Tivoli® NetView® for OS/390®

Installation: Configuring Graphical Components

Version 1 Release 4

SC31-8770-00

Tivoli® NetView® for OS/390®

Installation: Configuring Graphical Components

Version 1 Release 4

SC31-8770-00

Installation: Configuring Graphical Components

Copyright Notice

Trademarks

IBM, the IBM logo, Tivoli, the Tivoli logo, Advanced Peer-to-Peer Networking, AIX, APPN, BookManager, DB2, IBMLink, Language Environment, MVS/ESA, NetView, OS/2, OS/390, RACF, SAA, SystemView, Tivoli Enterprise, VTAM, and z/OS are trademarks or registered trademarks of International Business Machines Corporation or Tivoli Systems Inc. in the United States, other countries, or both.

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

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

Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Intel is a trademark of Intel Corporation in the United States, other countries, or both.

Other company, product, and service names may be trademarks or service marks of others.

Notices

References in this publication to Tivoli Systems or IBM products, programs, or services do not imply that they will be available in all countries in which Tivoli Systems or IBM operates. Any reference to these products, programs, or services is not intended to imply that only Tivoli Systems or IBM products, programs, or services can be used. Subject to valid intellectual property or other legally protectable right of Tivoli Systems or IBM, any functionally equivalent product, program, or service can be used instead of the referenced product, program, or service. The evaluation and verification of operation in conjunction with other products, except those expressly designated by Tivoli Systems or IBM, are the responsibility of the user. Tivoli Systems or IBM may have patents or pending patent applications covering subject matter in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to the IBM Director of Licensing, IBM Corporation, North Castle Drive, Armonk, New York 10504-1785, U.S.A.

Programming Interfaces

This publication documents information NOT intended to be used as Programming Interfaces of Tivoli NetView for OS/390.

Contents

Preface

This book is designed to help system programmers install Tivoli® NetView® for OS/390® (NetView) graphics.

Who Should Read This Document

This document is written for system programmers, network planners, and system designers who install NetView graphics.

This document is **for Graphical Enterprise feature users.**

Prerequisite and Related Documents

This document assumes a working knowledge of the installation tasks as described in *Tivoli NetView for OS/390 Installation: Getting Started*. The tasks described here assume that you have completed the basic NetView installation process and have a functional system.

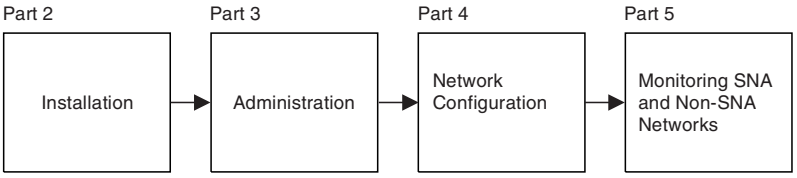


Figure 1. Installation Manuals

After you install NetView graphics, keep this document for reference for changing the system configuration, reviewing system definitions, or modifying settings after a PTF is installed.

If you are migrating from a previous release of NetView, review the changes and procedures described in the *Tivoli NetView for OS/390 Installation: Migration Guide*.

To read about the new functions offered in this release, refer to the *Tivoli NetView for OS/390 Installation: Getting Started*.

You can find additional product information on these Internet sites:

Table 1. Resource Address (URL)

IBM®	http://www.ibm.com/
Tivoli Systems	http://www.tivoli.com/
Tivoli NetView for OS/390	http://www.tivoli.com/nv390

The Tivoli NetView for OS/390 home page offers demonstrations of NetView, related products, and several free NetView applications you can download. These applications can help you with tasks such as:

- Getting statistics for your automation table and merging the statistics with a listing of the automation table
- Displaying the status of a JES job or cancelling a specified JES job
- Sending alerts to NetView using the program-to-program interface (PPI)

- Sending and receiving MVS commands using the PPI
- Sending TSO commands and receiving responses

What This Document Contains

This document is divided into the following sections:

- "Chapter 1. Introduction" on page 1

  Provides an overview of configuring components for NetView graphics.
- "Chapter 2. Enabling Graphics Support" on page 5

  Provides information for enabling various elements of the NetView graphics function, such as:
  - The Resource Object Data Manager (RODM)
  - The Graphic Monitor Facility host subsystem (GMFHS)
  - The SNA Topology Manager
  - The MultiSystem Manager
  - The NetView Resource Manager (NRM)
- "Chapter 3. Preparing and Activating RODM and GMFHS" on page 9

  Provides the following types of information about RODM and GMFHS:
  - Installing
  - Defining parameters
  - Bypassing or defining security
  - Verifying the installation
- "Chapter 4. Installing and Configuring the NetView Management Console for Graphics" on page 31

  Provides the following types of information about the NetView management console (NMC):
  - Setup of the NetView program
  - Workstation Setup for the NMC
  - Installing the NMC Topology Console
  - Using Java™ SNMP Services with NMC Applications
  - Verifying the installation
- "Chapter 5. Installing and Configuring SNA Topology Manager Agents" on page 49

  Provides the following types of information about SNA topology manager agents:
  - Establishing a plan to implement topology monitoring in your network
  - Establishing a plan to implement SNA APPN® Accounting in your network
- "Chapter 6. Preparing and Activating the SNA Topology Manager and the APPN Accounting Manager" on page 51

  Provides the steps for enabling the SNA topology manager and the APPN accounting manager:
  - Setting up VTAM
  - Setting up APPN accounting manager
- "Chapter 7. Installing and Configuring MultiSystem Manager Agents" on page 57

  Provides the steps for installing and configuring MultiSystem Manager agents:
  - Installing topology agents
  - Modifying the open environment
  - Modifying the LAN network manager environment

- "Chapter 8. Preparing and Activating the MultiSystem Manager" on page 61

  Provides information about modifying or establishing elements of the MultiSystem Manager, including (but not limited to) the following:
  - MVS Environment
  - Event Automation Service (EAS)
  - Security
  - NetView start procedure
  - Automation options
  - Assigning operator profiles to autotasks
  - Initialization
  - NetView Cross-Domain Environment
  - RUNCMD
  - NetView RATE and AUTORATE statements
  - Starting and verifying the installation
- "Chapter 9. Preparing and Activating the NetView Resource Manager" on page 71

  Provides information about configuring and installing the NetView Resource Manager, including the following:
  - Autotask AUTONRM
  - Security
  - Initialization
  - Startup of NetView Resource Manager
  - Verification of the installation
- "Chapter 10. Managing Views" on page 77

  Provides information about managing views, which are presented by the NetView management console (NMC) in a top-down, hierarchical perspective. This chapter discusses the types of views and addresses the following tasks:
  - Restricting access to views
  - Applying Policy to views
  - Initialization
  - Correlating views
  - Managing views
- "Appendix A. SNA Definitions for the NMC Topology Servers" on page 85

  Provides information about establishing LU 6.2 connectivity between your host and the NMC topology servers. This chapter includes the following information:
  - Ethernet LAN-Attached NMC Servers
  - Token-Ring LAN-Attached NMC Servers
  - SDLC-Attached NMC Servers
  - Defining the logical unit (LU) name
- "Appendix B. NetView Graphics Samples" on page 95

  Provides information about data samples for the following elements of NetView graphics:
  - GMFHS
  - MultiSystem Manager
  - NetView Management Console
  - NetView Resource Manager
  - RODM

– APPN

Conventions Used in This Document

The document uses several typeface conventions for special terms and actions. These conventions have the following meaning:

Bold	Commands, keywords, flags, and other information that you must use literally appear like **this**, in **bold**.
Italics	Variables and new terms appear like *this*, in *italics*. Words and phrases that are emphasized also appear like *this*, in *italics*.
`Monospace`	Code examples, output, and system messages appear like `this`, in a `monospace` font.
ALL CAPS	Tivoli NetView for OS/390 commands are in ALL CAPITAL letters.

Platform-specific Information

For more information about the hardware and software requirements for NetView components, refer to the *Tivoli Netview for OS/390 Licensed Program Specification.*

Terminology

For a list of Tivoli NetView for OS/390 terms and definitions, refer to http://www.networking.ibm.com/nsg/nsgmain.htm.

For brevity and readability, the following terms are used in this document:

NetView

- Tivoli NetView for OS/390 Version 1 Release 4
- Tivoli NetView for OS/390 Version 1 Release 3
- TME® 10 NetView for OS/390 Version 1 Release 2
- TME 10 NetView for OS/390 Version 1 Release 1
- IBM NetView for MVS Version 3
- IBM NetView for MVS Version 2 Release 4
- IBM NetView Version 2 Release 3

MVS MVS/ESA™, OS/390, or z/OS operating systems.

Tivoli Enterprise™ software
Tivoli software that manages large business networks.

Tivoli environment
The Tivoli applications, based upon the Tivoli Management Framework, that are installed at a specific customer location and that address network computing management issues across many platforms. In a Tivoli environment, a system administrator can distribute software, manage user configurations, change access privileges, automate operations, monitor resources, and schedule jobs. You may have used TME 10 environment in the past.

TME 10
In most product names, TME 10 has been changed to Tivoli.

V and R
Specifies the version and release.

VTAM® and TCP/IP
VTAM and TCP/IP for OS/390 are included in the IBM Communications Server for OS/390 element of the OS/390 operating system. Refer to http://www.software.ibm.com/enetwork/commserver/about/csos390.html.

Unless otherwise indicated, references to programs indicate the latest version and release of the programs. If only a version is indicated, the reference is to all releases within that version.

When a reference is made about using a personal computer or workstation, any programmable workstation can be used.

Reading Syntax Diagrams

Syntax diagrams start with double arrowheads on the left (►) and move along the main line until they end with two arrowheads facing each other (►◄).

As shown in the following table, syntax diagrams use ***position*** to indicate the required, optional, and default values for keywords, variables, and operands.

Table 2. How the Position of Syntax Diagram Elements Is Used

Element Position	Meaning
On the command line	Required
Above the command line	Default
Below the command line	Optional

Required Syntax

The command name, required keywords, variables, and operands are always on the main syntax line. Figure 2 specifies that the *resname* variable must be used for the CCPLOADF command.

CCPLOADF

Figure 2. Required Syntax Elements

Keywords and operands are written in uppercase letters. Lowercase letters indicate variables such as values or names that you supply. In Figure 3, MEMBER is an operand and *membername* is a variable that defines the name of the data set member for that operand.

TRANSMSG

Figure 3. Syntax for Variables

Optional Keywords and Variables

Optional keywords, variables, and operands are below the main syntax line. Figure 4 specifies that the ID operand can be used for the DISPREG command, but is not required.

DISPREG

Figure 4. Optional Syntax Elements

Default Values

Default values are above the main syntax line. If the default is a keyword, it appears only above the main line. You can specify this keyword or allow it to default.

If an operand has a default value, the operand appears both above and below the main line. A value below the main line indicates that if you choose to specify the operand, you must also specify either the default value or another value shown. If you do not specify an operand, the default value above the main line is used.

Figure 5 shows the default keyword STEP above the main line and the rest of the optional keywords below the main line. It also shows the default values for operands `MODNAME=*` and `OPTION=*` above and below the main line.

RID

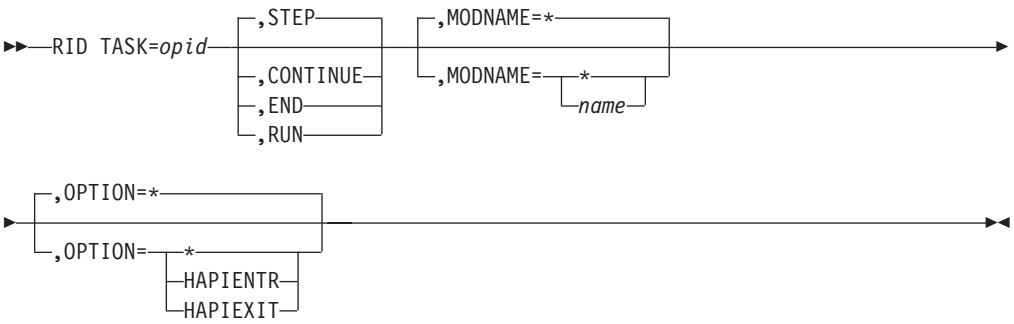


Figure 5. Sample of Defaults Syntax

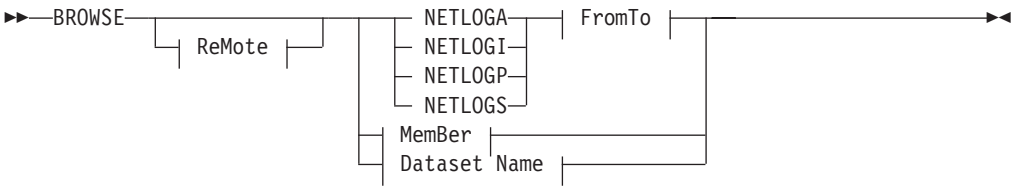
Long Syntax Diagrams

When more than one line is needed for a syntax diagram, the continued lines end with a single arrowhead (►). The following lines begin with a single arrowhead (►), as shown in Figure 5.

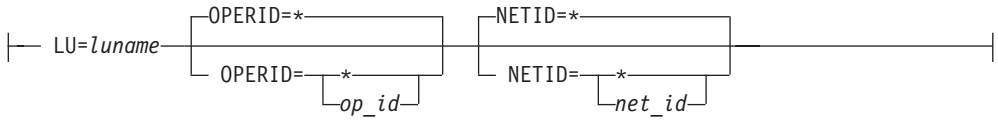
Syntax Fragments

Commands that contain lengthy groups or a section that is used more than once in a command are shown as separate fragments following the main diagram. The fragment name is shown in mixed case. See Figure 6 on page xiii for a syntax with the fragments `ReMote` and `FromTo`.

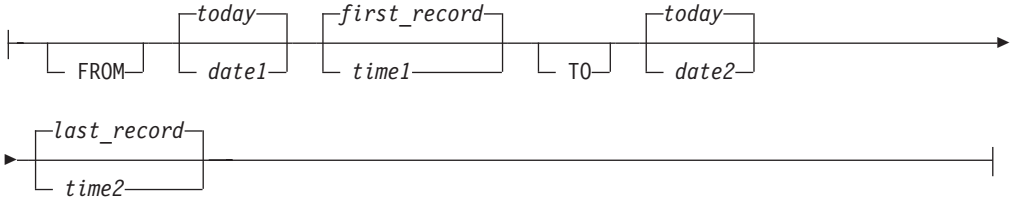
BROWSE



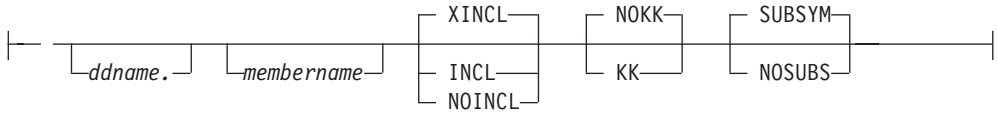
ReMote:



FromTo:



MemBer:



Dataset Name:

Figure 6. Sample Syntax Diagram with Fragments

Commas and Parentheses

Required commas and parentheses are included in the syntax diagram. When an operand has more than one value, the values are typically enclosed in parentheses and separated by commas. In Figure 7 on page xiv, the OP operand, for example, contains commas to indicate that you can specify multiple values for the *testop* variable.

CSCF

PurgeBefore

Pu

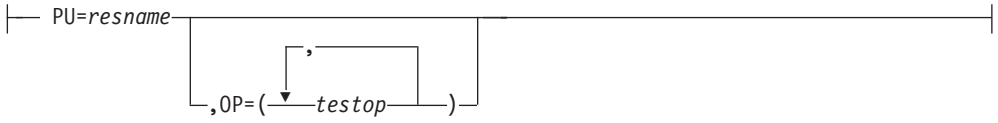


Figure 7. Sample Syntax Diagram with Commas

If a command requires positional commas to separate keywords and variables, the commas are shown before the keyword or variable, as in Figure 5 on page xii.

For example, to specify the BOSESS command with the *sessid* variable, enter:

```
NCCF BOSESS applid,,sessid
```

You do not need to specify the trailing positional commas. Positional and non-positional trailing commas either are ignored or cause the command to be rejected. Restrictions for each command state whether trailing commas cause the command to be rejected.

Highlighting, Brackets, and Braces

Syntax diagrams do not rely on highlighting, underscoring, brackets, or braces; variables are shown italicized in hardcopy or in a differentiating color for NetView help and BookManager® online books.

In parameter descriptions, the appearance of syntax elements in a diagram immediately tells you the type of element. See Table 3 for the appearance of syntax elements.

Table 3. Syntax Elements Examples

This element...	Looks like this...
Keyword	CCPLOADF
Variable	*resname*
Operand	MEMBER=*membername*
Default	*today* or INCL

Abbreviations

Command and keyword abbreviations are described in synonym tables after each command description.

Accessing Publications Online

The Tivoli Customer Support Web site (http://www.tivoli.com/support/) offers a guide to support services (the *Customer Support Handbook*); frequently asked questions (FAQs); and technical information, including release notes, user's guides, redbooks, and white papers. You can access Tivoli publications online at http://www.tivoli.com/support/documents/. The documentation for some products is available in PDF and HTML formats. Translated documents are also available for some products.

To access most of the documentation, you need an ID and a password. To obtain an ID for use on the support Web site, go to http://www.tivoli.com/support/getting/.

Resellers should refer to http://www.tivoli.com/support/smb/index.html for more information about obtaining Tivoli technical documentation and support.

Business Partners should refer to "Ordering Publications" for more information about obtaining Tivoli technical documentation.

Note: Additional support is also available on the NETVIEW CFORUM (Customer Forum) through the IBMLink™ system. This forum is monitored by NetView developers who answer questions and provide guidance. When a problem with the code is found, you are asked to open an official problem management record (PMR) to get resolution.

Ordering Publications

Order Tivoli publications online at http://www.tivoli.com/support/Prodman/html/pub_order.html or by calling one of the following telephone numbers:

- U.S. customers: (800) 879-2755
- Canadian customers: (800) 426-4968

Providing Feedback about Publications

We are very interested in hearing about your experience with Tivoli products and documentation, and we welcome your suggestions for improvements. If you have comments or suggestions about our products and documentation, contact us in one of the following ways:

- Send e-mail to pubs@tivoli.com.
- Fill out our customer feedback survey at http://www.tivoli.com/support/survey/.

Contacting Customer Support

The *Tivoli Customer Support Handbook* at http://www.tivoli.com/support/handbook/ provides information about all aspects of Tivoli Customer Support, including the following:

- Registration and eligibility
- How to contact support, depending on the severity of your problem
- Telephone numbers and e-mail addresses, depending on the country you are in
- What information you should gather before contacting support

Chapter 1. Introduction

This document discusses NetView graphics, which are based on the Resource Object Data Manager (RODM). RODM is an object-oriented data cache. Objects in RODM represent resources in your network. The data cache is located in the memory of the host processor. Many applications can interact with a single RODM, and more than one RODM can run on a host processor.

The Graphic Monitor Facility host subsystem (GMFHS) is the host program that works with RODM and the NetView management console (NMC) to manage resources. GMFHS supports resources that can send status updates to NetView in a standard format. GMFHS works with the SNA topology manager component of NetView and the NMC to manage SNA resources. GMFHS works with the MultiSystem Manager component and the NMC to manage non-SNA resources. GMFHS works with the NetView Resource Manager component and NMC to manage NetView tasks.

This document describes the steps required to install and enable the graphical functions of NetView. Also included are workstation-based components that support graphical or other access.

If you want information about...	Refer to...
Installing RODM and GMFHS	"Chapter 3. Preparing and Activating RODM and GMFHS" on page 9

Managing SNA Resources with the SNA Topology Manager and APPN Accounting Manager

Using the SNA topology manager and APPN accounting manager, NetView provides subarea and Advanced Peer-to-Peer Networking® (APPN) network management from the NMC.

The SNA topology manager and APPN accounting manager functions for managing SNA and APPN network environments include:

- Collection of subarea and APPN network, local, and LU status and topology. These functions provide:
  - Collection and storage of topology data, including real-time updates, in the RODM data cache
  - Dynamic graphical display of topology
- Centralized collection of advanced program-to-program communication (APPC) session and conversation accounting information. The architecture is known as LU 6.2 and an implementation of LU 6.2 is APPC. This information is logged to the System Management Facilities (SMF) or user-defined external log.

The manager applications for the SNA topology manager and APPN accounting manager are NetView applications. The accounting manager provides the ability to

retrieve and store LU 6.2 session and conversation accounting data. The topology manager provides the ability to obtain, monitor, and display the logical topology of your subarea and APPN networks. These manager applications use VTAM CMIP services to provide the actual communications support with their corresponding agent applications.

Three agent applications collect information for transmission to NetView:

- VTAM topology agent that gathers topology information about APPN network and end nodes, the subarea network, or both. VTAM CMIP services agents do not gather accounting information.
- Communications-Manager/2-(CM/2)-based APPN topology and accounting agent (APPNTAA) that gathers APPN accounting information from OS/2® servers.
- IBM 3746 Models 900 and 950 Network Node Processor feature can be set up to include the APPN topology agent that gathers information about APPN network nodes and end nodes.

Communication between the manager and agent applications is over LU 6.2 sessions that are using:

- OSI common management information protocols (CMIP)
- SNA multiple-domain support (MDS)

The manager applications can retrieve information from multiple agent applications. In addition, the agent applications can forward data to multiple managers (for example, if there is another NetView system in the network with the SNA topology manager and APPN accounting manager installed).

If you want information about...	Refer to...
Installing SNA topology manager and APPN accounting manager	“Chapter 6. Preparing and Activating the SNA Topology Manager and the APPN Accounting Manager” on page 51
Installing SNA topology manager and APPN accounting manager agents	“Chapter 5. Installing and Configuring SNA Topology Manager Agents” on page 49

Managing Non-SNA Resources with MultiSystem Manager

MultiSystem Manager is a NetView application that provides dynamic topology and status data in RODM for management of non-SNA domains. Agents are supported for the following network environments:

- LAN Network Manager LANs
- Novell NetWare network servers
- Internet protocol (IP) networks managed by NetView for AIX®, NetView for Windows NT®, NetView for Solaris, HP OpenView on HP-UX, or HP OpenView on Solaris
- ATM networks
- NetFinity networks
- Tivoli management regions
- Other types of networks using OPEN topology agents

MultiSystem Manager topology manager uses the GETTOPO command to request topology information and resource status from its agents. This topology and status information is stored in RODM. As topology and status changes occur, these agents notify the MultiSystem Manager topology manager, which updates RODM.

If you want information about...	Refer to...
Installing MultiSystem Manager	“Chapter 8. Preparing and Activating the MultiSystem Manager” on page 61
Installing MultiSystem Manager agents	“Chapter 7. Installing and Configuring MultiSystem Manager Agents” on page 57

Managing NetView Resources with the NetView Resource Manager

The NetView Resource Manager (NRM) enables you to manage all NetView hosts in an enterprise from the NetView management console (NMC). NetView hosts are monitored at the task level using the resource utilization function of the NetView product.

NRM consists of NetView hosts that are either managers or agents. Those that are managers require RODM and NMC to use the full function of NRM. NetView hosts that are agents forward their local host information to a manager or managers.

The following information applies to NRM:

- Both managers and agents can forward their local host information to one or more managers.
- The ability to forward local host information is an agent function.
- You can use SNA or TCP/IP as your communication vehicle between managers and agents.

Planning the Installation Process

The following set of tasks is an overview of the installation process. This overview lists steps that are required and those that are optional, depending upon the types of resources to be managed graphically.

1. **Required:** Prepare RODM, which includes:
   - Updating CNMSTYLE
   - Preparing MVS for RODM
   - Allocating the RODM data sets
   - Preparing the security system for RODM
   - Updating the RODM procedure
   - Customizing RODM startup values in EKGCUST
   - Customizing the automated operations RODM task
   - Updating the RODM load utility job to include the GMFHS data model
   - Updating the RODM load utility job to include the MultiSystem Manager data model
   - Updating the RODM load utility job to include the SNA topology manager data model
2. **Required:** Prepare GMFHS, which includes:
   - Updating CNMSTYLE
   - Updating the GMFHS start procedure
   - Customizing GMFHS startup values in DUIGINIT
   - Defining GMFHS-related tasks
3. **Required:** Prepare the NetView management console, which includes:
   - Updating CNMSTYLE

- Establishing a SNA LU 6.2 or TCP/IP session for the NETCONV command
- Installing the NMC topology server
- Installing NMC topology consoles

4. **Optional:** Prepare for managing non-SNA resources (for example IP resources), which includes:
   - Updating CNMSTYLE
   - Installing the appropriate MultiSystem Manager topology agents
   - Preparing the security system for the MultiSystem Manager
   - Preparing REXX for MultiSystem Manager
   - Preparing the MultiSystem Manager initialization file FLCSAINP
5. **Optional:** Prepare for managing SNA resources, which includes:
   - Updating CNMSTYLE
   - Preparing VTAM for the SNA topology manager
   - Preparing the security system for the SNA topology manager
   - Preparing the SNA topology manager initialization file FLBSYSD
6. **Optional:** Prepare for managing NetView resources, which includes:
   - Updating CNMSTYLE
   - Defining AUTONRM to DSIOPF
   - Preparing the security system for the NetView Resource Manager
7. **Optional:** Enabling SNMP management, which includes:
   - Enabling Java SNMP services
   - Enabling the MIB browser
   - Enabling the Real Time Poller

Chapter 2. Enabling Graphics Support

The base definitions in NetView that enable the graphics functions are contained in DSIPARM member CNMSTYLE and the files that it includes. The CNMSTYLE statements are read during NetView initialization. Changes to CNMSTYLE require a restart of the NetView program before they take effect. For this reason, make all CNMSTYLE updates at the same time for the graphics functions that you plan to use as part of this installation. The following sections provide details on the CNMSTYLE definitions for the various graphics functions.

Note: The actual member name for CNMSTYLE is controlled by the value of NV2I in the NetView start procedure.

For more information, refer to *Tivoli NetView for OS/390 Installation: Getting Started*.

If you want information about...	Refer to...
CNMSTYLE statements	*Tivoli NetView for OS/390 Administration Reference*

The graphics components are activated with TOWER statements in CNMSTYLE. Uncomment the following components that you will be using:

TOWER	Description
MSM	Enables the MultiSystem Manager.
Graphics	Enables the NetView management console.

