of International Business Machines Corporation in the United States, other countries, or both:

Table 4. IBM trademarks

<table>
<thead>
<tr>
<th>Term</th>
<th>Trademark</th>
</tr>
</thead>
<tbody>
<tr>
<td>AIX®</td>
<td>Informix®</td>
</tr>
<tr>
<td>BNT</td>
<td>iSeries®</td>
</tr>
<tr>
<td>ClearQuest®</td>
<td>Lotus®</td>
</tr>
<tr>
<td>Cognos®</td>
<td>Netcool</td>
</tr>
<tr>
<td>DB2</td>
<td>NetView®</td>
</tr>
<tr>
<td>DB2 Universal Database</td>
<td>Notes®</td>
</tr>
<tr>
<td>developerWorks®</td>
<td>OMEGAMON®</td>
</tr>
<tr>
<td>DS8000</td>
<td>PowerPC</td>
</tr>
<tr>
<td>Enterprise Storage Server®</td>
<td>PowerVM®</td>
</tr>
<tr>
<td>IBM</td>
<td>PR/SM™</td>
</tr>
<tr>
<td></td>
<td>RDN</td>
</tr>
<tr>
<td></td>
<td>solidDB®</td>
</tr>
<tr>
<td></td>
<td>System p</td>
</tr>
<tr>
<td></td>
<td>System z®</td>
</tr>
<tr>
<td></td>
<td>Tivoli</td>
</tr>
<tr>
<td></td>
<td>WebSphere</td>
</tr>
<tr>
<td></td>
<td>z/OS®</td>
</tr>
<tr>
<td></td>
<td>z/VM®</td>
</tr>
<tr>
<td></td>
<td>zSeries</td>
</tr>
</tbody>
</table>

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 may collect IP addresses, user names and passwords for the purpose of performing network discovery. Failure to enable the collection of this information would likely eliminate important functionality provided by this Software Offering. You as customer should seek your own legal advice about any laws applicable to such data collection, including any requirements for notice and consent.

Index

A
accessibility ix
architecture
  EMS-based discovery 21, 22
  for event enrichment 34
  for Event Gateway 34
  for Event Gateway plug-ins 34
large deployment 73
plug-ins
  architecture 34
  simple deployment 71
  Topology Visualization Web
    application 48
    visualization 48
audience v

C
components of EMS-based discovery 22
configuration tasks
  for Event Gateway 31
  for Event Gateway plug-ins 31
  plug-ins
    configuration tasks 31
root-cause analysis 33
conventions, typeface xi

D
deployment
  large, architecture 73
  simple architecture 71
device alerts
  root cause analysis 30
devices
  monitoring with polling 25
discovery
  about 16
  architecture 17
cross-domain
  about 19
  EMS-based 21

E
education
  see Tivoli technical training ix
EMS-based discovery 21
  architecture 21
  components 22
detailed architecture 22
tasks 24
environment variables, notation xi
event enrichment
  architecture 34
tasks 31
Event Gateway
  architecture 34
  configuration tasks 31
Event Gateway plug-ins
  architecture 34
  configuration tasks 31
  events
    enrichment 36
    storage 36
    visualization 43
examining network devices
topology visualization GUls 41

G
glossary 77
GUls
  Tivoli Integrated Portal (TIP) 47
  using GUls from different Tivoli products 47

I
IBM Tivoli Network Manager IP Edition
  architecture v
  introducing v
installation
  distributed deployment 70
  for single sign-on 50
  order of components 70

M
manuals vi
MIB Browser
  definition 47
monitoring devices
  polling 25
multiple Web client views 47

N
NCIM (Network Connectivity and Inventory Model)
  topology data 28
  topology database 28
ncp_poller 26
Network Connectivity and Inventory Model (NCIM)
  topology data 28
  topology database 28
  network devices
    examining with topology visualization
      GUls 41
Network Discovery GUI
  definition 47
Network Manager
  overview 1
  network topology generating 16

O
online publications vi
OQL Workbench
  definition 47
ordering publications vi
overview information v

P
polling
  about 25
  architecture 26
  engine 26
ncp_poller 26
polling architecture 26
polling engine 26
product overview 1, 70
publications vi

R
RCA
  configuration tasks 33
Reporting
  About 45
  root cause analysis 30
  root-cause analysis
    configuration tasks 33

S
service x
service management connect x
single sign-on 50
SMC x
support x
support information x

T
tasks
  discovery
tasks for EMS-based discovery 24
  EMS-based discovery 24
  for event enrichment 31
TIP (Tivoli Integrated Portal) 47
Tivoli Integrated Portal (TIP) 47
Tivoli software information center vi
tivoli technical training ix
topology
  generating 16
  topology data 28
  topology database 28
  topology visualization GUls 41
Topology Visualization Web
  application 48
  architecture 48
  training, Tivoli technical ix
typeface conventions xi
V
variables, notation for xi
viewing the network
topology visualization GUIs 41
visualization
events 43

W
Web applications
definition 47