An example TOWER statement follows:

```
TOWER =  *SA *AON  MSM  Graphics *AMI MVScmdMgt
```

When the TOWER is enabled, various associated functions are also enabled during initialization. NetView must be recycled for the TOWER statements to take effect. For the MultiSystem Manager and the SNA topology manager, you must also enable the subtowers.

See “Enabling SNA Topology Manager” on page 6 and “Enabling MultiSystem Manager” on page 6 for more information.

Enabling RODM

Many NetView processes require the Resource Object Data Manager (RODM) name to be specified. The RODMNAME statement sets the common global variable CNMSTYLE.RODMNAME.

If you set this name using the system symbolic &CNMRODM in SYS1.PARMLIB, you can use the following CNMSTYLE statement to define the RODM name:

```
RODMname =  &CNMRODM
```

If you do not set the &CNMRODM system symbolic in SYSI.PARMLIB, substitute the correct RODM name for &CNMRODM.

The NetView-supplied member CNMSTYLE also contains the following global variables, which are used by the RODM command list (CNME1098):

```
COMMON.EKGHNAM = RODM
COMMON.EKGHPRC = EKGXRODM
```

Update these variables as follows:

EKGHPRC Change the EKGHPRC global variable to the name of your RODM procedure (if it is not the same as the NetView-supplied default).

EKGHNAM If you started the RODM procedure using an identifier, change the EKGHNAM global variable to this identifier (if it is not the same as the NetView-supplied default). If you are not using an identifier, the EKGHNAM global variable is not required.

Enabling GMFHS

The NetView-supplied member CNMSTYLE contains the following global variables, which are used by the Graphic Monitor Facility host subsystem (GMFHS) command list (CNME2101):

```
COMMON.DUIFHNAM = GMFHS
COMMON.DUIFHPRC = CNMGMFHS
```

Update these variables as follows:

DUIFHPRC Specify your system PROC name for starting GMFHS. DUIFHPRC must be defined to the same name as the GMFHS procedure.

DUIFHNAM Specify the identifier you use when starting GMFHS. If you do not use an identifier when starting the GMFHS procedure, do not specify the global variable DUIFHNAM.

Enabling SNA Topology Manager

When the TOWER statement is enabled for Graphics, the SNA topology manager is automatically enabled and instructions are issued to start communication with the local VTAM agent.

SNATM is a subtower to the Graphics tower. If you are not using the SNA topology manager, type an asterisk at the beginning of the following statement:

```
TOWER.Graphics = SNATM
```

Statements preceded by an asterisk are disabled.

Enabling MultiSystem Manager

To enable the MultiSystem Manager agents, locate the following statement in CNMSTYLE:

```
TOWER.MSM = LNM NVL IP *ATM OPN NTF TMR
```

Agent names preceded by an asterisk are disabled. Remove or add asterisks as necessary to enable the agents that you will use:

LNM LAN Network Manager agent

NVL Novell Netware agent

IP IP agent

ATM ATM agent

OPN Open agent

NTF NetFinity agent

TMR Tivoli management regions agent

Note: When the TOWER.MSM statement is enabled, the %INCLUDE statement for FLCSOPF (used for operator profiles) is also enabled.

Enabling NetView Resource Manager

To enable NetView Resource Manager (NRM), perform the following steps:

1. Locate the following statement in CNMSTYLE:

   ```
   INIT.NRM = No
   ```

2. Change No to Yes.

   This starts NRM at NetView initialization.

   If you do not change parameters in CNMSTYLE, NRM defaults to type manager. No host destinations are defaulted.

3. If you want the NRM type to be agent, modify the following parameters in CNMSTYLE:

   - NRM.TYPE

     Change this parameter to NRM.TYPE = Agent.

   - NRM.HOSTDEST.x

     Code as many NRM.HOSTDEST.x statements as are applicable.

     Code CMODE and PORT statements as necessary (based on your HOSTDEST statements).

Chapter 3. Preparing and Activating RODM and GMFHS

NetView graphics are based on Resource Object Data Manager (RODM) and Graphic Monitor Facility host subsystem (GMFHS). Before you can define graphics, you must install and define parameters for RODM and GMFHS.

RODM and GMFHS require the Language Environment® for OS/390 run time library. Refer to the NetView program directory for more information.

Preparing RODM

Use the steps in this section to prepare RODM.

Defining RODM as an MVS Subsystem

Before RODM can be used, it must be defined as an MVS subsystem in SYS1.PARMLIB member IEFSSN*xx*. For more information, refer to *Tivoli NetView for OS/390 Installation: Getting Started*.

Allocating VSAM Clusters for RODM

Job EKGSJ004 defines the VSAM clusters for RODM. Table 4 on page 10 lists the data set names associated with RODM, as well as the names of members containing VSAM cluster information for these data sets. Any sample members

needed to run EKGSJ004 are located in one of two places: *either* in the NETVIEW.V1R4USER.INSTALL data set that was created when you ran job CNMSJBUP *or* in the NETVIEW.V1R4M0.SEKGSMP1 data set.

Note: CNMSJ004 is a sample job that is run during basic NetView installation and might already have been used to create the RODM VSAM clusters. In this case, EKGSJ004 does not need to be run.

Before allocating VSAM clusters:

1. Review EKGSID01 and each of the following VSAM allocate members in Table 4.

   These sample input members provide the name, volume, catalog, and password specifications for the VSAM clusters that get allocated and deleted.

   Ensure that the values specified in these input members are correct for your system.

Table 4. VSAM Clusters for the V1R4 Graphical Enterprise Feature Facilities

V1R4 Facility	VSAM Allocate Member	Data Sets	VSAM Delete Member
RODM log	EKGSI101	NETVIEW.CNM01.EKGLOGP NETVIEW.CNM01.EKGLOGS	EKGSID01
RODM checkpoint	EKGSI201	NETVIEW.CNM01.EKGMAST NETVIEW.CNM01.EKGTRAN NETVIEW.CNM01.EKGCK001 NETVIEW.CNM01.EKGCK002	EKGSID01

   The EKGSI101 sample specifies VSAM SHAREOPTIONS(2) for performance reasons. However, with SHAREOPTIONS(2), the RODM logging function makes no provision for secondary allocations. The results of RODM logging with SHAREOPTIONS(2) using secondary allocations are unpredictable.

   Note: The MVS MODIFY command enables you to specify RODM logging options.

2. Make sure that the VSAM cluster names match the names in your RODM startup procedure, EKGXRODM.
3. The checkpoint facility enables you to save a copy of the data cache to DASD.

   The data can then be read from DASD at RODM restart (warm start). Programs cannot access RODM data during checkpointing.

   The following VSAM clusters are used by the RODM checkpoint facility:

   - NETVIEW.CNM01.EKGMAST is the RODM master window checkpoint data set.

     This data set contains system information for RODM.
   - NETVIEW.CNM01.EKGTRAN is the RODM translation window checkpoint data set.

     This data set contains the translation address information that allows for correct data mapping and addressing in the RODM data cache.
   - NETVIEW.CNM01.EKGCK001 and NETVIEW.CNM01.EKGCK002 are RODM data window checkpoint data sets.

     These data sets contain the actual data in the RODM data cache.

     You can create additional data window checkpoint data sets. The low-level qualifier of these data sets must follow the format EKGCK*xxx*. If you add data

window checkpoint data sets, ensure that you have defined them to the RODM startup procedure, EKGXRODM.

The size of the RODM checkpointing facility VSAM clusters that were defined to the RODM startup procedure, EKGXRODM, determines the number of objects and classes that you can add to RODM. Refer to *Tivoli NetView for OS/390 Tuning Guide* for information on estimating the size of these data sets.

If you do not use the checkpoint facility, update the RODM procedure to remove the DD statements for the checkpoint data sets.

During initialization RODM looks for the checkpoint data sets. When it does not find them, you see one or more of the following messages:

```
EKG1123I EKGXRODM : THE CHECKPOINT FUNCTION IS NOW DISABLED
```

To allocate VSAM clusters:

1. The default volume specified in EKGSI101 and EKGSI201 is VOL(CPDLB2). If your volume is not CPDLB2, change CPDLB2 in these members to a valid volume.

   Note: If you are deleting any of your existing VSAM clusters, uncomment //STEP1 and review EKGSID01 to make sure you delete the correct VSAM cluster names.
2. Run EKGSJ004.
3. Verify your return codes before continuing with the next step.

   Note: If you need to allocate VSAM clusters for a second instance of RODM, you can run job EKGSLOG to allocate the RODM log data sets as defined in EKGSI102 and job EKGSCKPT to allocate the RODM checkpoint data sets as defined in EKGSI202.

If you want information about...	Refer to...
RODM logging options	"Diagnostic Tools for RODM" in *Tivoli NetView for OS/390 Diagnosis Guide*

Using the RODM Log Formatter

The RODM log formatter (EKGRLOG) provides formatted output of the contents of the RODM log.

If you want information about...	Refer to...
RODM log formatter	"Diagnostic Tools for RODM" in *Tivoli NetView for OS/390 Diagnosis Guide*

Using the RODM Dump Utility

The RODM dump utility is a service program that makes it possible to print data residing in the RODM data cache. The dump utility provides multiple formats for printing this information.

You can generate five types of reports to print the contents and structure of the RODM classes and objects. These reports include:

- Class listing
- Class index
- Object listing
- Object index

- Statistical report

If you want information about...	Refer to...
The output for each of the RODM dump utility reports	"Diagnostic Tools for RODM" in *Tivoli NetView for OS/390 Diagnosis Guide*

Bypassing or Defining Security

The commands you issue to define RODM and the operators to the security class can vary, depending on whether you use RACF® or another SAF product.

If you are using a SAF product, such as RACF, on your system, you can use one of the following methods to define RODM security:

- Bypass system security with `*TSTRODM`.
- Define the RODM task and authority level to the RODMMGR class of your SAF product, if it is available.
- Define the RODM task and resources that represent authority levels to a user-defined class in your SAF product.

If you want information about...	Refer to...
RODM security	*Tivoli NetView for OS/390 Security Reference*

Bypassing RODM Security

To bypass RODM security, initialize RODM with `*TSTRODM` in the `SEC_CLASS` field in EKGCUST when:

- Your system uses a SAF product, such as RACF, but you do not want to define RODM and operator tasks to the SAF product for security.
- Your system does not use a SAF product.
- The SAF product is not active on your system.

Note: You can now recycle the target system with the create link pack area (CLPA) option.

Defining RODM Security to the RODMMGR Class

If you are using the RACF product for RODM security, you need to have RACF at this level or higher to use the SAF RODMMGR class:

- RACF Version 1 Release 9 with PTF UW00497
- RACF Version 1 Release 9.2 with PTF UW00498
- RACF Version 2 Release 1 with PTF UW90113

Notes:

1. If your security product provides the RODMMGR class, you can now recycle the target system with the CLPA Option.
2. If you are using a SAF product which provides a RODMMGR class, define security resource names to that product and authorize users to the correct SAF resources.

Defining RODM Security to a User-Defined Class

If you are using a SAF product for RODM security and if the SAF product does not provide the SAF RODMMGR class, these steps must be completed before RODM can initialize:

- Define a security class in the SAF product for RODM.

For RACF, create a RACF router table for this security class, as described in the *Tivoli NetView for OS/390 Security Reference*.

- Define security resource names for the class you define. For more information, see "Resource Object Data Manager (RODM) Definition Statements" in the *Tivoli NetView for OS/390 Administration Reference*.

Defining the Resource Class to the RACF Class Descriptor Table

If you defined RODM security, the SEC_CLASS operand in EKGCUST in SEKGSMP1 enables you to specify the security class definition for your installed security system.

If you do not define the class name in the EKGCUST customization file, or if you do not include the EKGCUST DD statement in the JCL, the default security name is RODMMGR.

To use another class name as the default RACF security name, define the *class_name* to the RACF class descriptor table and the RACF router table. Locate the RFTABLE CNMSHjob in the RACINSTL member in SYS1.SAMPLIB. RACINSTL contains sample RACF installation jobs.

If you want information about...	Refer to...
RFTABLE job	*Tivoli NetView for OS/390 Security Reference*

Using RACF for RODM Security

If you use "Defining RODM Security to the RODMMGR Class" on page 12 or "Defining RODM Security to a User-Defined Class" on page 12, you must also perform the following operations:

- Define six RACF resource names under RODMMGR or your user-defined security class for the six user authority levels.
- Define user IDs for users who are connecting to RODM.
- Authorize user IDs to the appropriate RACF resource names.

Note: RODM only verifies security levels for API calls into RODM, and not on the MODIFY command interface. To implement security for the MODIFY command interface, refer to the RACF library and the MVS/ESA library.

Defining RACF Resource Names: To define the RACF resource names under RODMMGR for the six user authority levels, complete the following steps from your RACF-authorized TSO ID.

1. To define the RODM resource names, if SEC_RNAME is RODM, enter:

```
RDEFINE RODMMGR RODM1 UACC(NONE)
RDEFINE RODMMGR RODM2 UACC(NONE)
RDEFINE RODMMGR RODM3 UACC(NONE)
RDEFINE RODMMGR RODM4 UACC(NONE)
RDEFINE RODMMGR RODM5 UACC(NONE)
RDEFINE RODMMGR RODM6 UACC(NONE)
```

   If you have your own user-defined *class_name*, replace RODMMGR with the security class name on the RDEFINE commands. The resource names used are an example.

   The RODM resource names consists of a prefix and a suffix. The suffix must have values of 1 through 6 for the different levels of security. The default

resource name prefix is the RODM name specified in the RODM startup JCL. For example, the RODM name would be ZZRODM using either of the following start commands:

```
S EKGXRODM,NAME=ZZRODM
S EKGXRODM.ZZRODM
```

It is recommended that your resources use the name of your RODM.

If you specify your own RODM resource names, the resource name prefix must be specified in EKGCUST on the SEC_RNAME statement if the resource name prefix is not the name of your RODM.

RODM restricts the length of resource names by requiring that the resource name be one less than the number you specify for `MAXLNTH`. For RODMMGR, `MAXLNTH` is 44, so the resource name must contain 43 characters or less. If you define your own security class, `MAXLNTH` is specified when you define the RACF class descriptor table. See "Defining the Resource Class to the RACF Class Descriptor Table" on page 13 for information on defining a security class.

2. To set the system-wide RACF options, enter:

```
SETROPTS CLASSACT(RODMMGR)
```

If you have your own user-defined *class_name*, replace `RODMMGR` with the security class name on the SETROPTS command. The resource class name used is an example.

Defining User IDs for Users Who Are Connecting to RODM: Typically, user IDs are already defined. If the user ID is already defined to RACF (for example, for normal logon), no additional registration is required. To define a user ID, locate the RFTABLE job in the RACINSTL member in SYS1.SAMPLIB. RACINSTL contains sample RACF installation jobs.

If you want information about...	Refer to...
RFTABLE job	*Tivoli NetView for OS/390 Security Reference*

Authorizing User IDs to RACF Resource Names: To access RODM, enter the following from your authorized TSO ID for each *userid* that requires access:

```
PERMIT resourcename CLASS(RODMMGR) ID(userid)
```

Where:

resourcename
: Specifies the name of the RODM resource (such as RODM1 through RODM6) that has the appropriate security level for the function that the *userid* needs to be able to perform. Indicate the highest level RODM resource name the *userid* needs to access. If you indicate a user is authorized for RODM3, that user also has authorization for security level 1 (RODM1) and security level 2 (RODM2) capabilities.

  For example:

```
PERMIT RODM3 CLASS(RODMMGR) ID(USER1)
```

  Indicates that USER1 is authorized to perform the capabilities of RODM security levels 1, 2, and 3. Table 5 on page 15 describes the RODM security levels.

userid Specifies the RACF user ID. You can list individual user IDs. You can also

use the PERMIT command to authorize a group to the authority level resources, which enables you to connect or remove user IDs from the group as their need for RODM capabilities changes.

There are three special user IDs for AON, MultiSystem Manager, and the NetView Resource Manager. These user IDs consist of the domain ID concatenated with the characters AON, MSM, and NRM respectively.

Note: If you have a user-defined *class_name*, replace RODMMGR with the security class name on the PERMIT commands.

Table 5 shows how each level of access security includes those preceding it.

Table 5. RODM Access Security Levels

Resource Name	Security Level	Capabilities
*rodm*1	1	Connect and disconnect to RODM
*rodm*2	2	Query and list of functions (queries only)
*rodm*3	3	Action and list of functions (queries or actions) including triggering methods and change methods
*rodm*4	4	Checkpointing
*rodm*5	5	Administrative functions (adding or deleting from the RODM data cache) and adding managerial objects
*rodm*6	6	Stopping RODM

NetView operators require RODM security level 2 or higher to use the QRS command to query whether they have span of control over resources defined using the CommandSpanName attribute in RODM.

Authorize the following:

- RODM load function

  The RODM load function requires a minimum of RODM security level 3. If your RODM loader job is run as a started procedure, you can define it to the STARTED class in the SAF product to enable it to run as a trusted user. You can define the task in the started procedure table, ICHRIN03; however, using the STARTED class is preferred.
- GMFHS procedure

  The GMFHS procedure requires a minimum of RODM security level 5.
- NetView procedure (if NetView user code accesses RODM)
- SNA topology manager

  The SNA topology manager requires a minimum of RODM security level 5. The user ID to authorize is APPNTM.
- DSIQTSK task

  The DSIQTSK task requires RODM security level 6. Define user ID DSIQTSK or the value of the ID keyword of the REP statement in the DSIQTSKI initialization member.
- Any user who submits one of these procedures:
  - RODM load function
  - NetView procedure (if NetView user code accesses RODM)
  - SNA topology manager
  - DSIQTSK task

- Any user who manipulates RODM from NetView using either RODMView panels or the RODMView command processors.

  A minimum of RODM security level 2 for the NetView domain name, if span of control is being applied to NetView management console (NMC) views and/or resources.

Connecting to RODM

When connecting to RODM, a user ID and password are part of the API request. A password is required, except when the program making the request is running in an APF-authorized library. The user ID can be specified on the connection request, or RODM can extract it from the SAF product.

You can connect to RODM with a blank user ID if the system on which RODM is installed has active RODM security. In this case, RODM extracts the user ID from the SAF product. Connecting to RODM is *not* allowed if you bypassed RODM security.

If you have RODM security active, the user ID that is associated with the connection request must be defined to your SAF security product.

For started procedures, you can define the started procedure name to the STARTED class of the SAF product. In RACF, this can also be accomplished by defining the task in the started procedure table, ICHRIN03; however, using the STARTED class is preferred.

Updating the RODM Start Procedure

NetView supplies a sample RODM start procedure EKGXRODM in PROCLIB. You can modify the RODM start procedure to match your environment. Items to consider include:

- TYPE of startup (C for cold, W for warm, or COLDFORC for cold force)
- NAME assigned to this RODM (default is the RODM procedure name)

  This name can be specified as a system symbolic.
- INIT to specify a method to run during RODM initialization (default is no initialization method)
- CUST to specify the customization member containing startup parameters

  EKGCUST is supplied as a sample.
- Correct data set names for STEPLIB, EKGLOG, EKGCUST data sets, as well as the checkpoint data sets if checkpointing is to be used
- Modification of the exit to accommodate the region size needed for RODM

  If you have the IEFUSI exit on your system, modify the exit to accommodate the region size needed for RODM. The IEFUSI exit can be used to limit:
  - The region size and region limit
  - The size of data spaces and hyperspaces for jobs started on an MVS system

  Because RODM allocates a 2 gigabyte data space at initialization, modify your IEFUSI exit to enable RODM to start. Note that IEFUSI is passed the address of the program name on input.

Defining RODM Using the EKGCUST Member

The EKGCUST member enables you to define values and tailor RODM to your installation's needs. If you choose not to customize your RODM, the default values apply.

RODM looks for an optional EKGCUST DD statement in the jobstream. EKGCUST names the input customization member. If you have defined a customization member, RODM reads it in. If not, RODM provides defaults.

You can use symbols to represent operands, values, and comments in EKGCUST if symbolic substitution is enabled on your system. Ensure that the symbols are defined in member IEASYMxx of SYS1.PARMLIB. See “RODM Customization Member Guidelines” for more information.

Note: You can specify each operand once.

After initialization is complete, you are prompted for instructions to continue with the default values or to stop the process of bringing up the RODM.

RODM Customization Member Format and Defaults

Figure 8 is an example of the format and defaults for the RODM operands that you can specify in the customization member. Except for the CELL_POOLS statement, the values in EKGCUST are the default values that are assumed if a statement is not present in the member or if the EKGCUST member is not available.

```
/*********************************************************************/
/* Number of asynchronous tasks.  The asynchronous tasks run         */
/* some of the methods that are triggered within RODM.               */
/*********************************************************************/
  ASYNC_TASKS           ( 5 )


/*********************************************************************/
/* Cell pool and cell size definition.  You can specify up to 200    */
/* cell sizes and cell pools.                                        */
/*********************************************************************/
 CELL_POOLS (  8, 12, 16, 20 )
 CELL_POOLS ( 24, 28, 32, 36:2 )
 CELL_POOLS ( 40:2, 48:2, 52:2, 56:2 )
 CELL_POOLS ( 60:2, 64:2, 68:3, 72:3 )
 CELL_POOLS ( 80:3, 88:3, 100:3, 104:3 )
 CELL_POOLS ( 112:3, 120:3, 128:3, 136:4 )
 CELL_POOLS ( 144:4, 152:4, 160:4, 168:4 )
 CELL_POOLS ( 176:4, 184:4, 192:4, 200:4 )
 CELL_POOLS ( 208:4, 216:4, 224:4, 232:4 )
 CELL_POOLS ( 240:4, 248:4, 256:4, 384:6 )
 CELL_POOLS ( 484:9, 512:8, 768:9, 1024:8 )
 CELL_POOLS ( 1536:12, 2048:16, 3072:24, 4096:32 )
 CELL_POOLS ( 6144:48, 8192:64, 12288:96, 16384:128 )
 CELL_POOLS ( 24576:192, 32768:256 )
```

Figure 8. Example of the EKGCUST Customization Input Member Syntax

RODM Customization Member Guidelines

You can code multiple operands in a single record (or line), separated by at least one blank, or you can code them individually in separate records. Blanks between the operands and the left parenthesis as well as blanks within the parentheses are ignored. For example,

```
CELL_POOLS (8,12,16,20)
```

reads the same as

```
CELL_POOLS (  8, 12, 16, 20 )
```

You can specify operands in any order, and all operands are optional. You can use symbols to represent operands and values in EKGCUST if symbolic substitution is enabled on your system. Ensure that the symbols are defined in member IEASYM*xx* of SYS1.PARMLIB.

Besides the operands specified above, you can code comments in the customization member. The beginning and closing comment delimiters, /* and */, need not appear in the same record or line. Comments are ignored by RODM and can appear any place a blank space is allowed.

You can also use symbols to represent comments if symbolic substitution is enabled on your system. Ensure that the symbols are defined in member IEASYM*xx* of SYS1.PARMLIB. Code a symbol for the beginning comment delimiter and a symbol for the closing comment delimiter.

Note: Note that the symbols for the beginning and closing comment delimiters *must* be coded on the same line in EKGCUST.

If you do not supply customization values, RODM provides the defaults for all operands. These default values should be considered as guides.

If you want information about...	Refer to...
Customizing the RODM EKGCUST operands	"Resource Object Data Manager (RODM) Definition Statements" in *Tivoli NetView for OS/390 Administration Reference*

RODM Customization Member Performance Considerations

Performance considerations differ depending upon your system environment and the other applications you have installed. You can make performance enhancements by modifying the following parameters in EKGCUST:

1. Set the following parameters to reduce the time spent acquiring and freeing storage:
   - PLI_ISA to 40K
   - PRIMARY_HEAP_SIZE to 64K
   - EXTEND_HEAP_SIZE to 32K
2. Set the CONCURRENT_USERS parameter to the maximum number of RODM user tasks you might have at the same time.

   If the number is too small, you cannot start all the RODM user applications you need. If the number is too large, you will have unused storage. Start with a value of 20 and increase or decrease as required.
3. Set the ASYNC_TASKS parameter to the number of concurrently running asynchronous tasks. This parameter controls the multiprogramming level of RODM and controls the asynchronous method API tasks.

   The default value of 5 asynchronous tasks is sufficient for most environments.

If you want information about...	Refer to...
Customizing the RODM EKGCUST operands	*Tivoli NetView for OS/390 Tuning Guide*

Defining Initialization Values for RODM DSIQTSK Task

DSIQTSKI contains the RODM initialization values for the DSIQTSK btask.

The DSIQTSK task allocates storage, reads the DSIQTSKI initialization file, and carries out RODM connections, disconnections, and checkpoint requests. This task is defined to the NetView program through DSIPARM member CNMSTYLE.

To automatically start the DSIQTSK task during NetView initialization, change INIT=N to INIT=Y in the task statement in CNMSTYLE:

```
TASK.DSIQTSK.INIT-Y
```

For each RODM you specify in DSIQTSKI you can define an initialization command processor to be invoked when the DSIQTSK task successfully connects to RODM. Specify the name of this command processor in each RODM definition statement.

The following is an example of the DSIQTSKI initialization file:

```
CMDRCVR ID=DSIQTSK
*
REP &CNMRODM,CONN=N,AO=Y,T=30
⋮
TASK TASK001
TASK TASK002
⋮
```

Notice that you can specify the RODM name with a system symbolic (&CNMRODM).

You can define up to 64 RODMs and 64 NetView tasks in DSIQTSKI to be used for command routing.

The repository definition statements (REP) specify the object repository name, as well as:

- Whether DSIQTSK connects to the repository during initialization (CONN)
- Whether the repository is the current run time RODM (AO)
- The number of seconds that DSIQTSK waits to access the repository while it is being checkpointed (T)

If you want information about...	Refer to...
The RODM automation task definition statements	*Tivoli NetView for OS/390 Administration Reference*
Using the DSIQTSK task to manage your RODMs. An example that illustrates using RODM to automate the recovery of a failed resource.	*Tivoli NetView for OS/390 Automation Guide*

Preparing GMFHS

Use the steps in this section to define GMFHS.

Updating the GMFHS Start Procedure

NetView supplies a sample GMFHS start procedure CNMGMFHS (distributed as CNMSJH10) in PROCLIB. You can modify the GMFHS start procedure to match your environment.

To update the CNMGMFHS procedure for starting GMFHS, perform the following steps:

1. Change the domain name to match your NetView domain or invoke CNMGMFHS with DOMAIN=*domain_name*.
2. For an aggregation warm start, code the AGGRST=YES parameter.

   An object-independent method (DUIFFAWS) is invoked to start methods associated with the fields related to status aggregation in the real and aggregate objects in the RODM data cache.

   The default is that DUIFFAWS is not invoked and GMFHS starts normally.
3. For a resource status warm start, code the RESWS=YES parameter.

   This causes GMFHS to bypass the normal clearing and soliciting of the initial status for all domain resources and, instead, to use the existing status within RODM.

   The default is that a resource status warm start is not invoked and GMFHS starts normally.
4. Set the ARM parameter.

   The ARM parameter controls GMFHS registration with the MVS Automatic Restart Manager (ARM).

   ***ARM** Register with ARM using the default name, which is NETVIEW@@ concatenated with the domain ID.

   name Register with ARM using a user-specified name. A valid name has the following characteristics:
   - Consists of 1–16 characters.
   - The first character cannot be numeric.
   - The remaining characters can be alphanumeric, or the following special characters: @, #, or $.
   - Alphabetic characters must be in uppercase.

   ***NOARM**
   Do not register with ARM. This is the default.

   You can also use the MVS Automatic Restart Manager to group applications together by element type. The GMFHS element type is SYSNETV4. The element type cannot be changed.
5. The SUBSYM parameter enables symbolic substitution for the dataset members that are read by GMFHS: DUIGINIT and DUIGPWLU.

   ***SUBSYM**
   Enable symbolic substitution. This is the default.

   ***NOSUBSYM**
   Disable symbolic substitution.

If you want information about...	Refer to...
Aggregation and resource status warm starts (AGGRST=YES and RESWS=YES)	"How GMFHS Uses RODM" in *Tivoli NetView for OS/390 Resource Object Data Manager and GMFHS Programmer's Guide*

Defining Initialization Values for the GMFHS Main Task

The GMFHS main task enables you to:

- Start the GMFHS host environment.
- Start and control the GMFHS host subcomponents.
- Provide an interface to the MVS operator.

- Provide reinitialization services on completion of a CONFIG NETWORK command.
- End the GMFHS host session.

The GMFHS main task initialization keywords are contained in DSIPARM member DUIGINIT. Ensure that your RODM name and user ID are correct. DUIGINIT contains the system symbolic `&CNMRODM` that can be used to set the RODM name. Refer to sample DUIGINIT for parameters that can be changed for your environment.

The NMC topology servers that are restricted from accessing GMFHS are listed in DSIPARM member DUIGPWLU. If no names are specified, all NMC topology servers can access GMFHS.

If you want information about...	Refer to...
The GMFHS host main task initialization keywords	"Graphic Monitor Facility Host Subsystem (GMFHS) Statements" in the *Tivoli NetView for OS/390 Administration Reference*

Defining the COS Gateway Autotask

The common operations services (COS) gateway autotask uses the DUIFCSGW command processor to receive commands for service points from the GMFHS scope checker OPT. The command processor routes these commands to the appropriate COS gateway service point autotask. The command processor also receives the command responses from the service point autotasks, correlates these to the appropriate commands, and returns the correlated commands to the network command manager in the GMFHS host.

The COS gateway autotask supports the GMFHS command support. COS gateway uses RUNCMD to execute commands on a service point.

If you specified `TOWER=Graphics` in DSIPARM member CNMSTYLE, the COS gateway autotask (DUIFCSGW) is started at NetView initialization. Statements are coded in DSIPARM members CNMSTYLE and CNMSTASK to facilitate the starting of the task. Do not change these statements.

The STARTCNM Graphics command also starts the COS gateway autotask.

Defining the Event Manager Autotask

The event manager autotask DUIFEAUT receives and interprets alerts from GMFHS. The event manager posts resource status provided by alerts to the appropriate objects in the RODM data cache.

If you specified `TOWER=Graphics` in DSIPARM member CNMSTYLE, the event manager autotask is started at NetView initialization. Statements are coded in DSIPARM members CNMSTYLE and CNMSTASK to facilitate the starting of the task. Do not change these statements.

Defining the Scope Checker OPT

The GMFHS scope checker OPT receives all workstation commands from the interprocess communications (IPC) host using the program-to-program interface (PPI).

The IPC component of GMFHS provides message routing both within the GMFHS address space and between the GMFHS address space (and other components). IPC does not perform application processing; but it is the originator of, and destination for, messages related to session establishment and release.

Some of the commands received by the scope checker OPT are:

- Activate, Deactivate, and Recycle commands
- Commands with NetView command model statements that might require command authorization

  For example:
  - Resource specific commands
  - Native commands
  - Generic commands

The scope checker OPT carries out NetView command authorization on these commands. Commands without proper authorization are sent with a command authorization failure reason code to the network command manager (NETCMD) in the GMFHS host.

The scope checker OPT receives NetView OST gateway and COS gateway command messages from the GMFHS NETCMD.

If you specified `TOWER=Graphics` in DSIPARM member CNMSTYLE, the scope checker OPT is started at NetView initialization. Statements are coded in DSIPARM members CNMSTYLE and CNMSTASK to facilitate the starting of the task. Do not change these statements.

If you want information about...	Refer to...
Protecting commands using scope of command authorization, the NetView command authorization table, or a SAF security product	"Controlling Access to Commands" in the *Tivoli NetView for OS/390 Security Reference*

Defining the NMCSTATUS Policy Autotask

DUIFPOLI is the autotask used to process NMCSTATUS policy definitions.

You can automatically start the NMCSTATUS policy autotask and processing of the NMCSTATUS policy definitions at NetView initialization. To do this, verify that you made the following updates in CNMSTYLE (see "Chapter 2. Enabling Graphics Support" on page 5):

- The NetvType statement is set to Enterprise.
- Graphics is uncommented on the TOWER statement.
- RODM identifier is specified on the RODMname statement.

You can change the name of the DUIFPOLI autotask to another autotask name. To do this, edit the following CNMSTYLE statement:

```
function.autotask.NMCpolicy = DUIFPOLI
```

Replace DUIFPOLI with your 8–character autotask name. When you do this, a common global variable CNMSTYLE.AUTO.NMCPOLICY is created with the new autotask name.

Note: Do not change the global variable CNMSTYLE.AUTO.NMCPOLICY in DSIPARM members CNMSTASK or DSITBL01.

If you want information about...	Refer to...
Creating and loading a policy file containing NMCSTATUS policy definitions	*Tivoli NetView for OS/390 Administration Reference*

Starting RODM Using Job EKGXRODM

To start RODM, enter the following at the system console:

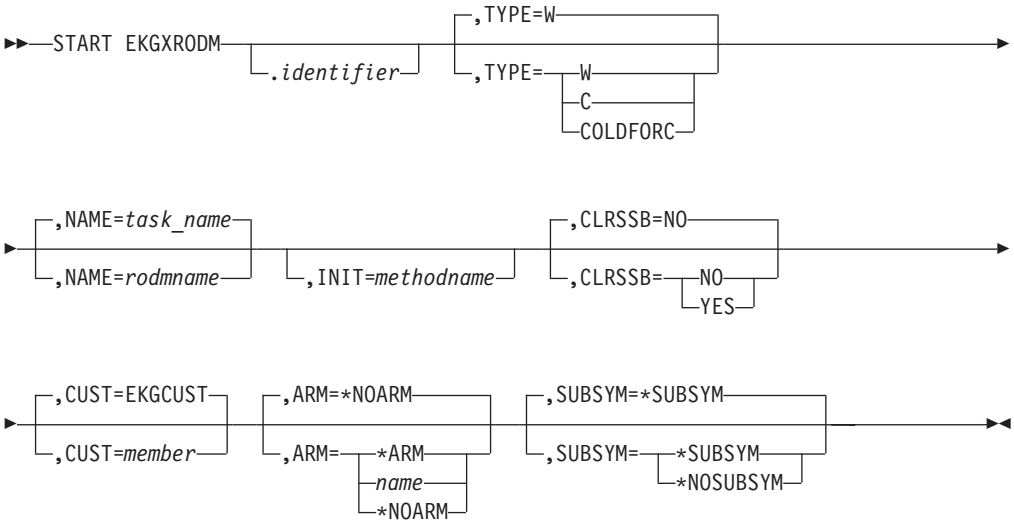
```
S EKGXRODM,TYPE=C,NAME=rodmname
```

or

```
S EKGXRODM.RODM,TYPE=C,NAME=rodmname
```

The syntax for the START EKGXRODM command follows.

START



The following are parameters for EKGXRODM:

TYPE Specifies whether to cold start or warm start RODM.

W Specifies warm start. This is the default if you do not specify a type.

Note: Because of the changes to DUIFSTRC, the first time you start RODM for the NetView program (after upgrading to V1R4), specify `C` to perform a cold start of RODM.

C Specifies cold start.

COLDFORC
Specifies cold start without issuing message EKG1918D, which requires operator intervention.

NAME Specifies the *rodmname* of the RODM program to be started. If you do not enter a value for *rodmname*, the NetView program defaults to the procedure name that you defined. The Tivoli-supplied samples use RODMNAME as the example RODM name.

INIT Specifies the name of the initialization method to run. If you leave this

parameter blank, no initialization method is run. Refer to “Writing RODM Methods” in the *Tivoli NetView for OS/390 Resource Object Data Manager and GMFHS Programmer’s Guide* for additional information on writing initialization methods.

CLRSSB

Specifies to clear MVS storage traces left by a RODM that has abended or has been forced off the system by the operator. This parameter should only be used after a RODM has been forced off the system in this manner, and then only after an attempt to restart RODM without the CLRSSB parameter has been unsuccessful.

NO Specifies not to force a restart of a RODM with the same name. This is the default.

YES Specifies to force a restart of a RODM with the same name.

Note: When using this parameter, specify `YES` only if, when you start RODM, you receive message `EKG1912 EKGXRODM: THE RODM rodmname IS ALREADY ACTIVE`, and are certain no RODM with the same name is active.

The CLRSSB keyword has no effect on the number of SSBs that are stored in the CSA area.

CUST Specifies the member name in EKGCUST DD PDS to be used as the customization file.

ARM Controls RODM registration with the MVS Automatic Restart Manager (ARM).

***ARM** Register with ARM using a RODM-generated name, which is NETVIEW# concatenated with the RODM name.

name Register with ARM using a user-specified name. A valid name has the following characteristics:

- Consists of 1–16 characters.
- The first character cannot be numeric.
- The remaining characters can be alphanumeric, or the following special characters: @, #, or $.
- Alphabetic characters must be in uppercase.

***NOARM**
Do not register with ARM. This is the default.

You can also use the MVS Automatic Restart Manager to group applications together by element type. The RODM element type is SYSNETV3. The element type cannot be changed.

SUBSYM
The SUBSYM parameter enables symbolic substitution for the RODM customization member.

***SUBSYM**
Enable symbolic substitution. This is the default.

***NOSUBSYM**
Disable symbolic substitution.

When you receive the following message:

```
EKG1918D  EKGXRODM: RODM rodmname WILL COLD START.
          ENTER '1' TO CONTINUE OR '2' TO TERMINATE.
```

Type **1** to continue.

You then receive the following two messages:

```
IEC161I 227-229,EKGXRODM,EKGXRODM,EKGD003
EKG5011I EKGXRODM : THE NUMBER OF CHECKPOINT FILES USED
         BY RODM IS 2.
```

The last parameter in the first message (EKGD003) changes, depending on the number of checkpoint files. It is always one more than the number of checkpoint files that you have specified in EKGXRODM. EKGXRODM is shipped with two checkpoint files defined.

If you are running RODM without checkpoint data sets, you receive the following message:

```
EKG1123I EKGXRODM : THE CHECKPOINT FUNCTION IS NOW DISABLED
```

To start a second RODM:

- Allocate VSAM clusters by running job EKGLOG to allocate the RODM log data sets as defined in EKGDLOG and job EKGWIND to allocate the RODM checkpoint data sets as defined in EKGDWIND.
- Specify a second *rodmname*. You cannot use the same *rodmname* (for example, RODMNAME) for the second RODM.
- Create a second RODM start procedure and modify it to use the data set names you allocated with EKGLOG and EKGWIND.
- Keep the checkpoint data sets of a RODM with the translation data sets (EKGTRAN) and the master window data sets (EKGMAST) for each RODM. After you have performed a warm start on RODM, mixing the checkpoint data sets from one RODM with the translation and master window data sets of another RODM can corrupt the RODM data cache.

After RODM is initialized, you receive the following message:

```
EKG1900I  EKGXRODM: RODM rodm INITIALIZATION IS COMPLETE WITH env
```

Note: You cannot stop RODM by ending the current NetView program. To end RODM when started using only its PROC name, enter the following at the system console:

```
F procname,TERM
```

If you started RODM using an identifier, enter the following at the system console:

```
F identifier,TERM
```

If you are running GMFHS and need to end or recycle RODM, end the GMFHS application before ending RODM.

Note: GMFHS is dependent on the GMFHS data model being present in RODM and it is present only when RODM is active. If you start GMFHS before RODM is active, or if you end RODM while GMFHS is still active, GMFHS periodically checks to see if RODM has become active. If you then start RODM, and GMFHS reconnects to RODM before the GMFHS data model becomes active, then GMFHS ends.

Verifying the RODM Installation

Start RODM using the instructions in “Starting RODM Using Job EKGXRODM” on page 23. To verify the RODM installation using job EKGSVER:

1. Edit the PARM statement in job EKGSVER in the NETVIEW.V1R4USER.INSTALL data set to reflect the *rodmname* and *userid* for your system.

   Note: The *rodmname* and *userid* are required parameters. The password is not required with *userid*. You can specify a password to your TSO user ID, but this is not recommended because the JES output log displays this password. If you do specify a password, you might want to keep this verification job in a RACF authorized data set to prevent access to the user ID passwords.
2. Run job EKGSVER.
3. Ensure that the return code is 0.

   If you specified the same user ID in DUIGINIT, you receive a return code of 8. Change the user ID in EKGSVER to fix this problem.

The following is an example of the output you receive after running EKGSVER:

```
RODM name from JCL -->  <RODM name>
User_id from JCL -->    <user id>
**No Password Specified**
====================
Function ID --- >                1101
Return code --- >                   0
Reason code --- >                   0
User ID     --- >       <user id>
Trans ID    --- >                6205
====================
Function ID --- >                1501
Return code --- >                   0
Reason code --- >                   0
Trans ID    --- >                6206
Query EKG_Name field from EKG_System class
Response    --- >       <RODM name>
====================
Function ID --- >                1501
Return code --- >                   0
Reason code --- >                   0
Trans ID    --- >                6207
Query EKG_Name field from EKG_System object
Response    --- >       <RODM name>
====================
====================
Function ID --- >                1501
Return code --- >                   0
Reason code --- >                   0
Trans ID    --- >                6208
Query EKG_ExternalLogState
Response    --- >                   1
                       *** Log Enabled ***
====================
Function ID --- >                1102
Return code --- >                   0
Reason code --- >                   0
Trans ID    --- >                6209
====================
Highest Return code -->             0
====================
```

If you want information about...	Refer to...
The parameters (FUNCTION ID, RETURN CODE, for example) listed in this output file	"Application Programming Reference" in the *Tivoli NetView for OS/390 Resource Object Data Manager and GMFHS Programmer's Guide*

Loading the Data Models

CNMSJH12 is a sample job provided with NetView that loads the GMFHS data model. It also loads RODM with the SNA topology data model class and object definitions that are required for the dynamic definition of SNA objects, as well as the MultiSystem Manager data model class and object definitions that are required for dynamic definition of the non-SNA objects. Each class contains a set of fields describing the characteristics of an object. Classes also contain presentation fields from the GMFHS data model that determine how an object is displayed in views on the NetView management console.

Job CNMSJH12 calls JCL procedure EKGLOADP to load the RODM data cache with a set of RODM class definitions and methods and sample object definitions. To load the class and object definitions using job CNMSJH12:

1. Verify that RODM is active.
2. Verify that your TSO ID is authorized if you are using a security facility, such as RACF.
3. Ensure that job CNMSJ003 has been run to copy EKGLOADP into one of your system PROCLIB data sets, and that EGKLOADP has been modified to run on your system.
4. Edit job CNMSJH12:
   - In the PARM field of the EXEC statement, replace *rodmname* with the name by which RODM is known to the applications to which it connects.
   - Ensure that the EKGIN1 DD statements include the data model members for the components you are using:

     DUIFNRM*x* NRM sample (uncomment to run)

     DUIFSNET Sample network member which can be loaded for test purposes (uncomment to run)

     DUIFSTRC GMFHS data model (required)

     FLBTRDM*x* SNA topology manager data model

     FLCSDM*x* MultiSystem Manager data model (uncomment to run)

     For more information on the data model samples, see "Appendix B. NetView Graphics Samples" on page 95.
   - To install topology correlation, remove the comment before the load statement for FLCSDM8 in the EKGIN1 data set concatenation.
5. Run CNMSJH12:
   - As a batch job, your TSO user ID or USER on the job statement must be RACF-authorized if you are using RACF to protect RODM.
   - As a started procedure, you must update the RACF started procedure table, which assigns a RACF-authorized user ID for the RODM load utility procedure if you are using RACF.
6. Ensure that the return code is 0 before proceeding.

If you need to run this job more than once to receive a return code of 0, you need to delete the classes that were allocated during the previous running. You can delete these classes by stopping and then cold starting RODM.

Note: When you cold start RODM, you delete all information in RODM.

If you want information about...	Refer to...
Defining your network resources to RODM	"Using the RODM Load Function" in the *Tivoli NetView for OS/390 Resource Object Data Manager and GMFHS Programmer's Guide*
Authorizing user IDs for RACF	"Defining NetView Security for RODM" in *Tivoli NetView for OS/390 Security Reference*.

Starting GMFHS Using Job CNMGMFHS

CNMGMFHS (CNMSJH10) is copied to PROCLIB when you load partitioned data sets during installation. The symbolics in CNMGMFHS must match the naming conventions for your system.

Complete the following before starting GMFHS:

1. Verify that RODM is active and the RODM data cache has been loaded with data models from CNMSJH12, as explained in "Loading the Data Models" on page 27.
2. Ensure that your host is designated as the focal point host on the CNMTAMEL statement in CNMSTASK. The member name is set to MEM=DUIISFP.

   Note: If you updated the CNMTAMEL task statement before you started CNMPROC, you can make the change by stopping and starting CNMTAMEL. Enter the following from the NetView command facility:

   ```
   STOP TASK=CNMTAMEL
   START TASK=CNMTAMEL,MEM=DUIISFP
   ```

To start GMFHS, enter the following at the system console:

```
S CNMGMFHS.GMFHS
```

You receive messages similar to the following:

```
DUI4027I GMFHS MAIN TASK INITIALIZATION IS COMPLETE FOR DOMAIN = domainid
DUI4003I GMFHS NETWORK CONFIGURATION INITIALIZED SUCCESSFULLY
```

The syntax for the START CNMGMFHS command follows:

START

where *domain_name* specifies the name of your NetView domain.

Verifying the GMFHS Installation

To verify that the GMFHS functions are installed correctly, issue the GMFHS STATUS command at the NetView operator console. This verifies that the GMFHS is installed correctly at the host.

You receive a response similar to the example shown in Figure 9.

```
NCCF                     Tivoli NetView    CNM01 NETOP1   2/1/01 17:03:42
C CNM01    MVS F GMFHS,STATUS
E CNM01    DUI4040I STATUS DISPLAY
E CNM01    DUI4041I RODM CONFIGURATION STATUS = COMPLETE
E CNM01    DUI4042I TYPE = CNMTAMEL  STATUS = ACTIVE     SESSION = CNM01H
           PPIST = OK
E CNM01    DUI4042I TYPE = SCOPT     STATUS = ACTIVE     SESSION = CNM01S
           PPIST = OK
E CNM01    DUI4043I TYPE = RODM      STATUS = ACTIVE     SESSION = RODMNA
E CNM01    DUI4037I END
```

Figure 9. GMFHS Status Command Results

Chapter 4. Installing and Configuring the NetView Management Console for Graphics

The NetView management console (NMC) consists of topology servers and topology consoles that communicate with each other, as shown in Figure 10 on page 32.

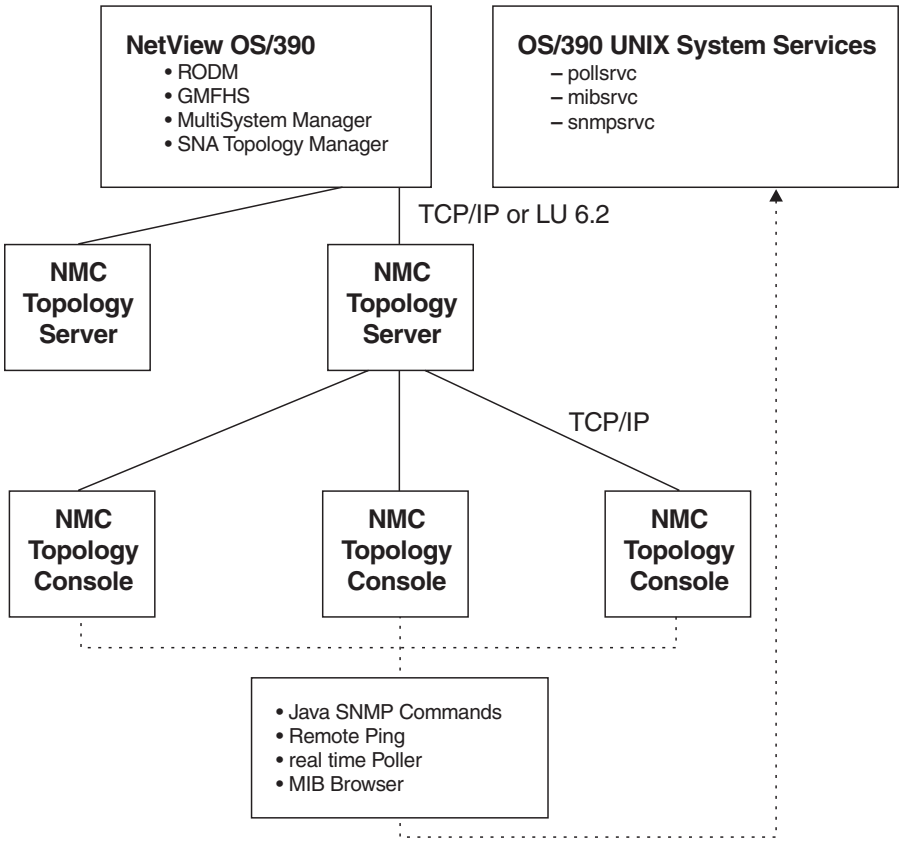


Figure 10. NMC Overview

The NMC topology server provides information to NMC topology consoles for displaying network topology and status. This information is retrieved from RODM. The NMC server also provides the NMC topology console with a set of tasks applicable to each resource type. It communicates with NetView using either an SNA LU 6.2 or TCP/IP session that has been established using the NETCONV command.

The NMC topology console graphically displays network topology and status. This includes system and network views, using color to show the status of resources. It receives this configuration and status from the NMC topology server and provides command facilities for managing monitored resources.

Installing and configuring the NMC for graphics requires definitions in NetView and the workstation environment.

Before using the MIB Browser, the Real Time Poller, and the Java SNMP commands, start the Java SNMP services. For details, refer to *Tivoli NetView for OS/390 Installation: Configuring Additional Components*.

NetView Setup for the NMC

NetView communicates with NMC servers using either TCP/IP or SNA LU 6.2.

For TCP/IP communication with NetView, TCP/IP must be enabled in member DUIFPMEM. The server workstation must be able to ping the NetView host.

For SNA LU 6.2 communication with NetView, the workstation PU and independent LUs must be defined to VTAM in a switched major node. The NMC LU 6.2 application must be defined to VTAM in a cross-domain resource major node. Refer to “Appendix A. SNA Definitions for the NMC Topology Servers” on page 85 for information on the VTAM definitions required.

Defining the Status Focal Point

This section describes how to set up connectivity between the status focal point and the NMC topology server.

CNMSJ009

Ensure that the NetView start procedure CNMSJ009 includes a DD statement for SYSTCPD that specifies the TCP/IP configuration information data set. This might already have been included when NetView was installed and configured.

CNMSTYLE

In the CNMSTYLE member, if TOWER=Graphics, the CNMTAMEL task starts during NetView initialization. Otherwise, you must start the CNMTAMEL task manually from the command line.

In the CNMSTYLE member, if TYPE=ENTERPRISE, the CNMTAMEL is initialized as a status focal point. Any other ″TYPE″ does not allow CNMTAMEL to have status focal point capabilities.

DUIISFP

This member defines the initialization values for the CNMTAMEL task. It contains the default values for each operand.

DUIFPMEM

This member is defined on the AMELINIT statement in the CNMTAMEL task definition member DUIISFP and initializes the CNMTAMEL task at your status focal point.

In order to enable NETCONV sessions across TCP/IP to NMC, specify USETCPIP=YES in DUIFPMEM. The default value shipped with NetView is NO.

You can change the values of the following keywords in DUIFPMEM to meet the requirements of your network.

The keywords are:

PORT Specifies the port used by the status focal point host for TCP/IP communication. This is not a required keyword. The default is 4020.

SOCKETS
Specifies the maximum number of sockets that this status focal point host uses for connecting to NMC workstations. This is not a required keyword. The default is 50.

TCPANAME
Specifies the TCP/IP application procedure name that the status focal point host uses. This is a required keyword for the TCP/IP function.

USETCPIP
Specifies that a status focal point host uses TCP/IP to communicate with NetView management console workstations. This is not a required keyword. Valid values are Y, YES, N, and NO. The default is NO.

If you want information about...	Refer to...
Keywords	*Tivoli NetView for OS/390 Administration Reference*

DUIIGHB

This DSIPARM member is used for the initialization of the DUIDGHB task. This task is used to resolve TCP/IP addresses and names including:

- Host names and addresses for establishing IP NETCONV sessions
- Host names for issuing IP REMOTECMD commands
- Host names and addresses of host destinations for the NetView Resource Manager

The keyword for DUIIGHB is:

TCPANAME
Specifies the TCP/IP application procedure name that the status focal point host uses. This is a required keyword for the TCP/IP function.

Note: You can also set the TCP/IP name as follows:

- As a system symbolic (&CNMTCPN) in SYS1.PARMLIB
- On the TCPname statement in CNMSTYLE

Security Considerations

You can restrict which operators perform administrative functions for the NetView management console. You can also restrict which commands operators can issue from the popup menus.

Using NGMFADMN

The NGMFADMN attribute can be used in NetView operator profiles and in the NETVIEW segment of the SAF product. It specifies whether operators are allowed to perform administrative functions for the NetView management console. Some examples of functions controlled by this keyword include:

- Using the command profile editor
- Adjusting aggregation for individual resources
- Customizing views
- Customizing settings and applying them to all topology consoles

NO
The operator does not have administrative authority for the NetView management console. This is the default.

YES
The operator has administrative authority for the NetView management console.

Using NGMFCMDS

The NGMFCMDS attribute can be used in a NetView operator profile only. It cannot be specified using the NETVIEW segment of the SAF product. It specifies whether operators are allowed to issue commands from the popup menus of the NetView management console. NGMFCMDS does not prevent operators from typing commands in a NetView command line window.

YES
The operator is allowed to issue commands from the popup menus on the NetView management console. This is the default.

NO
The operator is not allowed to issue commands from the popup menus on the NetView management console.

Workstation Setup for the NMC

NetView can communicate with NMC servers using either TCP/IP or SNA LU 6.2.

After installing the workstation code, it must be configured to enable communication with the host components of the NMC.

Installing and Configuring the NMC Topology Server

The NMC topology server can be installed on a system running Windows NT, Windows® 2000, AIX, or OS/2.

To install the NMC topology server, refer to the EGVREAD2.ME file on the product CD or on the Tivoli web site.

After the server is installed, do the following:

- If you are using TCP/IP sessions, see "Defining IP Socket Port Numbers for TCP/IP". Otherwise see "Appendix A. SNA Definitions for the NMC Topology Servers" on page 85.
- See "Storing Topology Server Databases" on page 36.
- For Windows NT/2000 server users, see "Setting Security for Your Operating System" on page 38 and "Configuring the Topology Server to Run as an NT/2000 Service" on page 39.
- For AIX server users, see "Configuring the Topology Server to Run as a UNIX Daemon" on page 39.

Enabling the Topology Display Subsystem View to Show NetView Components

NetView instrumentation must be enabled to monitor the status of the components that provide NMC views.

The Topology Display Subsystem view shows the status of the components that provide NMC views. To open this view from the Business tree, do the following:

1. Click the plus sign next to Business Systems.
2. Double-click Systems Management Business System.

The resulting view shows the topology consoles, topology servers, NETCONV connections, Graphic Monitor Facility host subsystem (GMFHS), RODM, and the RODM managers. Without NetView instrumentation, this view shows only one topology server, and the topology consoles signed onto that server.

If you want information about...	Refer to...
Enabling NetView instrumentation (INITAMI, DSIAMII, and DSIAMIAT).	*Tivoli NetView for OS/390 Customization Guide*

Defining IP Socket Port Numbers for TCP/IP

Communication with the NetView environment is achieved through either TCP/IP or LU 6.2 (as described in "Defining the Logical Unit (LU) Name" on page 88). The topology server communicates through TCP/IP with the topology console. The topology server default uses the following IP socket ports:

- Port 4000 for NMC console communications
- Port 4020 for NetView communications

No action is required if these port numbers do not conflict with your current application settings. However, if there is a conflict, change the port number or numbers using these instructions:

1. On the topology server workstation, open a command window.
2. Change to one of the following directories:
   - For Windows NT/2000: `%windir%\system32\drivers\etc`

     Where `%windir%` is usually WINNT or WINNT40. You can display the value of `%windir%` by entering **set windir** from a command prompt.
   - For OS/2: `%etc%`

     You can display the value of `%etc%` by entering **echo %etc%** from an OS/2 command prompt.
   - For UNIX®: `/etc`
3. Add the appropriate entry to the **services** file. Substitute your port number for the following default port number(s):

```
tserver_console 4000/tcp  #NetView management console server console port
tserver_390     4020/tcp   #NetView management console server 390 port
```

Notes:

1. If you reassign the topology server console port, you must specify the port number on the NMC **Sign On** window each time you sign on. However, the topology console retains the *server:port* setting from the last sign on and uses it as the default the next time you sign on.
2. If you reassign the topology server 390 port, you must also customize the NetView **Port** keyword in the DUIFPMEM file. Refer to the *Tivoli NetView for OS/390 Administration Reference* for more information on the **Port** keyword.

Storing Topology Server Databases

Depending on your operating system, the topology server uses one of the following directories to store its databases:

- Intel: `%BINDIR%\TDS\server\db`
- UNIX: `$BINDIR/TDS/server/db`

Notes:

1. If you are running NMC in the Japanese AIX environment, the default locale is `shift JIS` with a locale value of `Ja_JP`.

   The EUC locale (`ja_JP`) is also supported. If you have configured your system on either locale and decide to change locales, you must convert the databases to the new locale. The `JA_JP` locale is not supported. Several database samples are provided to assist you if you would like to change locales. The following shell scripts are provided:
   - `convert_ja_JP_databases.sh` converts the databases from the `Ja_JP` to the `ja_JP` locale and preserves your databases.
   - `convert_Ja_JP_databases.sh` converts the databases from the `ja_JP` to the `Ja_JP` locale and preserves your databases.
   - `setup_ja_JP_databases.sh` creates databases with the `ja_JP` locale and deletes any existing databases.
   - `setup_Ja_JP_databases.sh` creates databases with the `Ja_JP` locale and deletes any existing databases.

2. If your server and client do not have identical locales, some of the command set dialogs for specific resources may appear corrupted, as follows:
   - For AIX: If your client is `Ja_JP` or `shift JIS` and your server is `ja_JP` or `EUC` some of your command set dialogs for specific resources will appear corrupted. The same is true if your client is `ja_JP` and your server is `Ja_JP`.
   - For HP UX: Resources can appear corrupted if your locales are named `ja_JP.SJIS` and `ja_JP.eucJP` which correspond to the `Ja_JP` and `ja_JP` locales respectively.
   - For Solaris: Only a `ja` locale corresponds to an `EUC` locale.
   - For OS/2 and Windows: Only a `shift JIS` locales exists.

   To avoid problems with dialogs that appear to be corrupted, use servers and clients that run with the same locale settings.

No action is required if these directory locations are satisfactory. However, to use a different directory or disk for your databases, perform the following steps to change the directory structure:

1. Update the **TSERVER_DB** environment variable.
   - NT: Set the **TSERVER_DB** environment variable at the user ID level using the following steps:
     a. Log on to the ID where the topology server is running.
     b. Right-click on the **My Computer** icon.
     c. Select **Properties**.
     d. Select **Advanced**.
     e. Select the **Environment Variables** tab.
     f. Click on **New** in the User Variables section.
     g. Change the variable name to **TSERVER_DB**.
     h. Type the value of the new directory.
     i. Click **OK**.
     j. Click **OK**.
     k. Click **OK**.
   - Windows 2000: Set the **TSERVER_DB** environment variable at the user ID level using the following steps:
     a. Log on to the ID where the topology server is running.
     b. Right-click the **My Computer** icon.
     c. Select the **Properties** from the resulting pull-down window.
     d. Select **Advanced** tab on the System Properties panel.
     e. Select **Environment Variables...** button on the System Properties panel.
        The Environment Variables panel displays.
     f. Click the **New...** button in the User Variables for Administrators section of the Environment Variables panel .
        The New User Variable panel displays.
     g. Type `TSERVER_DB` in the **Variable Name** text field.
     h. In the **Variable Value** field, type the name of the new directory where you want the topology server databases to reside.
     i. Click **OK**.
     j. Click **OK**.
     k. Click **OK**.

- OS/2: Update the **TSERVER_DB** environment variable in the config.sys file.
- UNIX: Update the **TSERVER_DB** environment variable in the appropriate profile.

2. Under your new directory, create the same directory structure as the one that is presently under one of the following:
   - Intel: `%BINDIR%\TDS\server\db`
   - UNIX: `$BINDIR/TDS/server/db`
3. Ensure the topology server is stopped.
4. Run the `tserver dbtransfer` command. This copies the contents of the databases from the default installation subdirectories to the subdirectories set by the TSERVER_DB variable. The `tserver dbtransfer` command prompts you for a confirmation before overwriting files in each subdirectory.

   Notes:

   a. When the server is started for the first time and it detects the TSERVER_DB environment variable, but the command `tserver dbtransfer` was not run, the server shuts down and you receive a message to run `tserver dbtransfer`.

   b. The `tserver dbtransfer` command does not erase or overwrite the files in the default installation database subdirectories.

Setting Security for Your Operating System

If you are using Windows NT or Windows 2000, additional security configuration steps are required. No additional security configuration is required for UNIX platforms.

For Windows NT or Windows 2000, add the following user rights to the user ID under which the topology server runs so that it does the following:

- Acts as part of the operating system.
- Logs on as a service.

  This is optional and only needs to be set if the topology server is run as an NT service.(See “Configuring the Topology Server to Run as an NT/2000 Service” on page 39 for instructions.)

Follow these instructions to set user rights for Windows NT:

1. Log on to an ID with administrator authority.
2. Select **Start –> Programs –> Administrative Tools –> User Manager** (or **User Manager for Domains** for NT servers).
3. Select **Policies –> User Rights.**
4. Check the **Show Advanced User Rights** check box.
5. Scroll through the **Right:** list and select **Act as part of the operating system**.
6. If your target user ID is already listed in the **Grant to:** list, go to step 11 on page 39; otherwise continue.

   The **Grant To:** list displays the groups and users that are presently granted this right.
7. Click **Add**.
8. Click **Show Users**.
9. Select the target user ID (the one under which the topology server will run) and click **Add**.
10. Click **OK**.

11. If the topology server is to run as a service, repeat steps 5 on page 38 through 10 on page 38 for **Log on as a service**.
12. Log off and log back on the target ID for the user rights to take effect.

Follow these instructions to set user rights for Windows 2000:

1. Log on to an ID with administrator authority.
2. Select **Start –> Settings –> Control Panel –> Administrative Tools –> Local Security Policy –> Local Policies –> User Rights Assignment**.
3. Select **Act** as part of the operating system.
4. Select **Action –> Security –> Add**.
5. Add your target user ID under which the topology server will run.
6. Select **OK** twice.
7. If the topology server is going to be run as a service, repeat step 3 through step 6, and select **Log on** as a service.
8. Log off and log back on the target ID for the user rights to take effect.

Configuring the Topology Server to Run as an NT/2000 Service

To run the topology server as a Windows NT or Windows 2000 service, perform the following steps:

1. On the topology server machine, open a workstation command window.
2. Change to the following directory: `%BINDIR%\TDS\server\bin`.
3. Enter the following command to install the topology server as an NT service:

   ```
   tserver service account_name password
   ```

   Where:

   account_name
   An account name in the form `DomainName\UserName`. If the account belongs to the local domain, you can specify `.\UserName`.

   password
   The corresponding password for the specified `account_name`.

   The topology server installs as two services: the topology server and the topology communications server, and the startup option will be **manual**.
4. To change the startup to **automatic**, use the NT/2000 Service control applet.
5. To delete the services, enter the **ihsxsrv delete** command from the same directory.

Configuring the Topology Server to Run as a UNIX Daemon

For UNIX, to start the topology server processes at system startup and have them run as daemons, issue the following command:

```
$BINDIR/TDS/server/bin/config -d
```

The processes start the next time the machine is booted.

Note: You must be the root user to issue this command.

Installing the NMC Topology Console

The NMC topology console requires TCP/IP communication capability. You can run the NMC topology server and topology console code on the same machine. However, this is not recommended in a production environment.

For information on installing the NMC topology console, refer to the EGVREAD1.ME file on the product CD.

Using Java SNMP Services with NMC Applications

The NMC applications that you can use include:

- MIB Browser
- Real Time Poller
- Java SNMP Commands
- Remote Ping

These NMC applications use MIB information and query data from SNMP agents in your network.

Using the MIB Browser, you can navigate through MIBs. The MIB Browser shows the structure of the MIB and MIB variables that can be queried. This browser also supports MIB Groups.

Java SNMP commands support provides a command line interface. You can issue SNMP commands to a selected resource running the SNMP agent and get a response. Remote Ping support sends an SNMP query to remote OS/390, NetView for AIX, or HP OpenView. The application that you queried then pings the IP resource in your network.

The Real Time Poller/Grapher enables you to view a real-time graph of the performance data of polling objects for the specified host. Data for multiple polling objects, from multiple hosts, can be combined on a single graph. You can also specify the polling interval for the polling objects and the maximum number of points that appear on the graph.

Before using the MIB Browser, the Real Time Poller, and the Java SNMP commands, start the Java SNMP services on OS/390 for Unix..

Services on the Java SNMP services include:

- MIB service
- Polling service
- SNMP service

You can start these services by one of the following methods:

- Through NetView
- Directly from OS/390 UNIX System Services
- Using the NMC Java Application Server (JAS)

To install and start the Java SNMP services, refer to *Tivoli NetView for OS/390 Installation: Configuring Additional Components*.

Enabling the MIB Browser

The MIB Browser can be launched for a resource which:

- Is managed by SNMP
- Has an IP address

The MIB browser can be launched for NMC-managed MultiSystem Manager resources.

Note: To have MultiSystem Manager resources displayed by NetView management console, MultiSystem Manager must be loaded and running.

For more information about MultiSystem Manager, see *Tivoli NetView for OS/390 MultiSystem Manager User's Guide*.

To enable the MIB Browser, perform the following tasks:

1. Start the Java SNMP services

   Note: MIB service (MIBSRVC) and SNMP service (SNMPSRVC) must already be started.

   The NetView program loads the information from the etc/netview/nv390srvr.conf file as global variables and uses this information when the MIB Browser is launched.

   For details about starting the Java SNMP services, refer to *Tivoli NetView for OS/390 Installation: Configuring Additional Components*.

   CAUTION:
   Although it is possible for you to perform the following step, it is neither required nor recommended!
2. **Optional:** It is possible to copy and change the connections specifications for the MIB Browser. To do this, copy `/etc/netview/nv390srvr.conf` to `$(bindir)/TDS/server/db/current/settings/defaultmibbr.properties` on the NMC topology server.

   Note: This step overrides the NetView global variables that you loaded in Step 1.
3. **Optional:** Define security authorizations for the MIB browser by creating the NetView span-of-control security authorization tables.

   Refer to the″NetView Commands, Keywords, and Values that Can Be Protected″ chapter in the Tivoli NetView for OS/390 Security Reference for information about protecting these commands.
4. **Optional:** Create or modify a .grp file and load it.

   You perform this task so that you can use MIB groups. You can do this in one of two ways:
   - Using the **loadmib** command
   - Including the .grp in the definitions file

     You use this when you start MIBSRVC (such as the /etc/netview/nv390mibs.def file on OS/390 UNIX System Services).

   Note: In order for the groups functions to work properly, you must also load the MIB files that contain the definitions of the MIBs that you include in your .grp file.

   A sample .grp file, /etc/netview/samples/fkxsnmp.grp, is included. Sample .mib files included are rfc1213-MIB-II.mib, rfc1903.mib, and rfc1907.mib.

You can now start the MIB Browser from the NetView management console.

Enabling the Real Time Poller

The Real Time Poller can be launched for a resource which:

- Is managed by SNMP

- Has an IP address

Start the Real Time Poller from NMC-managed resources.

Note: To have MultiSystem Manager resources displayed by NetView management console, MultiSystem Manager must be loaded and running.

For more information about MultiSystem Manager, see *Tivoli NetView for OS/390 MultiSystem Manager User's Guide*.

To enable the Real Time Poller, perform the following tasks:

1. Start the Java SNMP services

   Note: Poll Service (POLLSRVC) must already be started.

   NetView loads the information from the /etc/netview/nv390srvr.conf file as global variables and uses this information when the Real Time Poller is launched.

   For details about starting Poll Service, refer to *Tivoli NetView for OS/390 Installation: Configuring Additional Components*.

   CAUTION:
   Although it is possible for you to perform the following step, it is neither required nor recommended!
2. **Optional:** It is possible to copy and change the connections specifications for the MIB Browser. To do this, copy `/etc/netview/nv390srvr.conf` to `$(bindir)/TDS/server/db/current/settings/defaultmibbr.properties` on the NMC topology server.

   Note: This step overrides the NetView global variables that you loaded in Step 1.
3. **Optional:** Define security authorization for the Real Time Poller by creating the NetView span-of-control security authorization tables.

   Refer to the"NetView Commands, Keywords, and Values that Can Be Protected" chapter in the *Tivoli NetView for OS/390 Security Reference* for information about protecting these commands.

You can now start the Real Time Poller from the NetView management console.

Enabling Java SNMP (JSNMP) Commands

The Java SNMP (JSNMP) commands can be launched for NMC-managed resources which:

- Are managed by SNMP
- Have an IP address

Note: To have MultiSystem Manager resources displayed by NetView management console, MultiSystem Manager must be loaded and running.

For more information about MultiSystem Manager, see *Tivoli NetView for OS/390 MultiSystem Manager User's Guide*.

To enable the Java SNMP commands, perform the following tasks:

1. Start the Java SNMP services (you need MIBSRVC and SNMPSRVC started).

Note: MIB service (MIBSRVC) and SNMP service (SNMPSRVC) must already be started.

NetView loads the information from the /etc/netview/nv390srvr.conf file as global variables and uses this information when the Java SNMP commands are launched.

For details about starting these services, refer to *Tivoli NetView for OS/390 Installation: Configuring Additional Components*.

CAUTION:
Although it is possible for you to perform the following step, it is neither required nor recommended!

2. **Optional:** It is possible to copy and change the connections specifications for the MIB Browser. To do this, copy `/etc/netview/nv390srvr.conf` to `$(bindir)/TDS/server/db/current/settings/defaultmibbr.properties` on the NMC topology server.

   Note: This step overrides the NetView global variables that you loaded in Step 1.

3. **Optional:** Define security authorizations for the Java SNMP commands by creating the NetView span-of-control security authorization tables.

   Refer to the“NetView Commands, Keywords, and Values that Can Be Protected” chapter in the *Tivoli NetView for OS/390 Security Reference* for information about protecting these commands.

You can now start the Java SNMP (JSNMP) commands and Remote Ping.

Enabling Remote Ping

Remote Ping uses remote OS/390, NetView for AIX, or HP OpenView to send pings to the IP resource in your network. Remote Ping can be launched for NMC-managed MultiSystem Manager resources which:

- Are managed by SNMP
- Have an IP address

Note: To have MultiSystem Manager resources displayed by NetView management console, MultiSystem Manager must be loaded and running.

For more information about MultiSystem Manager, see *Tivoli NetView for OS/390 MultiSystem Manager User's Guide*.

Before using Remote Ping you might have to customize your remote platform, such as OS/390, NetView for AIX, or HP OpenView. For more information about enabling the Remote Ping facility, refer to the documentation for your remote platform.

If Java SNMP commands are enabled, Remote Ping is also enabled.

Enabling OSNMP Commands

OSNMP commands are provided for backward compatibility and they use NetView NVSNMP commands. OSNMP Commands with the -jsnmp option and previously mentioned Java SNMP (JSNMP) commands are compatible.

OSNMP commands with the -jsnmp option provide the same functionality as NVSNMP commands with -jsnmp option.

We recommend that you use Java SNMP commands because they offer better performance than the OSNMP commands with the -jsnmp option.

If you require only SNMP command support (excluding the Java SNMP TRAP command) and if you do not require the Real Time Poller, MIB Browser or the NMC remote ping function, consider enabling NVSNMP commands, which do not require the NetView Java SNMP services.

For more information, refer to *Tivoli NetView for OS/390 Installation: Configuring Additional Components*.

Configuring NMC for the NetView 3270 Management Console

The NetView 3270 management console enables you to access the NetView command facility and full-screen applications using a window on your NMC topology console.

The NetView 3270 management console code is shipped with the NetView management console.

Follow these instructions to configure the NetView 3270 management console:

1. The file FLBREAD1 (English) or FLBREAD2 (Japanese) contains information about configuring the NetView 3270 management console. You can find this file in one of the following places:
   - In the README folder under the ENU (for English) or JPN (for Japanese) subdirectories on the NetView CD-ROM
   - From the NetView Web page: http://www.tivoli.com/nv390_supported

   Notes:

   a. In order for NMC to launch the NetView 3270 management console, it must be installed on each topology console machine.

   b. The NetView 3270 management console can only be accessed by operator IDs that, when they are defined, have the OPERID statement in member DSITCPRF (in DSIPRF).

      Each operator ID must be assigned a pair of encryption keys set to **default** to send and receive information between NetView and the console.

   c. For the NetView 3270 management console to communicate with the OS/390 host running NetView, the NetView DSITCPIP task must be started.

      This NetView task manages TCP/IP communication.
2. Enable the menu item by right-clicking on the view background.
   - A command profile editor (CPE) response file, `ihssnv390cons.rsp`, is shipped with NMC to enable the NetView 3270 management console on view backgrounds.

     a. Edit the response file and follow the directions in this response file.

        Remember that the TCP/IP port number must match the port number in DSITCPCF in NetView; the default is 9999.

     b. After you edit the response file, run the CPE batch utility (once) to enable the NetView 3270 management console menus from NMC.

        For example, to run the CPE batch utility, perform the following steps:

        1) Change to one of the following directories:
           – For Intel: %BINDIR%\TDS\server\bin
           – For UNIX: $BINDIR/TDS/server/bin

2) Enter one of the following commands:
   – For Intel: **cpebatch ..\sample\ihssnv390cons.rsp -i -g**
   – For UNIX: **cpebatch ../sample/ihssnv390cons.rsp -i -g**

- Optional parameters can be specified for each NetView 3270 management console menu item.

  These include:

  – The home session (for example, NCCF and NPDA)
  – An HACL application that is automatically executed

  For more information, refer to the comments in the command profile editor (CPE) response file, ihssnv390cons.rsp, which is shipped with NMC. For information on coding HACL applications, refer to the *Tivoli NetView for OS/390 Customization Guide*.

There are two ways to launch the NetView 3270 management console:

- Select a NetView resource in the Topology Display Subsystem view and then select **NetView 3270 Management Console** from the context menu.

  NMC launches the NetView 3270 management console application. The selected resource must represent a NetView host.
- Right-click on a view background, select **NetView 3270 Management Console** and then select the menu item corresponding to the NetView that you want to access.

In either case, if a session already exists with the target NetView 390 host, that session is automatically reused instead of starting an additional operator session.

If you do not have the NetView 3270 management console installed, you can remove the NetView 3270 Management Console from the context menu for resources in the Topology Display Subsystem view.

To disable this menu item, use the ihssdisnv390cons.rsp CPE response file, which is shipped with NMC.

To disable the NetView 3270 Management Console menu item, run the CPE batch utility (once) by performing the following steps:

1. Change to one of the following directories:
   - For Intel: %BINDIR%\TDS\server\bin
   - For UNIX: $BINDIR/TDS/server/bin
2. Enter one of the following commands:
   - For Intel: **cpebatch ..\sample\ihssdisnv390cons.rsp -i -m**
   - For UNIX: **cpebatch ../sample/ihssdisnv390cons.rsp -i -m**

Verifying the NetView Management Console Installation

To verify the NetView management console installation, start the NMC topology server and then start an NMC topology console as described in the following sections.

Begin Communication with the NMC Topology Server

To begin communication between the NMC topology server and NetView, issue the NETCONV command from an autotask. For example, to start communication using TCP/IP, enter:

```
EXCMD AUTO1 NETCONV,ACTION=START,IP=hostname
```

To start communication using LU 6.2, enter:

```
EXCMD AUTO1 NETCONV,ACTION=START,LU=luname
```

Note: You can issue the NETCONV command from an OST, but the connection is cancelled when the OST logs off.

Starting the Topology Server

You can start the topology server in the following ways;

Note: This task is usually completed by a system administrator for all topology console operators. For specific information, refer to the *Tivoli NetView for OS/390 NetView Management Console User's Guide*.

- manually
- as a daemon when using UNIX
- as a Windows service when using Windows NT or Windows 2000

Installing and Configuring the NetView Inventory Server

You can use NMC to retrieve inventory data from the Tivoli inventory server for managed resources for which NetView has an IP address. The Tivoli inventory server resides on a managed node where Tivoli inventory is installed. To determine if Tivoli inventory is installed, issue the **wgetinvdata** command. If the command is recognized, the inventory server is installed. If the command is not recognized, install the Tivoli inventory server. For installation procedures, refer to the *Tivoli Inventory User's Guide*.

Using the Tivoli inventory interface, you can specify which fields are required when a user issues the GET INVENTORY command. You can also define queries in addition to those provided by Tivoli inventory. You can designate which query names are to be displayed in the popup menu by modifying the TivoliInventory.properties file. The inventory administrator performs scans to populate the inventory database prior to using this function for the first time.

The NMC server has a sample NMC TivoliInventory.properties file. It is located in the following directory:

- **For Intel:** %BINDIR%\TDS\server\db\current\settings
- **For UNIX:** $BINDIR/TDS/server/db/current/settings

The TivoliInventory.properties file contains the following entries:

```
INV_Q.1=INVENTORY_HARDWARE
INV_Q.2=INVENTORY_SOFTWARE
INV_Q.3=INVENTORY_HARDDISK
INV_Q.4=INVENTORY_MEMORY
```

You can modify this file as follows:

1. Add or delete query names to this list.

   As you modify the file, consider the following statements:

   - INV_Q.1 is the default query.

     If no INV_Q entry is found, INVENTORY_HARDWARE is used.
   - INV_Q list must begin with .1 and subsequent entries must be consecutive numbers.

- Query names must match the query names available on the Tivoli inventory server.

2. Update the HOSTNAME entry with the name of the Tivoli managed server where the Tivoli inventory server is installed. For example:

   ```
   HOSTNAME = myTIServer.mynode.com
   ```

   where `HOSTNAME = myTIServer.mynode.com` is the HOSTNAME of the machine where the Tivoli inventory server is installed.
3. If necessary, update the port.

If you want information about...	Refer to...
Tivoli Framework to install software	*Tivoli Inventory User's Guide*
Gathering inventory data on IP resources	*Tivoli NetView for OS/390 NetView Management Console User's Guide*

Starting the Topology Console

You can start the topology console from a desktop icon in Windows and OS/2 or by issuing the command in line mode from one of the following:

- Any operating system
- The Tivoli Desktop

For specific information, refer to the *Tivoli NetView for OS/390 NetView Management Console User's Guide*.

Signing On to the NMC Topology Console

Sign on to the NMC topology console and do the following:

- Connect to the topology server
- Display views

If you want information about...	Refer to...
Signing on and using the NMC topology workstation	*Tivoli NetView for OS/390 NetView Management Console User's Guide*
Online help	NMC topology console help index

Customizing the NMC

You can customize the NMC for your environment. You can add or change the following:

- Topology console icons
- Topology console views

  This can include creating views and aggregate resources (RODM Collection Manager), changing the background image, and moving resources and text labels.
- Topology console help
- Server.properties configuration file for the topology server
- Time and date format

You can also use plug-ins and Java applications for additional customization.

If you want information about...	Refer to...
Customizing the NMC	“Customizing the NetView Management Console” in the *Tivoli NetView for OS/390 NetView Management Console User’s Guide*
Java applications and plug-ins	“Topology Console Java Applications and Plug-ins” in the *Tivoli NetView for OS/390 NetView Management Console User’s Guide*

Chapter 5. Installing and Configuring SNA Topology Manager Agents

The key to SNA topology management is the SNA Topology Manager communication with Common Management Information Protocol (CMIP) topology agents. Manager-to-agent communication is through LU 6.2 sessions using the CMIP services component of VTAM and SNA MDS (Multiple Domain Support). There are three types of topology agents:

1. VTAM topology agent

   The VTAM topology agent is a standard component of VTAM. The agent collects topology and status data for subarea and APPN resources.
2. CM based APPN Topology and Accounting agent (APPNTAA)
3. IBM 3746 Models 900 and 950 Network Node Processor feature that can be set up to include the APPN topology agent

   The CCM (Controller Configuration and Management) facility can be used to do this. It is an OS/2–based agent.

The APPN Topology and Accounting agent (APPNTAA) is available from the Tivoli web site (http://www.tivoli.com/nv390_supported). For installation instructions, perform the following steps:

1. Select **APPN Topology and Accounting Agent**.
2. Select either **Download** or **View** for instructions on downloading the agent package.
3. Unzip the file and look at the BNTREAD1.ME file, which is inside the agent package.

   This file contains the remaining instructions for installing the package.

Planning for Implementation of Topology Monitoring in Your Network

To plan for the implementation of topology monitoring in your network, perform the following steps:

1. Activate the SNA topology manager at focal points where RODM is installed.
2. Activate the VTAM topology agent on each VTAM node to manage SNA.
3. At a minimum, place one APPN topology agent on a network node in each APPN-capable subnetwork to collect network topology; this enables you to view all network nodes in each subnetwork and the transmission groups (TGs)between these network nodes.
4. Place an APPN topology agent at each network node and end node to collect their local topology. This enables you to view the ports, links, and TG connections to adjacent nodes. For APPN topology agents, it also enables you to activate and deactivate the ports and links.

The APPN accounting-manager application, in conjunction with APPN accounting agent application for OS/2 APPN environments, includes the following functions:

- Provides operator commands to start and stop the collection of APPC session and conversation accounting data at agent nodes

- Retrieves APPC session and conversation accounting information from agent applications in the network
- Formats the data for output to the SMF external record log

Planning for Implementation of SNA APPN Accounting in Your Network

To plan for implementation of SNA APPN accounting in your network, perform the following steps:

1. For each part of your APPN network, decide whether you need conversation data or session data.
   - If you need conversation data, place an agent at one end of the conversation.
     - If you have a database for which you want to account for access or measure use, place an agent at the server's node.
     - If the user is on an agent node (not a dial-in user), collect conversation data directly at the user's node.
   - If you can use intermediate session data, place an agent at a network node on the path of the session. If your network is structured so that the network traffic you want to measure all passes through a small group of network nodes, place agents at these critical points. Session data can also be collected at the end points of the session, which can be either a network node or end node.
2. Place the APPN accounting agents at nodes that have the processing and storage capacity to handle the extra overhead.
3. Consider running additional APPN accounting managers:
   - If you gather a great deal of accounting data and want to improve the performance of an individual accounting manager. Multiple accounting managers can collect the same or different data from the same agent.
   - If there are advantages, for example, for an accounting manager on one node to retrieve session data, whereas a manager on another node retrieves conversation data, all from the same agents.
   - For backup of critical data.

   If you have several accounting managers gathering data from the same agents, remember that each agent can control a maximum of 10 accounting management control objects.

One common situation is a business that provides the network services for external customers. For that business, place an agent on the network nodes that its customers dial to access its network.

Another typical situation is a business that owns all the resources for which it wants to charge groups or departments. That business can place an agent on the nodes to charge for access to those resources.

If you want information about...	Refer to...
APPN Topology and Accounting agents	*Tivoli NetView for OS/390 APPN Topology and Accounting Agent Guide*
Placing the APPN agents and managers	*Tivoli NetView for OS/390 SNA Topology Manager and APPN Accounting Manager Implementation Guide*

Chapter 6. Preparing and Activating the SNA Topology Manager and the APPN Accounting Manager

This section describes the necessary steps to enable the SNA topology manager and the APPN accounting manager:

- "VTAM Setup"
- "Security Considerations" on page 52
- "Automation Options" on page 52
- "Initialization File" on page 52

This chapter also describes how to start and verify the SNA topology manager and the APPN accounting manager.

VTAM Setup

VTAM needs APPL definitions for both the SNA topology manager and the APPN accounting manager applications. Sample definitions are included in member A01APPLS (CNMS0013). The APPL name for the SNA topology manager and the APPN accounting manager must match the APPLNAME specified in the *VTAM* section of the FLBSYSD and FLBSYSDA initialization files used by the manager applications.

Ensure that the VTAM procedure includes DD statements for ISTCMIP, ACYGDMO, and ISTASN1. In addition, the VTAM CMIP agent needs to be started prior to the manager applications. At a minimum, this requires the OSIMGMT start option in VTAM to be specified as YES. NetView issues the appropriate MODIFY command to VTAM to enable OSIMGMT if the SNATM subtower is enabled in CNMSTYLE.

VTAM agent local topology can report many switched logical lines and PUs. This can increase the network traffic and the manager-to-agent work load. CMIP filtering provides improved control. CMIP filtering allows an entire major node and all its subordinate lines and PUs to be suppressed (not reported in local topology). To enable filtering, VTAM definitions for XCA, NCP, and SWNET nodes must include the VTAMTOPO keyword.

Note: If an NCP major node is reported, its physical lines cannot be suppressed, but its switched logical lines can be suppressed.

If you specify the VTAM start option MSGMOD, turn it off while running NetView command lists and NetView automation.

If you want information about...	Refer to...
VTAM requirements	FLBSYSD and FLBSYSDA initialization files (comments) or *Tivoli NetView for OS/390 SNA Topology Manager and APPN Accounting Manager Implementation Guide*
VTAM CMIP services	VTAM library
ISTCMIP, ACYGDMO, and ISTASN1 statements	VTAM library

Security Considerations

The SNA topology manager requires that a userid of APPNTM be defined to an SAF product, such as RACF, and authorized to use RODM. This requires the use of the RACF ADDUSER and PERMIT commands.

If you want information about...	Refer to...
RACF requirements	"Using RACF for RODM Security" in "Defining NetView Security for RODM" in the *Tivoli NetView for OS/390 Security Reference*

Automation Options

The manager applications start automatically under standard NetView installation procedures. The VTAM CMIP agent starts upon initialization of NetView. The default automation table, DSITBL01, also starts automation for the managers (FLBAUT).

If you want information about...	Refer to...
Automation options	*Tivoli NetView for OS/390 SNA Topology Manager and APPN Accounting Manager Implementation Guide*

Initialization File

The SNA topology manager is shipped with an initialization file (FLBSYSD) that enables you to tailor it to your installation. This file contains the names of your RODM, the VTAM application names for the managers, values for controlling automatic topology gathering, message suppression options, and other customization information. In addition, three other files (FLBOSIDS, FLBSRT, and FLBEXV) enable you to further customize SNA topology manager views.

The FLBSYSD initialization file is not refreshed until the topology manager is stopped and started again.

The FLBOSIDS, FLBSRT, and FLBEXV customization files can be refreshed using the TOPOSNA REFRESH command.

You can use symbols to represent your RODM name, VTAM APPLNAME, and VTAM APPLPASS, if symbolic substitution is enabled on your system. Ensure that the symbols are defined in member IEASYM*xx* of SYS1.PARMLIB.

If you want information about...	Refer to...
Customizing the initialization files	*Tivoli NetView for OS/390 SNA Topology Manager and APPN Accounting Manager Implementation Guide*

Starting the SNA Topology Manager

If you updated CNMSTYLE as described in "Chapter 1. Introduction" on page 1, the SNA topology manager should automatically start when NetView is started. To verify this:

- The VTAM CMIP services must be active.

  To see if the agent is active, type:

  ```
  D NET,VTAMOPTS
  ```

  and search for:

  ```
  VTAMOPTS,OSIMGMT=YES
  ```

  where *vtamproc* is the name of the VTAM start procedure.
- The FLBTOPO autotask must be active.

  To see if it is active, type:

  ```
  LIST FLBTOPO
  ```

  If it is not active, enter:

  ```
  AUTOTASK OPID=FLBTOPO
  ```

  Notes:

  1. This only starts SNATM, not APPNAM.
  2. You can also issue the STARTCNM SNATM, STARTCNM GRAPHICS, or the STARTCNM ALL command to start the SNA topology manager.

If you want information about...	Refer to...
Enabling the SNA topology manager and the APPN accounting manager	*Tivoli NetView for OS/390 SNA Topology Manager and APPN Accounting Manager Implementation Guide*
STARTCNM command	HELP STARTCNM (online help)

Verifying the SNA Topology Manager and the APPN Accounting Manager Installation

The procedures in this section help you verify that the SNA topology manager functions are installed correctly. Certain commands can be issued even if the agents are not started; these commands verify that the manager code is properly installed.

Topology Manager Verification

Once the topology manager is active, you can use the following commands to control the collection of SNA subarea and APPN topology information.

Command	Use
TOPOSNA ACTIVATE	Request activation of a link or port

Command	Use
TOPOSNA CRITICAL	Monitor critical LU or CDRSC
TOPOSNA INACT	Deactivate a link or port
TOPOSNA LISTPOOL	Display internal storage pool statistics
TOPOSNA LISTREQS	Display status of pending requests
TOPOSNA LISTRODM	Display RODM activity and object counts
TOPOSNA LISTSTOR	Display storage usage counts
TOPOSNA MONITOR	Start monitoring topology
TOPOSNA PURGE	Delete expired unreachable resources from RODM
TOPOSNA QUERYDEF	Display current default settings
TOPOSNA RECYCLE	Recycle a link or port
TOPOSNA REFRESH	Change NetView-provided defaults
TOPOSNA SETDEFS	Modify defaults
TOPOSNA STOP	Stop monitoring topology
TOPOSNA STOPMGR	Stop the topology manager
TOPOSNA TRACE	Control tracing

To begin verifying the topology manager host installation, issue the TOPOSNA QUERYDEF command. You will receive a response similar to that shown in Figure 11.

```
NCCF                    Tivoli NetView    NTVE9 NETOP1   2/1/01 08:17:32
* NTVE9    TOPOSNA QUERYDEF
- NTVE9    FLB494I SNA TOPOLOGY MANAGER DEFAULT SETTINGS FOLLOW:
- NTVE9    FLB495I  MONITOR SNA LOCAL TOPOLOGY FOR NEW NETWORK NODES : NO
- NTVE9    FLB496I  MONITOR SNA LOCAL TOPOLOGY FOR NEW END NODES     : NO
- NTVE9    FLB650I  MONITOR SNA NETWORK TOPOLOGY FOR NEW T5 NODES    : NO
- NTVE9    FLB651I  MONITOR SNA LOCAL TOPOLOGY FOR NEW T5 NODES      : NO
- NTVE9    FLB497I  SNA NETWORK TOPOLOGY IMMEDIATE RETRY INTERVAL    : 60
- NTVE9    FLB498I  SNA NETWORK TOPOLOGY IMMEDIATE RETRY LIMIT       : 5
- NTVE9    FLB499I  SNA NETWORK TOPOLOGY LONG-TERM RETRY INTERVAL    : 1800
- NTVE9    FLB500I  SNA NETWORK TOPOLOGY LONG-TERM RETRY LIMIT       : 48
- NTVE9    FLB501I  SNA LOCAL TOPOLOGY IMMEDIATE RETRY INTERVAL      : 60
- NTVE9    FLB502I  SNA LOCAL TOPOLOGY IMMEDIATE RETRY LIMIT         : 5
- NTVE9    FLB503I  SNA LOCAL TOPOLOGY LONG-TERM RETRY INTERVAL      : 1800
- NTVE9    FLB504I  SNA LOCAL TOPOLOGY LONG-TERM RETRY LIMIT         : 48
- NTVE9    FLB546I  SNA LU COLLECTION IMMEDIATE RETRY INTERVAL       : 60
- NTVE9    FLB547I  SNA LU COLLECTION IMMEDIATE RETRY LIMIT          : 5
- NTVE9    FLB548I  SNA LU COLLECTION LONG-TERM RETRY INTERVAL       : 1800
- NTVE9    FLB549I  SNA LU COLLECTION LONG-TERM RETRY LIMIT          : 48
- NTVE9    FLB411I TOPOSNA QUERYDEF COMMAND COMPLETED SUCCESSFULLY
```

Figure 11. TOPOSNA QUERYDEF Command Results

If an agent is running in the network, issue the TOPOSNA MONITOR command to start monitoring at the agent node. This command verifies that communication between the manager and agent is working properly. Issue the TOPOSNA STOP command to stop monitoring at the agent node.

For detailed user scenarios for the topology function, refer to “SNA Topology Manager User Scenarios” in the *Tivoli NetView for OS/390 SNA Topology Manager and APPN Accounting Manager Implementation Guide*.

Access the first SNA topology manager view (from the NetView management console) to verify that RODM has been loaded correctly.

From the NMC tree view, select **SuperclusterView**. This opens a view of the NN domain network cluster object and verifies that the topology data model files is loaded correctly.

At this point, no further navigation is possible because RODM does not yet contain SNA resource objects for monitoring (unless you issued the TOPOSNA MONITOR command to start monitoring). You have two options for creating resources in RODM:

- Load the APPN sample network.

  This option provides you and your operators an opportunity to gain experience with the topology manager in a test environment. The sample network is designed so you can simulate dynamic addition of resources in the network and gain familiarity with APPN views. Command actions against resources (such as activating or recycling) are not available because it is a sample network.

  For scenarios describing how to load and use this sample network, refer to "SNA Topology Manager User Scenarios" in the *Tivoli NetView for OS/390 SNA Topology Manager and APPN Accounting Manager Implementation Guide*.
- Start monitoring the SNA topology from agents in your network.

  This option loads RODM with the actual resources from your network. See the NetView online help for instructions on using the TOPOSNA MONITOR command to start monitoring agents in your network.

If you want information about...	Refer to...
TOPOSNA command	HELP TOPOSNA (online help)

Accounting Manager Verification

To control the collection of APPC session and conversation accounting information, use the ACCTSNA command to start the accounting management control object (AMCO). The agent cannot send notifications and the accounting manager cannot retrieve data until the ACCTSNA START request completes. Once the accounting manager is active, you can use the following commands to control the collection of SNA subarea and APPN topology information.

Command	Use
ACCTSNA DISPLAY	Display current status
ACCTSNA LISTNODE	Display a list of agents
ACCTSNA MODIFY	Change parameters
ACCTSNA QUERYDEF	Display current settings
ACCTSNA RETRIEVE	Transfer data from the agent
ACCTSNA SETDEFS	Modify defaults
ACCTSNA START	Start data collection
ACCTSNA STOP	Stop data collection
ACCTSNA STOPMGR	Stop the accounting manager
ACCTSNA TRACE	Control tracing

Verify the accounting manager installation by issuing the ACCTSNA QUERYDEF command at the NetView operator console. This verifies the accounting manager is installed correctly at the host. You will receive a response similar to that shown in Figure 12.

```
NCCF                    Tivoli NetView    NTVD0 NETOP1   2/1/01 17:03 :42
* NTVD0    ACCTSNA QUERYDEF
- NTVD0    FLB260I ACCOUNTING MANAGER DEFAULT SETTINGS FOLLOW:
- NTVD0    FLB261I  BUFFER FULL BEHAVIOR               : HALT
- NTVD0    FLB262I  NOTIFY RECORD COUNT                : 0
- NTVD0    FLB263I  NOTIFY BUFFER-FULL PERCENT         : 50
- NTVD0    FLB264I  PERIODIC DATA RETRIEVAL INTERVAL   : NONE
- NTVD0    FLB265I  PERIODIC DATA RETRIEVAL START TIME : 00:00
- NTVD0    FLB267I  ACKNOWLEDGEMENT COUNT              : 100
- NTVD0    FLB268I  IMMEDIATE RETRY INTERVAL           : 60
- NTVD0    FLB269I  IMMEDIATE RETRY LIMIT              : 5
- NTVD0    FLB270I  LONG-TERM RETRY INTERVAL           : 1800
- NTVD0    FLB271I  LONG-TERM RETRY LIMIT              : 48
- NTVD0    FLB272I  RETRIEVE FINAL DATA                : YES
```

Figure 12. ACCTSNA QUERYDEF Command Results

If an agent is running in the network, issue the ACCTSNA START command to start data collection at the agent node. This command verifies that communication between the manager and agent is working properly. Issue the ACCTSNA STOP command to stop the data collection.

If you want information about...	**Refer to...**
Detailed user scenarios for the accounting function	“Using the APPN Accounting Manager” in the *Tivoli NetView for OS/390 SNA Topology Manager and APPN Accounting Manager Implementation Guide*

Chapter 7. Installing and Configuring MultiSystem Manager Agents

Topology agents are supported for the following network environments:

- LAN Network Manager LANs
- Novell NetWare LAN Servers and Requesters
- Internet protocol (IP) networks managed by NetView for AIX, NetView for Windows NT, NetView for Solaris, HP OpenView on HP-UX, or HP OpenView on Solaris
- ATM networks
- NetFinity networks
- Tivoli management regions
- Other types of networks using Open topology agents

If you are using MultiSystem Manager to manage workstations in your network, you need to install the topology agent on a service point workstation. A service point workstation can be any workstation in your distributed network that is running the topology agent and associated applications required for managing the distributed network.

The default MultiSystem Manager Configuration file is shipped as FLCSAINP. The intent is for you to copy FLCSAINP to FLCAINP and make your enterprise specific customizations in FLCAINP.

Installing Topology Agents

Topology agents for MultiSystem Manager are available in a variety of formats:

CD-ROM This format allows installation using platform-specific methods. These include:

Windows/NT
InstallShield to install and uninstall

AIX Install format (SMIT) to install and uninstall

OS/2 Software Installer format to install and uninstall

HP-UX and Solaris
Tape archive (TAR) format to install

The README documentation is located in the README*language* subdirectory, where *language* specifies the language. For example, US-English is ENU.

Tivoli Web site
Contains the same data formats as the CD-ROM. This contains the latest version of the workstation-based code for downloading. The Web address is: http://www.tivoli.com/nv390_supported

The following table identifies README file names and installation methods associated with the workstation-based components.

Table 6.

Component Name	CD-ROM	Tivoli Web	README
MultiSystem Manager ATM agent	x	x	msmatm.me
MultiSystem Manager IP agent	x	x	msmip.me
MultiSystem Manager LAN Network Manager agent	x	x	See “Modifying the LAN Network Manager Environment”.
MultiSystem Manager NetFinity agent	x	x	msmnf.me - included with the NetFinity product
MultiSystem Manager NetWare agent	x	x	msmnw.me
MultiSystem Manager Open agent	x	x	Included with the vendor-supplied topology agent
MultiSystem Manager TMR agent	x	x	flctmr1.html

Install the topology agent on the service point workstation. In cases such as NetWare, it is also necessary to install an agent on all NetWare servers in the managed environment. Refer to the README files for specific installation instructions.

Open agents can be written by customers or provided by vendors.

Documentation and tools for building a MultiSystem Manager Open Topology Interface agent application can be downloaded from the following Web address: http://www.tivoli.com/nv390_tools

Select the downloads from the MSMTOOLK package.

If you want information about...	Refer to...
MultiSystem Manager agents	*Tivoli NetView for OS/390 MultiSystem Manager User’s Guide*

Modifying Your Open Environment

The modifications needed for a particular network of distributed resources depends on the requirements of the vendor-supplied topology agent managing that network. For information on topology agent requirements, refer to the documentation provided with your topology agent.

Modifying the LAN Network Manager Environment

The resource hierarchy list in the alerts and resolutions sent by the LAN Network Manager to NetView contains the OS/2 service point name as it is defined to the agent. This field must be defined correctly at the LAN Network Manager workstation to properly process topology and status alerts and resolutions that are sent by the agent.

If the LAN Network Manager is already set up as an OS/2 service point and is presently sending alerts to NetView, this step might already be complete. This step is performed at the agent workstation.

To set up the agent on the LAN Network Manager workstation:

1. Verify that the OS/2 service point name has been assigned at the agent work station.
   - Select Host parameters on the System Parameters Actions pull-down menu.
   - Select OS/2 as the Host connection if OS/2 is not already selected.
   - Specify the OS/2 service point name to be the same as the PU name (for a SSCP-PU session) or the LU name (for an LU 6.2 session).

     This is the name specified on the SP parameter in the GETTOPO command for this agent.
2. Restart the LAN Network Manager if you modify the host parameters.

Chapter 8. Preparing and Activating the MultiSystem Manager

This chapter includes the following steps, which are necessary to enable MultiSystem Manager:

- "Modifying Your MVS Environment"
- "Modifying the Event Automation Service" on page 62
- "Modifying Your NetView Start Procedure" on page 62
- "Security Considerations" on page 63
- "Automation Options" on page 65
- "Assigning Operator Profiles to Autotasks" on page 65
- "Initialization Files" on page 65
- "Setting Up Your NetView Cross-Domain Environment" on page 66
- "RUNCMD Considerations" on page 67
- "Modifying the NetView RATE and AUTORATE Statements" on page 68
- "Starting MultiSystem Manager" on page 69
- "Verifying the MultiSystem Manager Installation" on page 69

If you want information about...	Refer to...
Using MultiSystem Manager	*Tivoli NetView for OS/390 MultiSystem Manager User's Guide*

Modifying Your MVS Environment

All data sets in the NetView STEPLIB DD concatenation must be authorized for NetView initialization.

Use the Authorized Program Facility (APF) to authorize the REXX data set (SEAGLMD or SEAGALT). To be authorized, a library's name must appear in the list of authorized libraries for the APF member (IEAAPF*xx*) of SYS1.PARMLIB. Refer to the MVS library for more information on APF authorization.

If you previously ran MultiSystem Manager on this system, the REXX data set might already be authorized.

Changes to the APF member do not take effect until you restart MVS or dynamically update the APF member by using the MVS SETPROG command.

1. To authorize the REXX data set (SEAGLMD or SEAGALT), edit your APF member in SYS1.PARMLIB and add the REXX data set (SEAGLMD or SEAGALT).
2. Restart MVS, if necessary.

   If your system is set up to use dynamic APF services, you can avoid restarting MVS by using the SETPROG command to dynamically update the APF list.

Modifying the Event Automation Service

This section is applicable only if the following statements are true:

- You are using the MultiSystem Manager Tivoli management region feature or the MultiSystem Manager IP feature.
- You are using TCP/IP to communicate between NetView for OS/390 and the MultiSystem Manager agent.

To communicate using TCP/IP, the event automation service must be installed and a port number must be specifically assigned using the PortNumber and UsePortMapper keywords in member IHSAECFG.

To explicitly assign a port, specify a value (other than zero) to the PortNumber keyword and ensure that UsePortMapper is set to NO in member IHSAECFG.

If you want information about...	**Refer to...**
Event Automation Service	*Tivoli NetView for OS/390 Installation: Configuring Additional Components*

Modifying Your NetView Start Procedure

If you are using TCP/IP to communicate between NetView for OS/390 and MultiSystem Manager agent, refer to *Tivoli NetView for OS/390 Installation: Getting Started*.

The REXX programs for MultiSystem Manager have been compiled with the ALTERNATE option.

If you access the REXX runtime library from NetView, MultiSystem Manager REXX programs run in compiled mode. Otherwise, the REXX alternate library is used and MultiSystem Manager REXX programs run in interpreted mode.

If the REXX runtime library or REXX alternate library is not accessible from the pageable link pack area (PLPA), you need to modify the NetView start procedure to access one of these libraries.

Security Considerations

You can add to your network security by authorizing access to RODM. Defining the application ID to an SAF product, such as RACF, allows you to verify security levels for API calls into RODM. A security level of 3 enables you to connect, disconnect, query, create objects, delete objects, change fields, and so on. It does not enable you to create classes, create fields, delete classes, delete fields, and so on.

Add the RODM user application ID to RACF or other security system with a security level of 3. The RODM user application ID for MultiSystem Manager is your NetView *domain ID* concatenated with the letters MSM, for example, CNM01MSM.

If you want information about...	Refer to...
Using RACF for RODM security	"Using RACF for RODM Security" on page 13

Command Authorization

The command model statements in DSICMENT define the MultiSystem Manager command procedures and their associated scope-checking parameters to NetView.

Note: If you modify member DSICMD or any of the members included by DSICMD, restart NetView so the changes take effect.

To implement command authorization checking:

1. To restrict the use of MultiSystem Manager commands to specific operators, modify the command authorization as described in *Tivoli NetView for OS/390 Administration Reference*.

   The command model statements that specify MOD=DSICCP represent REXX command lists. NetView command authorization checking for REXX command lists does not apply to keyword and keyword values.
2. Ensure that the following commands can be issued by the same operators who can issue GETTOPO commands:
   EXCMD
   RUNCMD or RMTCMD, depending on your system
3. You can have command authorization performed on the keywords and values of the FLCARODM command.

   If you are using scope checking (CMDAUTH=SCOPE), modify the FLCARODM command model statement with the appropriate KEYCLASS and VALCLASS statements, and check the OPCLASS statement in the operator profiles to ensure that the intended users of these commands, keywords, and values are assigned to the correct scope class. Refer to sample DSICMRMT for KEYCLASS values for FLCARODM keywords.

   If you are using the CMDAUTH table or the System Authorization Facility (SAF), use the NetView command, SECMIGR, to migrate these definitions. Refer to the *Tivoli NetView for OS/390 Administration Reference* for more information.

If you want information about...	Refer to...
Scope of commands and other forms of command authorization	*Tivoli NetView for OS/390 Security Reference*

Implementing Span of Control

Span of control provides a means to control access to particular resources and views. The preferred method for defining the span of control is the NetView span

table. MultiSystem Manager fills in the UserSpanName field in RODM whenever a viewable object is created. The UserSpanName is composed, when available, of the following parts (in order):

- service point
- application
- element (if any)
- object specific data

Note: If UserSpanName and the NetView span table are used for span of control, the CommandSpanName field in RODM is ignored.

To modify the operator profiles, edit member FLCSPRFB, which is shipped in the CNMSAMP data set. Specify CTL=SPECIFIC or CTL=GENERAL on the AUTH statement.

The following example shows how operator profile FLCSPRFB can be modified for span of control:

```
FLCSPRFB     PROFILE
             AUTH     MSGRECVR=NO,CTL=GENERAL
             ISPAN    SPAN1,SPAN2
             OPCLASS  1,2
             END
```

If you use the CommandSpanName field in RODM, span of control support by MultiSystem Manager requires that you code BLDVIEWS control statements specifying the SPANNAME keyword for the resources that you want to control.

Specify the SPANNAME keyword on the BLDVIEWS control statements for the resources that you want to control. This sets the CommandSpanName field on MultiSystem Manager RODM objects that you specified on the BLDVIEWS control statements.

For example, the following BLDVIEWS control statement adds span of control for all bridge resources whose names begin with A03:

```
BRIDGE=A03*,TYPE=REAL,SPANNAME='SPAN1,SPAN2'
```

Note: If you are using NetFinity, the statement is slightly different. The following example adds span of control for all NetFinity-managed OS/2 systems whose name begins with CLIENT:

```
NTF_OS=CLIENT*,TYPE=NTF_OS2,SPANNAME='SPAN1,SPAN2'
```

If you want information about...	Refer to...
Operator profiles	*Tivoli NetView for OS/390 Security Reference*
Span of control	"Using Spans to Protect Resources and Views" in the *Tivoli NetView for OS/390 Security Reference*
Examples of UserSpanName	*Tivoli NetView for OS/390 Data Model Reference*
BLDVIEWS	*Tivoli NetView for OS/390 Resource Object Data Manager and GMFHS Programmer's Guide*

Automation Options

MultiSystem Manager provides automation statements for each of its features. These statements allow for automation of alerts and resolutions sent to NetView by the topology agent. MultiSystem Manager automation statements are included before any GMFHS automation statements and are controlled by the MSM tower and MSM subtower statements specified in CNMSTYLE.

The following automation statement is controlled by the GRAPHICS tower specification in CNMSTYLE:

```
IF (MSUSEG(0000) ¬= '' | MSUSEG(0002) ¬= '') & HIER ¬= ''
   THEN EXEC(CMD('DUIFECMV') ROUTE(ONE DUIFEAUT)) CONTINUE(Y);
```

During GMFHS initialization, the topology agent can send alerts to NetView before GMFHS has completed initialization. If these alerts arrive before GMFHS is initialized, they are not processed by GMFHS. As a result, the status of MultiSystem Manager objects is not automatically updated during this period. The automation statements provided in DSITBL01 generate a message when GMFHS is initialized.

Assigning Operator Profiles to Autotasks

Autotasks are driven by the NetView automation table when processing alerts and resolutions from the topology agents.

Autotasks must be created for each of the MultiSystem Manager agents to distribute the SNA and TCP/IP work load across different tasks in NetView. These autotasks can be specified on the AUTOTASK parameter on GETTOPO commands or on the GETTOPO statements in the initialization file.

If you modify the operator profile or member DSIOPF, restart NetView or issue the REFRESH OPERS command from the NetView command line to activate the changes.

1. Add autotasks to FLCSOPF as needed to distribute the workload across tasks.

   Note: FLCSPRFB in CNMSAMP is the sample operator profile used by the autotasks that are added for each MultiSystem Manager agent.
2. Change the passwords shipped in the sample file.

Initialization Files

There are two categories of MultiSystem Manager initialization file statements:

- General
- Agent-specific

General statements define various aspects of your system and network environment to MultiSystem Manager. They define system defaults, information about the RODM used, and information about your networking environment.

GETTOPO RES and ONLY statements are agent-specific. They are used to define service points to MultiSystem Manager during initialization.

The initialization file must contain the following statements:

- DEF_AUTOTASK
- RODMNAME or system symbolic

If you do not include other general statements, default values are used.

If you do not include GETTOPO statements in the initialization file, MultiSystem Manager is enabled, but topology and status information is not initially stored in RODM when you issue an INITTOPO command. Topology and status information is retrieved only when GETTOPO commands are later issued or when alerts are received.

If you include GETTOPO statements in the initialization file, MultiSystem Manager is enabled and topology and status information is retrieved from the specified service points and added to RODM during MultiSystem Manager initialization.

MultiSystem Manager provides a sample initialization file named FLCSAINP. You can modify and rename it to either the default initialization file name, FLCAINP, or to a unique name. You can also create your own initialization file.

MultiSystem Manager provides sample initialization files that contain examples of GETTOPO statements. These file are included by the sample file FLCSAINP. Table 7 shows the MultiSystem Manager agent and its sample initialization file.

Table 7. MultiSystem Manager Agent and Sample Initialization File

MultiSystem Manager Agent	Sample Initialization File
ATM	FLCSIATM
Internet Protocol	FLCSIIP
Internet Protocol TN3270	FLCS3270
LNM	FLCSILNM
NetFinity	FLCSINF
NetWare	FLCSINW
Open Topology Interface	FLCSIOPN
TMR	FLCSITME

Use the DISPTOPO command to display some of the general information that is defined by the initialization file. If you change the initialization file, issue the INITTOPO command to inform MultiSystem Manager of the changes.

If you want information about...	Refer to...
Initialization file statements	*Tivoli NetView for OS/390 MultiSystem Manager User's Guide*
GETTOPO command	NetView online help or *Tivoli NetView for OS/390 Command Reference*

Setting Up Your NetView Cross-Domain Environment

Note: This step is for MultiSystem Manager agents that use SNA to communicate with the topology agent.

Perform this step only if you have service points that are not in the same domain as the NetView running MultiSystem Manager. Your cross-domain environment affects:

- Alerts and resolutions
- RMTCMDs
- RUNCMDs

Setting up your cross-domain environment enables MultiSystem Manager to send RUNCMDs to, and receive alerts and resolutions from, service points in other SNA domains.

1. If alerts from a service point are sent to a NetView in a different domain, forward these alerts to the NetView running MultiSystem Manager.

   You can forward all alerts and resolutions from one NetView to another by issuing the following command from the NetView running MultiSystem Manager:

   ```
   FOCALPT CHANGE FPCAT=ALERT TARGET=remote_domain
   TARGNET=remote_netid
   ```

   Specify the same values for *remote_domain* and *remote_netid* as specified for the REMOTE keyword in the GETTOPO command. If *remote_netid* is not specified on the REMOTE keyword in the GETTOPO command, then the TARGNET keyword can be omitted.
2. To ensure RMTCMDs work between the NetView running MultiSystem Manager and remote NetViews, define dedicated autotasks at your remote NetViews specifically for this purpose.

   The RMTCMDs issued from MultiSystem Manager specify OPERID=*. Refer to *Tivoli NetView for OS/390 Command Reference* for more information about the RMTCMD command.
3. If the REMOTE parameter is used on the GETTOPO command for a service point, allocate additional NetView DSRBs at the remote NetView domain.
4. If the REMOTE parameter is used on the GETTOPO command for a service point (meaning a RMTCMD command is used to forward the RUNCMD command to another NetView domain), ensure that RMTCMDs work from the NetView domain running MultiSystem Manager to the destination NetView domain as specified in the REMOTE parameter of the GETTOPO command.
5. You might need to define the topology agent LU in a VTAM cross-domain resource (CDRSC) definition in the domain running MultiSystem Manager. If you are using cross-network sessions, the CDRSC definition might be needed in both networks.

RUNCMD Considerations

Note: This section is for MultiSystem Manager agents that use SNA to communicate with the topology agent.

MultiSystem Manager uses RUNCMDs to gather initial topology and status from topology agents that use SNA to communicate with NetView.

Setting the RUNCMD Time-Out Value

The RUNCMD time-out value specifies the amount of time to wait before a RUNCMD is cancelled. To prevent RUNCMDs from prematurely timing out, increase the time-out value.

The default value for RUNCMD time-out is stored in DSICTMOD. This value is assigned each time that NetView is initialized. If you modify member DSICTMOD, restart NetView for the changes to take effect.

If you previously installed a MultiSystem Manager topology agent on this system, the RUNCMD time-out value might already be set to an appropriate value for your site.

1. Edit DSICTMOD and set the RUNCMD time-out value.

2. Assemble and link-edit DSICTMOD using job CNMS0055.
3. Restart NetView for these changes to take effect.

   If you do not want to restart NetView at this time, specify the RUNCMD time-out value by using the COSTIME keyword of the DEFAULTS command.

If you want information about...	Refer to...
DEFAULTS command	*Tivoli NetView for OS/390 Command Reference*

Allocating Additional NetView DSRBs

Note: This step is for MultiSystem Manager agents that use SNA to communicate with NetView for OS/390.

DSRBs are used to store information about each RUNCMD request, and RUNCMD processing is managed by NetView task DSIGDS. If a RUNCMD request is made and there are no DSRBs available to task DSIGDS, the RUNCMD request is queued.

Allocate additional DSRBs to minimize the queuing of these requests.

If your connection between the MultiSystem Manager topology manager and its topology agents is over an SSCP-PU session, you might need to modify the NetView DSRB count for task DSIGDS:

1. Determine the DSRB allocation for task DSIGDS.
2. Edit the DSTINIT statement in NetView initialization member DSICPINT and modify the current DSRBO value with the value that is appropriate for your site.
3. Stop and restart task DSIGDS.

Notes:

1. Allocating additional DSRBs is not necessary if all of your sessions are LU 6.2 sessions.
2. If you have previously installed a MultiSystem Manager topology agent on this system, the DSRB count for task DSIGDS might already be modified.

If you want information about...	Refer to...
DSTINIT statements and DSICPINT	*Tivoli NetView for OS/390 Administration Reference*
Determining the DSRB allocation for task DSIGDS	*Tivoli NetView for OS/390 Tuning Guide*

Modifying the NetView RATE and AUTORATE Statements

It is possible for several to send numerous alerts to NetView that could overload the NetView hardware monitor database. Update the NetView RATE and AUTORATE statements with an appropriate value to prevent overloading the NetView hardware monitor database.

The RATE statement enables you to set the maximum rate at which alerts and resolutions can be logged into the NetView hardware monitor database. If an alert is blocked by a recording filter set by the RATE statement, it is not sent to automation unless you code an AUTORATE statement. The AUTORATE statement

enables you to send a blocked alert to automation.

If you want information about...	Refer to...
RATE and AUTORATE statements	*Tivoli NetView for OS/390 Administration Reference*

Tuning the REXX Environment

One of the most powerful features of the NetView program is the ability to run REXX code in the NetView environment. Several of the NetView components, as well as base NetView functions, exercise code that has been written in REXX.

MultiSystem Manager uses REXX command lists to acquire topology and status and to update objects in RODM. Tuning the number and size of REXX environments can imporove performance.

If you want information about...	Refer to...
REXX Environment	*Tivoli NetView for OS/390 Installation: Configuring Additional Components*

Starting MultiSystem Manager

To start MultiSystem Manager, issue the INITTOPO command. By default, this command uses an initialization file named FLCAINP. If your initialization file has a different name, specify this name immediately following the INITTOPO command. If your initialization member includes GETTOPO commands, they will be processed after the general statements in the MultiSystem Manager initialization file have been verified.

Verifying the MultiSystem Manager Installation

Once the MultiSystem Manager is active, you can use the following commands to control the collection of topology information:

Command	Use
GETTOPO ATM*xxxx*	Get topology and initial status for ATM resources
GETTOPO IP*xxxx*	Get topology and initial status for IP resources
GETTOPO LNM*xxxx*	Get topology and initial status for LNM resources
GETTOPO NF*xxxx*	Get topology and initial status for NetFinity resources
GETTOPO NWC*xxxx*	Get topology and initial status for NetWare resources
GETTOPO OPENRES	Get topology and initial status for OPEN resources
GETTOPO TME*xxxx*	Get topology and initial status for Tivoli Management Region resources

If you want information about...	Refer to...
Using MultiSystem Manager	*Tivoli NetView for OS/390 MultiSystem Manager User's Guide*
GETTOPO command	NetView online help

Chapter 9. Preparing and Activating the NetView Resource Manager

This chapter addresses the following technical considerations:

Defining Autotask AUTONRM

AUTONRM is the autotask used by the NetView Resource Manager (NRM) for processing. This is defined, as follows, in DSIOPF:

```
AUTONRM      OPERATOR    PASSWORD=AUTONRM
             PROFILEN    DSIPROFC
```

If you modify member DSIOPF, either restart NetView or issue the REFRESH OPERS command from the NetView command line to activate the changes.

Note: The autotask name for the NRM function *must* be AUTONRM.

Security Considerations

You can add to your network security by authorizing access to RODM.

Defining the RODM user application ID to an SAF product, such as RACF, allows you to verify security levels for API calls into RODM. For example, a security level of 3 enables you to connect, disconnect, query, create objects, delete objects, change fields, and so on. It *prevents* you from activities which include (but are not limited to): creating classes and fields, deleting classes and fields.

Add the RODM user application ID (for NRM) to RACF or other security system with a security level of 3. The RODM user application ID for NRM is your NetView *domain ID* concatenated with the letters NRM (for example, CNM01NRM).

Note: Authorizing access to RODM is necessary only for manager hosts.

If you want information about...	Refer to...
Using RACF for RODM security	"Using RACF for RODM Security" on page 13

Command Authorization

The command model statements in DSICMD (%INCLUDE member DSICMRMT) define the NetView Resource Manager command procedures.

```
CNME8600   CMDMDL   MOD=DSICCP,TYPE=R,RES=Y,ECHO=N
CNME8601   CMDMDL   MOD=DSICCP,TYPE=R,RES=Y
           CMDSYN   INITNRM
CNME8602   CMDMDL   MOD=DSICCP,TYPE=R,RES=Y
           CMDSYN   SUSPNRM
NRMCTL     CMDMDL   MOD=DSINRCTL,TYPE=R
```

Note: If you modify the member DSICMD or any of the members included by DSICMD, restart NetView so the changes take effect.

To implement command authorization checking:

1. Restrict the use of NetView Resource Manager commands to specific operators by modifying the command authorizations of NetView Resource Manager commands as described in ***Tivoli NetView for OS/390 Security Reference***.

   The command model statements that specify MOD=DSICCP represent REXX command lists. NetView command authorization checking for REXX command lists does not apply to keyword and keyword values.

   Note: CNME8600 runs on autotask AUTONRM, and should not be executed by other operators.
2. Ensure that the following commands can be issued by the same operators who can issue INITNRM and SUSPNRM commands:
   - EXCMD
   - RMTCMD
   - SOCKET
3. Ensure that the following commands can be issued by AUTONRM:
   - EXCMD
   - RMTCMD
   - SOCKET
4. You can have command authorization performed on the keywords and values of the NRMCTL command.

   If you are using scope checking (CMDAUTH=SCOPE), modify the NRMCTL command model statement with the appropriate KEYCLASS and VALCLASS statements, and check the OPCLASS statement in the operator profiles to ensure that the intended users of these commands, keywords, and values are assigned to the correct scope class.

   If you are using the CMDAUTH table or a System Authorization Facility (SAF) product, such as RACF, use the NetView command, SECMIGR, to migrate these definitions. Refer to the ***Tivoli NetView for OS/390 Security Reference*** for more information.
5. NetView Resource Manager uses FLCARODM. You can have command authorization performed on the keywords and values of the FLCARODM command.

   If you are using scope checking (CMDAUTH=SCOPE), modify the FLCARODM command model statement with the appropriate KEYCLASS and VALCLASS statements, and check the OPCLASS statement in the operator profiles to ensure that the intended users of these commands, keywords, and values are assigned to the correct scope class.

   If you are using the CMDAUTH table or an SAF product, such as RACF, use the NetView command, SECMIGR, to migrate these definitions. Refer to the ***Tivoli NetView for OS/390 Security Reference*** for more information.

If you want information about...	**Refer to...**
Scope of commands and other forms of command authorization	*Tivoli NetView for OS/390 Security Reference*

Initialization Considerations

You can start the NRM at NetView initialization by specifying INIT.NRM = Yes in CNMSTYLE.

You can also start the NRM at a later time by issuing the INITNRM command.

Note: Before starting the NRM, review the initialization parameters in CNMSTYLE, and customize them as necessary.

You can override all parameters except the DISPLAY STATUS values on the INITNRM command.

You can also use the RESTYLE NRM command to stop the NRM and restart it (if INIT.NRM is Yes) after customizing parameters for it in CNMSTYLE.

Sampling

NRM detects RODM failures when an NRM request fails. NRM continues trying to connect to RODM until either a connection is established or the NRM is shut down.

When the connection is reestablished, it requests status from all the agents NRM was monitoring at the time of the failure.

You may want to specify a value for sampling in order for the NRM to detect RODM failures, especially if the status of your NRM objects does not change very often.

If you want sampling to occur, either modify NRM.SAMPLERATE in CNMSTYLE or specify the SAMPLE keyword on the INITNRM command.

Display Status Defaults

NRM uses the following DisplayStatus values for its real objects:

Active	Satisfactory
Inactive	Unknown
Unknown	Unknown
Threshold 1 met or exceeded	Intermediate
Threshold 2 met or exceeded	Medium Unsatisfactory
Threshold 3 met or exceeded	Unsatisfactory
RESET	Unknown

Monitoring Remote NetView Systems

To monitor a remote NetView system, perform the following steps:

1. Code a HOSTDEST parameter at the remote system.

   Note: For the following reasons, do *not* code multiple HOSTDESTs for the same destination host (For example, an SNA HOSTDEST for CNM01, and an IP HOSTDEST for CNM01).

   - Coding multiple HOSTDESTs for the same target host causes unpredictable behavior.
   - NRM does not support backup host destinations.

2. If you code HOSTDEST statements, you must also code the CMODE and PORT parameters, if applicable.

   Examples are provided in CNMSTYLE.

3. Activate other NetView functions that are required to monitor remote NetViews:

- The manager that is receiving remote host data uses the following functions:
  - RMTCMD
  - Hardware Monitor
  - LU 6.2 transport
  - TCP/IP Alert Receiver (if your communication method is TCP/IP)
- The agent sending its data to a manager uses the following functions:
  - LU 6.2 transport (if your communication method is SNA)
  - RMTCMD

Note: The agent can be a NetView host with NRM.TYPE=MGR. It is the *agent function* that uses the above listed items.

RMTCMD Considerations

NRM uses RMTCMD to communicate between NetView hosts.

Ensure that RMTCMD capability exists for each mode of communication (SNA or IP) to be used for transporting data between manager(s) and agent(s). For example, if an agent is reporting to a manager using TCP/IP, TCP/IP-based remote operations must be enabled at the agent and manager hosts.

Starting the NetView Resource Manager

To start the NRM, specify INIT.NRM = Yes in CNMSTYLE or issue the INITNRM command.

If you issue the INITNRM command with no parameters, the parameter values in CNMSTYLE are used to initialize the Resource Manager.

Verifying the NetView Resource Manager Installation

After the NRM is active, you can issue NRMCTL LISTPARM to verify your initialization parameters.

You can verify domains that are being monitored from a manager host with the NRMCTL LISTMON command.

Access the NRM view (from the NetView management console) to verify that RODM has been populated correctly. From the NMC tree view, select **NetViewTasks**.

This opens a view of the NRM domain aggregate objects and verifies that NRM data has been stored in RODM. You can right-click on this object and select a command(s) to execute.

You can then navigate from the NRM domain aggregate object to the NRM task aggregate objects.

From the NRM task aggregate objects, you can navigate to the NRM real statistical objects.

Commands are available at both of these levels.

If you want information about...	Refer to...
INITNRM command	NetView online help

If you want information about...	Refer to...
NRMCTL command	NetView online help
RESTYLE command	NetView online help

Chapter 10. Managing Views

As SNA topology manager and MultiSystem Manager agents gather network topology information, they store that information in RODM.

By default, this information is presented by the NetView management console (NMC) in a top-down, hierarchical perspective. You can change these default views in NMC to customize the graphical representation of your network to accomplish the following goals:

- Better meet the needs of your organization
- Customize the graphical representation of your network

The list of views including access to individual objects within those views can be granted on a per-operator basis.

Types of Views

An NMC operator can view the following:

- Network views
- Exception views

Network views are created by gathering topology and status from topology agents (or from an NMC user) by using tools such as the RODM collection manager or BLDVIEWS. Network views are capable of displaying objects as they relate to other objects, by showing relationships or links between them.

The current state of each device within the view is represented by a color scheme. For example, by default the colors used are: Green for satisfactory status; red for unsatisfactory status.

Exception views are special types of views which show objects that are in an *exception state* and that might require operator attention.

For example, as a network device becomes inoperable, its state might change from satisfactory to unsatisfactory. If an exception view is defined to display this type of device, an object representing that device appears within this exception view, alerting an operator about the unsatisfactory state. When the device becomes operable again, it disappears from the exception view.

Because the RODM collection manager is capable of creating views that actively track changes in RODM, you can use the RODM collection manager to create a network view with characteristics that are similar to an exception view. For example, you can create a RODM collection that specifies all IP Routers that are not in satisfactory state; this becomes a network view that resembles the functionality of an exception view.

Exception views and RODM collection manager views with similar characteristics have a few differences that might make one a more appropriate choice than the other in a given situation.

- Advantages of exception views:
  1. Exception views use less CPU resource on the host system.

     They are only created and maintained when an NMC user has an exception view open.
  2. Exception views can be opened at NMC by default in Details mode while network views are still opened by default in Topology mode.
- Advantages of RODM collection manager views with similar characteristics:
  1. RODM collection manager views are easier to define and maintain than exception views.
  2. The status that constitutes an exception can easily be defined on an object-by-object basis.
  3. RODM collection manager views provide more flexible options.

     For example, more criteria (than just status) can be applied.
  4. When two objects in a RODM collection manager view are connected, that connection is displayed.

These two view types appear as expandable branches of the view tree in NMC:

Figure 13. Collapsed View Tree

As the operator clicks on the plus sign, the tree is expanded to show the available views:

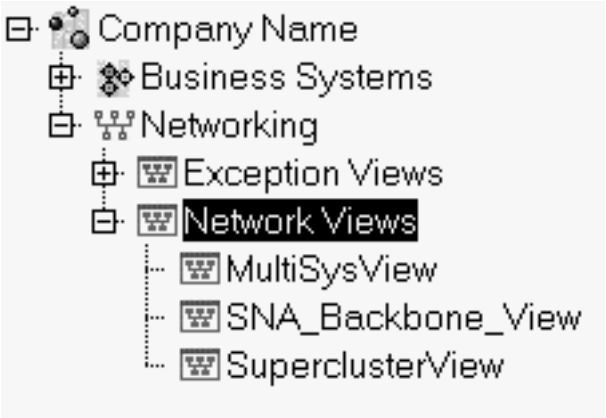


Figure 14. Expanded View Tree

The operator can then double-click on a selection to open the view.

Restricting Access to Views

If necessary, operator access can be restricted to:

- Views
- Individual resources within views

This is done using span of control.

To implement span of control use the:

- NGMFVSPN attribute to specify if operators are to be subject to span checking
- NetView span table to define views and resources within views to spans
- CTL attribute to specify that span checking is to be done for an operator

If you want information about...	Refer to...
Using spans to protect resources and views	*Tivoli NetView for OS/390 Resource Object Data Manager and GMFHS Programmer's Guide* and *Tivoli NetView for OS/390 Security Reference*

Applying Policy to Views

Using NMCSTATUS policy definitions, you can define time schedules for resources in NMC views. With these schedules, policy can be applied to views to specify when:

- The status for one or more resources in a view is disabled at the NMC console
- One or more resources in a view is suspended from aggregation

If you want information about...	Refer to...
Creating and loading a policy file containing NMCSTATUS policy definitions	*Tivoli NetView for OS/390 Administration Reference*

Correlating Views

NetView uses topology correlation to automatically tie together resources managed by different types of topology agents, such as IP and NetWare. Topology correlation is provided for all MultiSystem Manager topology agents, the SNA topology manager, and customer or vendor applications that use the GMFHS data model.

Most managed resources participate in more than one network or system. For example, a workstation may serve as a station in a token ring LAN network, a NetWare LAN requester, and a Windows 95 platform.

Most management agents recognize only one type of network or system, as opposed to every resource running in the system. For example, the NetWare agent recognizes only NetWare LAN servers and requesters. Therefore, a NetWare view of the example workstation contains a NetWare server and requester, but not the token ring adapter and Windows 95 platform.

Topology correlation overcomes the agent-specific perspective of management agents and lets an NMC operator view all components of a network device. This single view includes the current status of all components.

Object correlation is enabled by loading input file FLCSDM8 into RODM. To load FLCSDM8, uncomment the following line in job CNMSJH12:

```
//* DD DSN=NETVIEW.V1R4M0.CNMSAMP(FLCSDM8),DISP=SHR  <-CORRELATE SAMPL
```

Correlation occurs when an application sets a valid value in a field of a RODM object that is enabled for correlation. Objects are enabled for correlation by loading file FLCSDM8.

MultiSystem Manager and SNA topology manager automatically set the value of these fields, which results in correlation.

Views are displayed on a NetView management console.

If you want information about...	Refer to...
Correlating views	*Tivoli NetView for OS/390 RODM Programmer's Guide*

Enabling MultiSystem Manager Object Correlation

To optimize navigation and storage for resources managed by the NetFinity and TMR agents, issue the GETTOPO commands in the following order:

1. GETTOPO NFRES
2. GETTOPO TMERES
3. Any other GETTOPO commands

Enabling SNA Topology Manager Object Correlation

SNA correlation occurs on PU resources for which SNA topology manager discovers a LAN MAC address. PU resources are excluded from TOPOSNA commands that do not include the `LOCAL` parameter. Use the `LOCAL` parameter on any TOPOSNA command issued to resources you want included in correlation. Correlation also occurs for LU resources which support a telnet 3270 (tn3270) server or requester.

Communications Server/390 Release 10 is required for correlation of LU resources and any PU resources other than PU 2.1 OS/2 workstations.

Enabling Object Correlation for Additional SNA Topology Manager and GMFHS Resources

If you have SNA topology manager or GMFHS resources that do not correlate to any LAN or IP resources, this indicates that the management agent did not discover a LAN MAC address or IP address that could be used for correlation.

If you know that some of these resources do participate in LAN or IP networks and if you have the MAC addresses and IP addresses for those resources, you can include them in correlation. This is done by setting those address values in the managed objects in RODM.

To set these values, you can write a NetView CLIST that uses MultiSystem Manager access through RODMView or Visual BLDVIEWS.

If you want information about...	Refer to...
Correlation	*Tivoli NetView for OS/390 Resource Object Data Manager and GMFHS Programmer's Guide*

Managing Views

This section describes the tools you can use to manage RODM views. These tools include:

- "RODM Collection Manager"
- "BLDVIEWS"
- "Visual BLDVIEWS" on page 83
- "DELVIEWS" on page 83
- "RODMView" on page 83
- "RODM Unloader" on page 84

RODM Collection Manager

The RODM collection manager actively manages the contents of views based on criteria set by the user. These criteria can be a naming convention, a set of statuses, or both. A view managed by the RODM collection manager can have characteristics of both network and exception views. NMC administrators have access to the RODM collection manager on their desktops. You can also manage views with complex criteria beyond names and statuses.

If you want information about...	Refer to...
RODM collection manager	NMC online help or to the *Tivoli NetView for OS/390 Resource Object Data Manager and GMFHS Programmer's Guide*.

BLDVIEWS

BLDVIEWS can gather a set of objects based on a naming convention. You can use BLDVIEWS to:

- Create either a network or an exception view out of that set of objects.
- Manipulate a field on that set of objects.

BLDVIEWS is controlled by an input file.

Network views created by BLDVIEWS are snapshots in time. As network topology changes occur, it might be necessary to recreate the views by rerunning BLDVIEWS. For this reason, it is advisable to migrate BLDVIEWS-based views to RODM collection manager views. The FLCV2RCM command can be used to do this.

For more information about FLCV2RCM, see NetView online help.

BLDVIEWS can also be used to:

- Create aggregate objects
- Link and unlink objects to and from views and aggregates

Views or aggregates can be grouped by:

- Networks
- Regions
- Locations
- Sites
- Plants
- Buildings
- Floors
- Individual operator responsibilities

You can set or change fields such as DisplayResourceOtherData.

Using the GMFHS data model, you can create objects in these classes:

- Various view object classes
- GMFHS_Aggregate_Objects_Class
- GMFHS_Managed_Real_Objects_Class

You cannot use BLDVIEWS to delete objects, classes, or fields, but you can modify certain fields of objects.

For the SNA topology manager and MultiSystem Manager data models, objects cannot be created or deleted using BLDVIEWS. However, views can be created to include objects of these classes. BLDVIEWS supports real and aggregate objects created in RODM by the MultiSystem Manager and SNA topology manager.

Views and aggregates created by BLDVIEWS are static. Only the objects that are in RODM at the time you run BLDVIEWS are processed. If new objects are added to RODM, you must rerun BLDVIEWS. BLDVIEWS changes to the RODM data cache are not part of any RODM loader file and are lost if RODM is cold-started.

Views that are created with BLDVIEWS might need to be recreated because they can become out-of-date as system and network resources change.

You can use BLDVIEWS to enable generic command support (for example activate and deactivate) from the NMC on MultiSystem Manager-discovered network resource objects. BLDVIEWS provides a way to map a default set of commands to these generic commands for key MultiSystem Manager resources.

If you want information about...	Refer to...
BLDVIEWS	*Tivoli NetView for OS/390 Resource Object Data Manager and GMFHS Programmer's Guide*

Visual BLDVIEWS

Visual BLDVIEWS (VBV) is a graphical front end to BLDVIEWS; it simplifies the use of BLDVIEWS by eliminating the need to know the syntax of the BLDVIEWS input file language.

Visual BLDVIEWS provides a way to gather a set of objects (based on a naming convention) and to modify a field for each of these objects. This is useful for making batch updates to many objects. With Visual BLDVIEWS, you can query and display objects as they appear in RODM by double-clicking on them. Immediate updates can be made to individual objects.

VBV consists of host and workstation applications communicating through TCP/IP.

To enable the VBV server on the NetView host, perform the following steps:

1. Uncomment the AUTOTASK.AUTOVBV statement in CNMSTYLE.
2. Ensure that the TOWER MSM statement is uncommented.

If you want information about...	Refer to...
VBV and BLDVIEWS	*Tivoli NetView for OS/390 Resource Object Data Manager and GMFHS Programmer's Guide* and "BLDVIEWS" on page 81

DELVIEWS

You can use DELVIEWS to delete the following customized views from RODM:

- Network views
- Configuration peer views
- Configuration backbone views
- Exception views

For example, to delete a specific exception view (NET_EX_VIEW), type:

```
DELVIEWS NETA_EX_VIEW TYPE=EXCP
```

You can also use an asterisk (*) as a wildcard at the end of a view name. For example, to delete all network views with names starting with LAN, enter:

```
DELVIEWS LAN*
```

An optional parameter is RODM=*rodmname*. This is not necessary if MultiSystem Manager is initialized and connected to RODM.

RODMView

RODMView is a menu-driven tool that enables you to display and manipulate RODM objects from the NetView command facility. It requires more knowledge of GMFHS and other data models than the graphical Visual BLDVIEWS. It does not require a separate workstation installation.

If you want information about...	Refer to...
RODMView	Online command help for RODMView

RODM Unloader

As data is dynamically added to RODM, you might want to periodically use the RODM Unloader to create text files that represent the current contents of RODM. This is a slightly different archival method than checkpointing. A checkpoint data set is a binary representation of the current contents of RODM, which can be used to restart RODM. The RODM Unloader utility takes the contents of RODM and creates RODM Loader language statements that can be reloaded later.

If you want information about...	Refer to...
RODM unloader utility	*Tivoli NetView for OS/390 Resource Object Data Manager and GMFHS Programmer's Guide*

Appendix A. SNA Definitions for the NMC Topology Servers

To establish LU 6.2 connectivity between your host and the NMC topology servers, define the workstation-to-host connections in your VTAM and NCP definitions. You can define your workstation connections to the host using any type of connection that allows parallel LU 6.2 conversion.

Ethernet LAN-Attached NMC Servers

To define NMC topology servers to an ethernet LAN, add definitions for NCP and VTAM. In the NCP generation, where the ethernet is attached, define a GROUP set that defines the physical ethernet adapter.

If you want information about...	Refer to...
The details of the parameters in the following example	*NCP, SSP, and EP Resource Definition Guide* and the *NCP, SSP, and EP Resource Definition Reference*

```
A03ETH   GROUP ETHERNET=PHYSICAL,                                  X
               DIAL=NO,                                            X
               ISTATUS=ACTIVE,                                     X
               LNCTL=SDLC,                                         X
               VIRTUAL=NO
A03E170  LINE  ADDRESS=(1070,FULL),                                X
               ANS=CONT,                                           X
               FRAMECNT=(100000,5000),                             X
               INTFACE=ETH2,                                       X
               LANTYPE=DYNAMIC
A03P170  PU    ANS=CONT,                                           X
               ARPTAB=(2000,20),                                   X
               INNPORT=YES,                                        X
               PUTYPE=1
         IPLOCAL LADDR=96.0.0.99,                                  X
               INTFACE=ETH2,                                       X
               METRIC=1,                                           X
               PROTOCOL=RIP
```

Because LAN-attached workstations appear as switched nodes to VTAM, they must be defined in a switched major node. In the following example, the LOCADDR=00 LU is the independent LU that the NMC topology server uses. The four other LUs are the dependent LUs that are used as the 3270 emulation screens.

Note: The LU names that you specify here must also be specified when you define the NMC topology server profile configurations.

If you want information about...	Refer to...
The following example	The VTAM library

```
A03LETH  VBUILD TYPE=SWNET,MAXGRP=25,MAXNO=25
A03L021  PU ADDR=01,IDBLK=05D,IDNUM=00001,MAXPATH=0,IRETRY=NO,       X
               MAXDATA=256,PUTYPE=2,DISCNT=NO,ISTATUS=ACTIVE,       X
               MAXOUT=7,PASSLIM=7
A03L0210 LU LOCADDR=00,DLOGMOD=DSIL6MOD,MODETAB=AMODETAB
A03L0212 LU LOCADDR=02,DLOGMOD=M2SDLCQ,MODETAB=AMODETAB
A03L0213 LU LOCADDR=03,DLOGMOD=M2SDLCQ,MODETAB=AMODETAB
A03L0214 LU LOCADDR=04,DLOGMOD=M2SDLCQ,MODETAB=AMODETAB
A03L0215 LU LOCADDR=05,DLOGMOD=M2SDLCQ,MODETAB=AMODETAB
```

Because the server workstation uses LU 6.2 parallel session support, you must update the ADDSESS, MAXSESS, and AUXADDR keywords specified on the build macro to support this configuration.

To update these keywords, perform the following tasks:

1. Verify that you specified the correct keywords and values on the LUDRPOOL macro.
2. If you already modified these keywords, verify that you defined them correctly for the configuration you are defining.
3. Perform the changes to these keywords on every NCP that has a server workstation.

.

Token-Ring LAN-Attached NMC Servers

To define NMC topology servers to a token-ring local area network (LAN), add definitions for NCP and VTAM. In the NCP generation, where the token ring is attached, define two GROUP sets. The first GROUP set defines the physical NTRI connection. The second GROUP defines the logical switched lines that are mapped onto the ring. The AUTOGEN parameter defines the number of lines created. There must be at least one line for each workstation.

If you want information about...	Refer to...
The details of the parameters in the following example	*NCP, SSP, and EP Resource Definition Guide* and the *NCP, SSP, and EP Resource Definition Reference*

```
N3PGRP1  GROUP ECLTYPE=(PHYSICAL,PER),                          X
               USSTAB=AUSSTAB,                                  X
               ANS=CONTINUE,                                    X
               ISTATUS=ACTIVE,                                  X
               XID=NO
N3N1088  LINE  ADDRESS=(1088,FULL),                             X
               PORTADD=1,                                       X
               LOCADD=400000311088
N3P8821  PU    MAXOUT=7
N3881D0  LU    ISTATUS=INACTIVE
*
*
*
N3BNNG1  GROUP ECLTYPE=LOGICAL,                                 X
               AUTOGEN=32,                                      X
               CALL=INOUT,                                      X
               MODETAB=AMODETAB,                                X
               USSTAB=AUSSTAB,                                  X
               ANS=CONTINUE,                                    X
               PHYPORT=1,                                       X
               ISTATUS=ACTIVE,                                  X
               XMITDLY=NONE
```

Because LAN-attached workstations appear as switched nodes to VTAM, they must be defined in a switched major node. In the following example, the LOCADDR=00 LU is the independent LU that the NMC topology server uses. The four other LUs are the dependent LUs that are used as the 3270 emulation screens.

Note: The LU names that you specify here must also be specified when you define the NMC topology server profile configurations.

If you want information about...	Refer to...
The following example	The VTAM library

```
N3LNTRI  VBUILD TYPE=SWNET,MAXGRP=25,MAXNO=25
N3L021   PU ADDR=01,IDBLK=05D,IDNUM=00001,MAXPATH=0,IRETRY=NO,        X
               MAXDATA=256,PUTYPE=2,DISCNT=NO,ISTATUS=ACTIVE,         X
               MAXOUT=7,PASSLIM=7
N3L0210  LU LOCADDR=00,DLOGMOD=DSIL6MOD,MODETAB=AMODETAB
N3L0212  LU LOCADDR=02,DLOGMOD=M2SDLCQ,MODETAB=AMODETAB
N3L0213  LU LOCADDR=03,DLOGMOD=M2SDLCQ,MODETAB=AMODETAB
N3L0214  LU LOCADDR=04,DLOGMOD=M2SDLCQ,MODETAB=AMODETAB
N3L0215  LU LOCADDR=05,DLOGMOD=M2SDLCQ,MODETAB=AMODETAB
```

Because the server workstation uses LU 6.2 parallel session support, you must update the ADDSESS, MAXSESS, and AUXADDR keywords specified on the build macro to support this configuration. Verify that you have specified the correct keywords and values on the LUDRPOOL macro. If you already modified these keywords, ensure that you defined them correctly for the configuration you are defining. Perform the changes to these keywords on every NCP that has a server workstation.

SDLC-Attached NMC Servers

When defining workstations with an SDLC connection, update only the NCP generation. In the following example, the LOCADDR=00 LU is the independent LU that the NMC topology server uses. The four other LUs are the dependent LUs that are used as the 3270 emulation screens.

Note: The LU names that you specify here must also be specified when you define the NMC topology server profile configurations.

```
N3PS2GRP GROUP DUPLEX=FULL,                                          X
               CLOCKNG=EXT,                                          X
               SPEED=9600,                                           X
               NRZI=NO,                                              X
               RETRIES=(7,4,4),                                      X
               ETRATIO=1,                                            X
               LNCTL=SDLC,                                           X
               ISTATUS=ACTIVE,                                       X
               IRETRY=YES,                                           X
               PASSLIM=12,                                           X
               ANS=CONT,                                             X
               XID=YES,                                              X
               PUDR=YES,                                             X
               MAXDATA=256,                                          X
               MAXOUT=7,                                             X
               USSTAB=AUSSTAB
**
N3L02    LINE  ADDRESS=000
**
N3L021   PU    ADDR=C1,
               PUTYPE=2
**
N3L0210  LU    LOCADDR=0,MODETAB=AMODETAB,DLOGMOD=DSIL6MOD
N3L0212  LU    LOCADDR=2,MODETAB=AMODETAB,DLOGMOD=M2SDLCQ
N3L0213  LU    LOCADDR=3,MODETAB=AMODETAB,DLOGMOD=M2SDLCQ
N3L0214  LU    LOCADDR=4,MODETAB=AMODETAB,DLOGMOD=M2SDLCQ
N3L0215  LU    LOCADDR=5,MODETAB=AMODETAB,DLOGMOD=M2SDLCQ
```

Because the server workstation uses LU 6.2 parallel session support, you must update the ADDSESS, MAXSESS, and AUXADDR keywords specified on the build macro to support this configuration. If you have already modified these keywords,

ensure that you have defined them correctly for the configuration you are defining. Perform the changes to these keywords on every NCP that has an NMC topology server.

Defining the Logical Unit (LU) Name

The topology server can communicate with the NetView environment through an LU 6.2 conversation. To communicate with the NetView environment through an LU 6.2 conversation, define the LU name for the topology server that will be used.

To define the LU name of the topology server on the topology server workstation, use the sample `ts.acg` configuration file in one of the following directories:

- Intel: `%BINDIR%\TDS\server\sample`
- UNIX: `$BINDIR/TDS/server/sample`

This file is shipped with NMC and represents a sample communications server configuration file for the topology server LU 6.2 communications. Use the configuration file as a guide to tailor the configuration to fit your environment. See the appropriate communications server documentation for more information about LU 6.2 connections.

Configuring Your Workstation for SNA

Before using NMC to monitor your network, define your workstation network configuration. This section outlines how to configure SNA on your topology server workstation.

Configuring Communications Server

To configure SNA LU 6.2 support for your workstation network, use the information in this section.

NCP should be channel-attached to the host, with token-ring connections between the NCP and server.

The topology communications server provides communications between the server workstation and the host. In a multi-workstation configuration, the topology server also handles communications between the server workstation and the client workstations.

The topology server uses a topology communications server to establish and communicate through LU 6.2 sessions. The topology communication server supports LU 6.2 sessions through any medium that supports LU 6.2 communication. The server workstation is directly connected to a status focal point host through an LU 6.2 session.

This type of workstation serves the client workstation by storing views and status and by distributing this data to the client workstations. The client workstation acts as a graphic operator console that retrieves data from the server workstation. An example of this is a topology server workstation communicating with topology console workstations over a LAN.

SNA requires an IBM SNA communications product to be properly configured. Although there are numerous products that support the NMC requirements for communications over SNA, they share certain parameters that must be configured. Before you configure your workstation for SNA, define the following:

Data Link Control (DLC)
Defines the underlying communications hardware to use. This includes token-ring and SDLC modems.

Local Node Definition
Defines the workstation control point name and ID to the hardware.

Connections
Specifies the link between the workstation and the host.

Local LU
Specifies the logical name that the topology communications server uses for communications.

Transaction Program
Specifies the attributes for the topology communications server.

Modes
Specifies the attributes of the communication session between the topology communications server and the host.

Note: Workstations can have additional local LU profiles and partner LU profiles that define LUs that are used by other applications. These profiles must not specify the same LU names that are used by the topology server.

To define the LU name of the topology server on the topology server workstation on Windows NT, use the sample `ts.acg` configuration file in one of the following directories:

- For Intel: `%BINDIR%\TDS\server\sample`
- For UNIX: `$BINDIR/TDS/server/sample`

This file is shipped with NMC and represents a sample communications server configuration file for the topology server LU 6.2 communications. Use the configuration file as a guide to tailor the configuration to fit your environment. See the appropriate communications server documentation for more information about LU 6.2 connections.

Using Communications Server to Configure your Workstation Network

The tables in this section provide examples of how to define the LU 6.2 sessions between the host and the server workstation.

The fields provided in the tables are fields that, typically, are in the configuration panel for each area. Because each SNA communications program is slightly different, the fields in the tables might not be available in all products or they might be labeled slightly differently.

Data Link Control Profile for SDLC: Table 8 shows an example of the topology server workstation configuration for the data link control profile for SDLC.

Table 8. Data Link Control Profile for SDLC

Communications Server Field Name	Example Field Value	Explanation
Adapter	0	The SLDC adapter used for this communication.
Line Type	Non-switched	This will be non-switched unless you are using a modem that must be dialed.

Table 8. Data Link Control Profile for SDLC (continued)

Communications Server Field Name	Example Field Value	Explanation
Link Station Role	Secondary	A link station role of secondary is required when connecting to a host through a boundary function.
Maximum I-Field Size	2048	The maximum RU size must be coordinated with the MAXDATA parameter in the NCP definition for the non-switched line.
Free Unused Links	No	This box should not be selected for non-switched communications.
Local Station Address	C1	The SDLC station address of this node. It must match the ADDR parameter in the NCP definition for this node.
Line Mode	Full-duplex	This parameter must be coordinated with the DUPLEX parameter in the NCP definition for the non-switched line.
Modem Speed	Full speed	Set the modem rate at full speed unless line noise is a problem. Half speed provides more reliable operation if line noise is a problem.
Use NRZI Data Encoding	No	This parameter must match the NRZI parameter in the NCP definition for the non-switched line. If you select NRZI, the communication hardware between the workstation and the host must support NRZI.
Send Window Count	7	The send window count must match the MAXOUT parameter in the NCP definition for the non-switched line.
Receive Window Count	7	The receive window count must match the MAXOUT parameter in the NCP definition for the non-switched line.

Data Link Control Profile for LAN: Table 9 shows an example of the topology server workstation configuration for the data link control profile for LAN.

Table 9. APPC APIs over Token Ring

Communications Server Field Name	Example Field Value	Explanation
Adapter	0	The LAN adapter used for this communication.
Free Unused Link	No	`No` indicates that the host connection should not be dropped when the last application is logged off.
Local Node Name	A19SRVCP	The 8–character name of the local control point. The name must match the CP name parameter in the host definition for this workstation.

Table 9. APPC APIs over Token Ring (continued)

Communications Server Field Name	Example Field Value	Explanation
Maximum I-field Size	2048	Specifies the maximum I-field size that the station can receive. This number must be greater than, or equal to, the maximum I-field size in the transmission service mode profile. For the host connection only, this number must coordinate with the MAXDATA value on the PU definition statement.
Maximum Number of Link Stations	16	One link is used to allocate sessions to another workstation. This parameter sets the maximum number of link stations that LU 6.2 communication uses.
Network ID	NETA	The 8–character name for the network containing this PU.
Receive Window Count	1	The receive window count specifies the number of frames that the station can receive before sending an acknowledgment.
Send Window Count	7	The send window count specifies the number of frames that the station can send before receiving an acknowledgment. For the host connection, this number must match the MAXOUT parameter on the PU definition statement.

Local Node Characteristics for SDLC or LAN: Table 10 shows an example of a topology server workstation configuration for the local node characteristics. Modify this profile for either an SDLC or LAN configuration.

Table 10. Local Node Characteristics for SDLC or LAN (Communications Server)

Communications Server Field Name	Example Field Value	Explanation
Network ID	NETA	The 8–character name of the network containing this PU.
Local Node Name (CP name)	A19SRVCP	The 8–character name of the local control point. The name must match the CP name parameter in the host definition for this workstation.
Local Node Type	Network node	The topology server can work with any node type. This example, however, assumes that the node type of the server is network node.
Local Node ID (in hexadecimal)	(05D) 00001	Used to determine which link station is primary for negotiable link stations. The node with the higher value becomes the primary link station. For a LAN connection, the first 3 hexadecimal digits are typically '05D'; the last 5 hexadecimal digits must match the IDNUM parameter in the switched major node PU definition.

Local LU Profile: Table 11 shows an example of the topology server workstation configuration for the logical unit profile you have specified.

Table 11. Local LU Profile (Communications Server)

Communications Server Field Name	Example Field Value	Explanation
LU Name	A19SRVLO	Specifies the name of the LU as it is known externally to the SNA network. The value must match the host definition for this workstation. This value is unique for each workstation.
LU Alias	EGVPC	The name of the local LU profile. Create a local LU profile with the name EGVPC. The topology server requires the value EGVPC. The topology server also requires the LU alias to be specified in uppercase.
NAU Address	Independent LU	The topology server requires that an independent LU be selected.

Modes Profile: Table 12 is an example of the topology server workstation configuration for the transmission service mode profile.

Table 12. Modes Profile (Communications Server)

Communications Server Field Name	Example Field Value	Explanation
Mode Name	DSIL6MOD	Specifies the mode name profile. Create a mode name profile with the name DSIL6MOD. The topology server requires this value.
Class of Service	#CONNECT	Specifies the class of service name. Use the system default for this value.
Mode Session Limit	3	Specifies the mode session limit. The topology server requires a minimum of 3. The sum of the mode session limits for a partner LU must be equal to or less than the partner LU session limit in the partner LU profile.
Minimum Contention Winners	1	The topology server requires 1 contention winner source session.
Receive Pacing Window	1	Should be coordinated with the PACING parameter in the NCP definition for this LU and the SSNDPAC and ARCVPAC parameter in the VTAM logmode entry table for DSIL6MOD. Pacing prevents a session from flooding a node with data and using all the available buffers.
RU Size	Default RU	Enables Communications Server to select the most appropriate maximum RU size. If Maximum RU size is selected, the maximum RU size must be smaller than or equal to the maximum RU size in the appropriate DLC profile.

Transaction Program Definitions: Table 13 shows an example of the topology server workstation configuration for the transaction program profile.

Table 13. Transaction Program Definitions

Communications Server Field Name	Example Field Value	Explanation
Service TP	No	The topology server transaction program is not a service TP.
Transaction Program (TP) Name	30F0F4F4	The required name of the topology server transaction program. This value is case-sensitive and the topology server requires the TP name to be specified in upper case.
Transaction Program (TP) File Name (for OS/2)	ihsctp.exe	The TP executable file which enables the host and server to communicate.
Conversation Security	No	The topology server does not support conversation security.
Program Parameter String		The topology server does not require this field.

Connections for SDLC or LAN: If you have an SDLC or LAN connection, configure a connection to the host.

Note: Define the connection to the host if you do not have a 3270 emulator session to that host.

Then configure a connection by performing the following steps:

1. Select the adapter type.
2. Verify that the appropriate adapter number is in the entry field.
3. Click **Continue**.

Table 14. Connections for SDLC or LAN

Communications Server Field Name	Example Field Value	Explanation
Link name	LINK0001	Specifies the user default.
Adjacent Node ID	Blank	Specifies the XID used by the partner node. Leave this value blank for the topology server.
Partner Network ID	NETID	Use the NETID in the VTAM ATCSTR*xx* member.
Partner Node Name	SSCPNAME	Use SSCPNAME in the VTAMATCSTR*xx* member. This field and the Partner Network ID may be left blank.
LAN Destination Address (LAN only)		The 12–character, hexadecimal LAN address of the computer to which you are defining the connection.
Address Format	Token-ring	This value should match the type of LAN connection you are using: token-ring or ethernet.

Table 14. Connections for SDLC or LAN (continued)

Communications Server Field Name	Example Field Value	Explanation
Remote SAP	04	Defines the address of the service access point (SAP) of the host with which you are communicating.

Appendix B. NetView Graphics Samples

This appendix contains the following:

- "GMFHS Samples"
- "MultiSystem Manager Samples" on page 97
- "NetView Management Console (NMC) Samples" on page 108
- "NetView Resource Manager Samples" on page 108
- "RODM Samples" on page 109
- "SNA Topology Manager and APPN Accounting Manager Samples" on page 117

Other NetView samples are described in *Tivoli NetView for OS/390 Installation: Getting Started*.

Note: Although the samples have been tested and are supported by Tivoli, you are responsible for making the proper modifications to the samples for your operating environment.

GMFHS Samples

Table 15 lists the samples and includes a brief description of each GMFHS sample and the data set name where each member resides once installation is complete.

Table 15. List of GMFHS Samples

Distributed As	Name	Description	Data Set Name
CNMS4402	CNMSNIFF	GMFHS automation example	CNMSAMP
CNMS4403	EKGSNIFF	GMFHS automation example	CNMSAMP
CNMSJH10	CNMGMFHS	GMFHS start procedure	CNMSAMP
CNMSJH13	same	Relink - edits the assembler table, DUIFSMT	CNMSAMP
CNMSTYLE	same	This sample defines many of the NetView initialization parameters. Refer to CNMSTYLE for more information. If you make changes to this sample while the NetView program is running, recycle the NetView program to merge these changes.	DSIPARM

Table 15. List of GMFHS Samples (continued)

Distributed As	Name	Description	Data Set Name
DSICMENT	same	NetView command model statements including those for GMFHS.	DSIPARM
DUIFCUX2	same	Sample RODM method (for exception views) that is driven as an installation exit by GMFHS if the USRXMETH keyword is set in the DisplayStatus mapping table For more information on DUIFCUX2, DUIFCUXM, USRXMETH, and the DisplayStatus mapping table, refer to "How GMFHS Uses RODM" in the *Tivoli NetView for OS/390 Resource Object Data Manager and GMFHS Programmer's Guide*	CNMSAMP
DUIFCUXM	same	Sample RODM method (for exception views) that is driven as an installation exit by GMFHS if the USRXMETH keyword is set in the DisplayStatus mapping table For more information on DUIFCUXM, refer to "How GMFHS Uses RODM" in the *Tivoli NetView for OS/390 Resource Object Data Manager and GMFHS Programmer's Guide* for information on DUIFCUXM, USRXMETH, and the DisplayStatus mapping table.	CNMSAMP
DUIFDEXV	same	Sample exception views object and exception views candidate definitions	CNMSAMP
DUIFEDEF	same	Non-SNA resource alert processor	CNMSAMP
DUIFEMDY	same	NETCENTER migration sample	CNMSAMP
DUIFEUSR	same	Provides alert translation constants for GMFHS event management	CNMSAMP
DUIFSNET	same	Sample load file	CNMSAMP
DUIFSTRC	same	GMFHS data model	CNMSAMP
DUIGINIT	same	GMFHS initialization parameters	DSIPARM
DUIGPWLU	same	Provides graphic data server LU names to GMFHS that are not to be allowed to acquire a session	DSIPARM
DUIIGHB	same	Initialization values for the DUIDGHB task	DSIPARM

MultiSystem Manager Samples

Table 16 lists the samples and includes a brief description of each MultiSystem Manager sample and the data set name where each member resides once installation is complete.

Table 16. List of MultiSystem Manager Samples

Distributed As	Name	Description	Data Set Name
CNMSTYLE	same	This sample defines some of the NetView initialization parameters. Refer to CNMSTYLE for more information. If you make changes to this sample while the NetView program is running, recycle the NetView program to merge these changes.	DSIPARM
DSICMENT	same	NetView command model statements including those for MultiSystem Manager. Modify this file if you want to change the scope checking values for the MSM commands.	DSIPARM
FLCSAINP	same	Sample initialization file. This file can be used as a template when creating the MultiSystem Manager initialization file (or files) for your site. Modify this file and rename it FLCAINP if you plan to use the default initialization file name when issuing the INITTOPO command. If you rename the file with a name other than FLCAINP, specify that file name when issuing the INITTOPO command. Each of the MultiSystem Manager initialization statements is described briefly in the file. FLCSAINP also contains an example of how to use the %INCLUDE statement to include other MultiSystem Manager initialization files.	CNMSAMP

Table 16. List of MultiSystem Manager Samples (continued)

Distributed As	Name	Description	Data Set Name
FLCSCCHK	same	Authorization checking for MultiSystem Manager commands sent using the TCP/IP transport. REXX command list FLCACCHK is called when processing a MultiSystem Manager command that uses TCP/IP to communicate between NetView for OS/390 and the MultiSystem Manager agent. As shipped, FLCACCHK does not do any authorization checking and all commands are processed. Sample FLCSCCHK contains the REXX source code from FLCACCHK. FLCSCCHK resides in the CNMSAMP dataset and is provided in sample format to allow you to add authorization checking for MultiSystem Manager commands sent using the TCP/IP transport. Refer to the sample's prolog for instructions on replacing the existing REXX command list with your modified sample.	CNMSAMP
FLCSDM1	same	MultiSystem Manager data model — part 1. This file contains the class and field definitions of the MultiSystem Manager data model that are SystemView®-compliant. This file contains the SystemView classes used to create view objects.	CNMSAMP
FLCSDM2	same	MultiSystem Manager data model — part 2. This file contains the class and field definitions of the MultiSystem Manager data model that are private extensions to the data model. This file contains the private classes used to create view objects.	CNMSAMP
FLCSDM3	same	MultiSystem Manager data model — part 3. This file sets default status aggregation on both real and aggregate objects. This file also sets some default values for fields previously created. This file can be rerun with changed defaults.	CNMSAMP
FLCSDM4	same	MultiSystem Manager data model — part 4. This file creates objects and sets fields used to graphically display objects.	CNMSAMP
FLCSDM5	same	MultiSystem Manager data model — part 5. This file contains the default values for view change notification. This file can be customized.	CNMSAMP

Table 16. List of MultiSystem Manager Samples (continued)

Distributed As	Name	Description	Data Set Name
FLCSDM6	same	MultiSystem Manager data model — part 6. This file creates the default exception view definitions. The Exception view classes that are created by this sample are the same as those referenced in sample FLCSEXV.	CNMSAMP
FLCSDM8	same	MultiSystem Manager data model — part 8. This file enables the use of RODM methods for dynamic topology correlation. The file loads the methods FLCMCON and FLCMCOR into RODM.	CNMSAMP
FLCSDM9	same	MultiSystem Manager data model — part 9. This file enables the use of RODM methods for linking TN3270 resources to IP resources.	CNMSAMP
FLCSEALH	same	Sample REXX alert processor. REXX command list FLCAEALH is called from the NetView automation table to process alerts generated by the IBM NetWare topology agent. Sample FLCSEALH contains the REXX source code from FLCAEALH. FLCSEALH resides in the CNMSAMP data set and is provided in the sample format to enable you to modify the automated alert processing. Refer to the sample's prolog for instructions on replacing the existing REXX command list with your modified sample.	CNMSAMP
FLCSEXV	same	Exception view file. This file contains sample statements for MultiSystem Manager Exception view processing. It contains exception view statements for the real RODM classes that are supported by the MultiSystem Manager data model. Remove the comments from those statements you want to use. The prolog of this file describes the usage and syntax of the Exception view statements.	CNMSAMP
FLCSOPF	same	Operator definitions for MultiSystem Manager autotasks. This file contains the operator definition statements for the MultiSystem Manager autotasks. You can add additional autotasks for processing topology requests.	CNMSAMP

Table 16. List of MultiSystem Manager Samples (continued)

Distributed As	Name	Description	Data Set Name
FLCSPRFB	same	Operator profile for MultiSystem Manager autotasks. This file contains a sample profile for MultiSystem Manager autotasks. This file can be modified to satisfy your site requirements for scope checking.	CNMSAMP
FLCSSMT	same	MultiSystem Manager Status Mapping Table required for Exception View processing. This file contains sample statements for MultiSystem Manager exception view processing. The statements in this sample MUST be included in your existing NetView Status Mapping Table (DUIFSMT) if you want to implement exception view processing for MultiSystem Manager resources in RODM. The prolog of this file describes the usage and syntax of the NetView Status Mapping Table statements.	CNMSAMP
FLCVBLDS	same	Sample BLDVIEWS control cards.	CNMSAMP
FLCVCARD	same	This is the Visual BLDVIEWS language support file. It is generated by machine and should not be edited.	DSIPARM

ATM Topology Samples

The following samples are specific to the MultiSystem Manager ATM topology feature.

Table 17. List of MultiSystem Manager ATM Topology Samples

Distributed As	Name	Description	Data Set Name
FLCSAALH	same	Sample REXX alert processor. REXX command list FLCSAALH is called from the NetView automation table to process alerts generated by ATM. Sample FLCSAALH contains the REXX source code from FLCAAALH. FLCSAALH enables you to modify the automated alert processing. Refer to the sample's prolog for instructions on replacing the existing REXX command list with your modified sample.	CNMSAMP

Table 17. List of MultiSystem Manager ATM Topology Samples (continued)

Distributed As	Name	Description	Data Set Name
FLCSDM6A	same	Sample file for ATM Exception_View_Classes. This data model sample loads sample instances of Exception_View_Class for the ATM topology agent. The exception view classes that are created by this sample are the same as those referenced in sample FLCSEXV. Both FLCSDM6A and FLCSEXV can be customized.	CNMSAMP
FLCSIATM	same	Sample GETTOPO initialization statement for ATM. Sample FLCSIATM is included by MultiSystem Manager initialization file sample FLCSAINP. FLCSIATM contains a sample GETTOPO statement for ATM. Each of the keywords on the GETTOPO command are briefly described.	CNMSAMP

IP Topology Samples

The following samples are specific to the MultiSystem Manager IP topology feature.

Table 18. List of MultiSystem Manager IP Topology Samples

Distributed As	Name	Description	Data Set Name
FLCS3270	same	Sample for support of TN3270 Manager (server/client). Sample FLCS3270 enables management of TN3270 resources, both servers and clients.	CNMSAMP
FLCSDM6I	same	Sample file for IP Exception_View_Classes. This data model sample loads sample instances of Exception_View_Class for the IP topology agent. The exception view classes that are created by this sample are the same as those referenced in sample FLCSEXV. Both FLCSDM6I and FLCSEXV can be customized.	CNMSAMP
FLCSIAUT	same	Sample REXX alert processor. REXX command list FLCAIAUT is called from the NetView automation table to process alerts generated by Tivoli NetView. Sample FLCSIAUT contains the REXX source code from FLCAIAUT. FLCSIAUT resides in the CNMSAMP data set and is provided in sample format to enable you to modify the automated alert processing. Refer to the sample's prolog for instructions on replacing the existing REXX command list with your modified sample.	CNMSAMP

Table 18. List of MultiSystem Manager IP Topology Samples (continued)

Distributed As	Name	Description	Data Set Name
FLCSIIP	same	Sample GETTOPO initialization statement for IP. Sample FLCSIIP is included by MultiSystem Manager initialization file sample FLCSAINP. FLCSIIP contains a sample GETTOPO statement for IP. Each of the keywords on the GETTOPO command are described briefly.	CNMSAMP

LAN Network Manager Topology Samples

The following samples are specific to the MultiSystem Manager LAN Network Manager topology feature.

Table 19. List of MultiSystem Manager LNM Topology Samples

Distributed As	Name	Description	Data Set Name
FLCSDM6L	same	Sample file for LNM Exception_View_Classes. This data model sample loads sample instances of Exception_View_Class for the LNM topology agent. The exception view classes that are created by this sample are the same as those referenced in sample FLCSEXV. Both FLCSDM6L and FLCSEXV can be customized.	CNMSAMP
FLCSILNM	same	Sample GETTOPO initialization statement for LNM. Sample FLCSILNM is included by MultiSystem Manager initialization file sample FLCSAINP. FLCSILNM contains a sample GETTOPO statement for LNM. Each of the keywords on the GETTOPO command are described briefly.	CNMSAMP
FLCSLALH	same	Sample REXX alert processor. REXX command list FLCALALH is called from the NetView automation table to process alerts generated by LNM. Sample FLCSLALH contains the REXX source code from FLCALALH. FLCSLALH is provided in sample format to allow you to modify the automated alert processing. Refer to the sample's prolog for instructions on replacing the existing REXX command list with your modified sample.	CNMSAMP

Table 19. List of MultiSystem Manager LNM Topology Samples (continued)

Distributed As	Name	Description	Data Set Name
FLCSLAUT	same	Sample REXX alert and resolution processor. REXX command list FLCALAUT is called from the NetView automation table to process alerts and resolutions generated by LNM. Sample FLCSLAUT contains the REXX source code from FLCALAUT. FLCSLAUT is provided in sample format to allow you to modify the automated alert processing. Refer to the sample's prolog for instructions on replacing the existing REXX command list with your modified sample.	CNMSAMP

NetFinity Topology Samples

The following samples are specific to the MultiSystem Manager NetFinity topology feature.

Table 20. List of MultiSystem Manager NetFinity Topology Samples

Distributed As	Name	Description	Data Set Name
FLCSDM6H	same	Sample file for NetFinity Exception_View_Classes. This data model sample loads sample instances of Exception_View_Class for the NetFinity topology agent. The exception view classes that are created by this sample are the same as those referenced in sample FLCSEXV. Both FLCSDM6H and FLCSEXV can be customized.	CNMSAMP
FLCSHALH	same	Sample REXX alert processor. REXX command list FLCAHALH is called from the NetView automation table to process alerts generated by NetFinity. Sample FLCSHALH contains the REXX source code from FLCAHALH. FLCSHALH is provided in sample format to allow you to modify the automated alert processing. Refer to the sample's prolog for instructions on replacing the existing REXX command list with your modified sample.	CNMSAMP

Table 20. List of MultiSystem Manager NetFinity Topology Samples (continued)

Distributed As	Name	Description	Data Set Name
FLCSHAT2	same	Sample REXX alert processor. REXX command list FLCSHAT2 is called from the NetView automation table to process alerts generated by NetFinity. Sample FLCSHAT2 contains the REXX source code from FLCSHAT2. FLCSHAT2 is provided in sample format to allow you to modify the automated alert processing. Refer to the sample's prolog for instructions on replacing the existing REXX command list with your modified sample.	CNMSAMP
FLCSHAUT	same	Sample REXX alert processor. REXX command list FLCAHAUT is called from the NetView automation table to process alerts generated by NetFinity. Sample FLCSHAUT contains the REXX source code from FLCAHAUT. FLCSHAUT is provided in sample format to allow you to modify the automated alert processing. Refer to the sample's prolog for instructions on replacing the existing REXX command list with your modified sample.	CNMSAMP
FLCSINF	same	Sample GETTOPO initialization statement for NetFinity. Sample FLCSINF is included by MultiSystem Manager initialization file sample FLCSAINP. FLCSINF contains a sample GETTOPO statement for NetFinity. Each of the keywords on the GETTOPO command are described briefly.	CNMSAMP

NetWare Topology Samples

The following samples are specific to the MultiSystem Manager NetWare topology feature.

Table 21. List of MultiSystem Manager NetWare Topology Samples

Distributed As	Name	Description	Data Set Name
FLCSDM6N	same	Sample file for NetWare Exception_View_Classes. This data model sample loads sample instances of Exception_View_Class for the NetWare topology agent. The exception view classes that are created by this sample are the same as those referenced in sample FLCSEXV. Both FLCSDM6N and FLCSEXV can be customized.	CNMSAMP

Table 21. List of MultiSystem Manager NetWare Topology Samples (continued)

Distributed As	Name	Description	Data Set Name
FLCSEALT	same	Sample REXX alert and resolution processor. REXX command list FLCAEALT is called from the NetView automation table to process alerts and resolutions generated by the IBM NetWare status agent for system monitors. Sample FLCSEALT contains the REXX source code from FLCAEALT. FLCSEALT is provided in sample format to allow you to modify the automated alert processing. Refer to the sample's prolog for instructions on replacing the existing REXX command list with your modified sample.	CNMSAMP
FLCSEAUT	same	Sample REXX alert and resolution processor. REXX command list FLCAEAUT is called from the NetView automation table to process alerts and resolutions generated by the IBM NetWare topology agent. Sample FLCSEAUT contains the REXX source code from FLCAEAUT. FLCSEAUT is provided in sample format to enable you to modify the automated alert processing. Refer to the sample's prolog for instructions on replacing the existing REXX command list with your modified sample.	CNMSAMP
FLCSINW	same	Sample GETTOPO initialization statement for NetWare. Sample FLCSINW is included by MultiSystem Manager initialization file sample FLCSAINP. FLCSINW contains a sample GETTOPO statement for NetWare. Each of the keywords on the GETTOPO command are described briefly.	CNMSAMP

Open Topology Samples

The following samples are specific to the MultiSystem Manager Open topology feature.

Table 22. List of MultiSystem Manager Open Topology Samples

Distributed As	Name	Description	Data Set Name
FLCSDM6O	same	Sample file for Open topology Exception_View_Classes. This data model sample loads sample instances of Exception_View_Class for the Open topology agent. The exception view classes that are created by this sample are the same as those referenced in sample FLCSEXV. Both FLCSDM6O and FLCSEXV can be customized.	CNMSAMP
FLCSIOPN	same	Sample GETTOPO initialization statement for Open. Sample FLCSIOPN is included by MultiSystem Manager initialization file sample FLCSAINP. FLCSIOPN contains a sample GETTOPO statement for the Open topology interface. Each of the keywords on the GETTOPO command are described briefly.	CNMSAMP
FLCSOALH	same	Sample REXX alert processor. REXX command list FLCSOALH is called from the NetView automation table to process alerts generated by Open topology agents. Sample FLCSOALH contains the REXX source code from FLCAOALH. FLCSOALH is provided in sample format to allow you to modify the automated alert processing. Refer to the sample's prolog for instructions on replacing the existing REXX command list with your modified sample.	CNMSAMP
FLCSOAUT	same	Sample REXX alert processor. REXX command list FLCSOAUT is called from the NetView automation table to process alerts generated by Open topology agents. Sample FLCSOAUT contains the REXX source code from FLCAOAUT. FLCSOAUT is provided in sample format to allow you to modify the automated alert processing. Refer to the sample's prolog for instructions on replacing the existing REXX command list with your modified sample.	CNMSAMP
FLCSOX01	same	Sample REXX creation of RODM views, using MultiSystem Manager Access (FLCARODM command) and the open data model. This file also demonstrates topology correlation between different resources.	CNMSAMP

Tivoli Management Region Topology Samples

The following samples are specific to the MultiSystem Manager TMR topology feature.

Table 23. List of MultiSystem Manager TMR Topology Samples

Distributed As	Name	Description	Data Set Name
FLCSDM6T	same	Sample file for TMR Exception_View_Classes. This data model sample loads sample instances of Exception_View_Class for the TMR topology agent. The exception view classes that are created by this sample are the same as those referenced in sample FLCSEXV. Both FLCSDM6T and FLCSEXV can be customized.	CNMSAMP
FLCSITME	same	Sample GETTOPO initialization statement for TMR. Sample FLCSITME is included by MultiSystem Manager initialization file sample FLCSAINP. FLCSITME contains a sample GETTOPO statement for TMR. Each of the keywords on the GETTOPO command are briefly described.	CNMSAMP
FLCSTALH	same	Sample REXX alert processor. REXX command list FLCATALH is called from the NetView automation table to process alerts generated by TMR. Sample FLCSTALH contains the REXX source code from FLCATALH. FLCSTALH is provided in sample format to allow you to modify the automated alert processing. Refer to the sample's prolog for instructions on replacing the existing REXX command list with your modified sample.	CNMSAMP
FLCSTAUT	same	Sample REXX alert processor. REXX command list FLCATAUT is called from the NetView automation table to process alerts generated by TMR. Sample FLCSTAUT contains the REXX source code from FLCATAUT. FLCSTAUT enables you to modify the automated alert processing. Refer to the sample's prolog for instructions on replacing the existing REXX command list with your modified sample.	CNMSAMP

NetView Management Console (NMC) Samples

Table 24 lists the samples and includes a brief description of each NMC sample and the data set name where each member resides after installation is complete.

Table 24. List of NMC Samples

Distributed As	Name	Description	Data Set Name
CNMSTYLE	same	This sample defines many of the NetView initialization parameters. Refer to CNMSTYLE for more information. If you make changes to this sample while the NetView program is running, recycle the NetView program to merge these changes.	DSIPARM
DUIFPMEM	same	Defines CNMTAMEL task definition statements	DSIPARM
DUIFSMT	same	Default DisplayStatus mapping table	CNMSAMP
DUIISFP	same	Defines initialization values for the CNMTAMEL task	DSIPARM
DUIPOLCY	same	Define NMCSTATUS policy definitions	DSIPARM

NetView Resource Manager Samples

Table 25 lists the samples and includes a brief description of each NetView Resource Manager (NRM) sample and the data set name where each member resides after installation.

Table 25. List of NetView Resource Manager Samples

Distributed As	Name	Description	Data Set Name
CNMSTYLE	same	This sample defines many of the NetView initialization parameters. All of the NRM initialization parameters are in CNMSTYLE. Refer to CNMSTYLE for more information. If you make changes to this sample while the NetView program is running, recycle the NetView program to merge these changes. You can RESTYLE NRM to pick up NRM specific CNMSTYLE changes.	DSIPARM
DSICMRMT	same	NetView command model statements including those for NetView Resource Manager. Modify this file if you want to change the scope checking values for the NRM commands.	DSIPARM

Table 25. List of NetView Resource Manager Samples (continued)

Distributed As	Name	Description	Data Set Name
DUIFNRM1	same	This file contains a group of RODM Collection Manager collections. The contents of the collections are various NRM objects from ALL monitored NetViews.	CNMSAMP
DUIFNRM2	same	This file contains a group of RODM Collection Manager collections. The contents of the collections are various NRM objects from a single NetView.	CNMSAMP

RODM Samples

Table 26 lists the samples found in the NETVIEW.V1R4M0.SEKGSMP1 library. The table includes a brief description of each sample and the data set name into which each member is copied. Samples with a data set name of SEKGSMP1 are not copied.

All required JCL and definitions can be found in the NETVIEW.V1R4M0.SEKGSMP1 library under the distributed sample name.

Table 26. List of RODM Samples

Distributed As	Name	Description	Data Set Name
CNMS4290	ARODMCON	Activates the assembler DSINOR that connects to RODM	CNMSAMP
CNMSJ004	same	Allocates logs and VSAM databases for base NetView and RODM	CNMSAMP
CNMSJH12	same	Loads the RODM data cache for GMFHS	CNMSAMP
CNMSJI18	same	IEBCOPY RODM procedure to PROCLIB	CNMSAMP
CNMSTYLE	same	This sample defines some of the NetView initialization parameters. Refer to CNMSTYLE for more information. If you make changes to this sample while the NetView program is running, recycle the NetView program to merge these changes.	DSIPARM
DSICMENT	same	NetView command model statements for RODM	DSIPARM
DSIQTSKI	same	Defines RODM tasks to the RODM access and control subtask (DSIQTSK)	DSIPARM
EKG51101	same	PL/I Sample - Function EKG_Connect	SEKGSMP1
EKG51102	same	PL/I Sample - Function EKG_Disconnect	SEKGSMP1
EKG51201	same	PL/I Sample - Function EKG_Checkpoint	SEKGSMP1

Table 26. List of RODM Samples (continued)

Distributed As	Name	Description	Data Set Name
EKG51202	same	PL/I Sample - Function EKG_Stop	SEKGSMP1
EKG51302	same	PL/I Sample - Function EKG_CreateClass	SEKGSMP1
EKG51303	same	PL/I Sample - Function EKG_DeleteClass	SEKGSMP1
EKG51304	same	PL/I Sample - Function EKG_CreateField	SEKGSMP1
EKG51305	same	PL/I Sample - Function EKG_DeleteField	SEKGSMP1
EKG51306	same	PL/I Sample - Function EKG_CreateSubfield	SEKGSMP1
EKG51307	same	PL/I Sample - Function EKG_DeleteSubfield	SEKGSMP1
EKG51401	same	PL/I Sample - Function EKG_ChangeField	SEKGSMP1
EKG51402	same	PL/I Sample - Function EKG_SwapField	SEKGSMP1
EKG51403	same	PL/I Sample - Function EKG_ChangeSubfield	SEKGSMP1
EKG51404	same	PL/I Sample - Function EKG_SwapSubfield	SEKGSMP1
EKG51405	same	PL/I Sample - Function EKG_LinkTrigger	SEKGSMP1
EKG51406	same	PL/I Sample - Function EKG_LinkNoTrigger	SEKGSMP1
EKG51407	same	PL/I Sample - Function EKG_UnLinkTrigger	SEKGSMP1
EKG51408	same	PL/I Sample - Function EKG_UnLinkNoTrigger	SEKGSMP1
EKG51409	same	PL/I Sample - Function EKG_CreateObject	SEKGSMP1
EKG51410	same	PL/I Sample - Function EKG_DeleteObject	SEKGSMP1
EKG51411	same	PL/I Sample - Function EKG_RevertToInherited	SEKGSMP1
EKG51412	same	PL/I Sample - Function EKG_AddNotifySubscription	SEKGSMP1
EKG51413	same	PL/I Sample - Function EKG_DeleteNotifySubscription	SEKGSMP1
EKG51415	same	PL/I Sample - Function EKG_TriggerNamedMethod	SEKGSMP1
EKG51416	same	PL/I Sample - Function EKG_TriggerOIMethod	SEKGSMP1
EKG51417	same	PL/I Sample - Function EKG_AddObjDelNotifySubs	SEKGSMP1

Table 26. List of RODM Samples (continued)

Distributed As	Name	Description	Data Set Name
EKG51418	same	PL/I Sample - Function EKG_DelObjDelNotifySubs	SEKGSMP1
EKG51419	same	PL/I Sample - Function EKG_ChangeMultipleFields	SEKGSMP1
EKG51501	same	PL/I Sample - Function EKG_QueryField	SEKGSMP1
EKG51502	same	PL/I Sample - Function EKG_QuerySubfield	SEKGSMP1
EKG51503	same	PL/I Sample - Function EKG_QueryEntityStructure	SEKGSMP1
EKG51504	same	PL/I Sample - Function EKG_QueryFieldStructure	SEKGSMP1
EKG51505	same	PL/I Sample - Function EKG_QueryFieldID	SEKGSMP1
EKG51506	same	PL/I Sample - Function EKG_QueryFieldName	SEKGSMP1
EKG51507	same	PL/I Sample - Function EKG_QueryNotifyQueue	SEKGSMP1
EKG51508	same	PL/I Sample - Function EKG_QueryMultipleSubfields	SEKGSMP1
EKG51509	same	PL/I Sample - Function EKG_Locate	SEKGSMP1
EKG51510	same	PL/I Sample - Function EKG_QueryResponseBlockOverflow	SEKGSMP1
EKG51600	same	PL/I Sample - Function EKG_ExecuteFunctionList	SEKGSMP1
EKG52001	same	PL/I Sample - Function EKG_QueryFunctionBlockContents	SEKGSMP1
EKG52002	same	PL/I Sample - Function EKG_LockObjectList	SEKGSMP1
EKG52003	same	PL/I Sample - Function EKG_UnlockAll	SEKGSMP1
EKG52004	same	PL/I Sample - Function EKG_ResponseBlock	SEKGSMP1
EKG52005	same	PL/I Sample - Function EKG_SendNotification	SEKGSMP1
EKG52006	same	PL/I Sample - Function EKG_SetReturnCode	SEKGSMP1
EKG52007	same	PL/I Sample - Function EKG_WhereAmI	SEKGSMP1
EKG52008	same	PL/I Sample - Function EKG_OutputToLog	SEKGSMP1
EKG52009	same	PL/I Sample - Function EKG_MessageTriggeredAction	SEKGSMP1
EKG52011	same	PL/I Sample - Function EKG_QueryObjectName	SEKGSMP1

Table 26. List of RODM Samples (continued)

Distributed As	Name	Description	Data Set Name
EKG5VDCL	same	PL/I sample that declares RODM data types	SEKGSMP1
EKG5WAIT	same	PL/I sample to invoke RODM EKGWAIT	SEKGSMP1
EKG61101	same	C Sample - Function EKG_Connect	SEKGSMP1
EKG61102	same	C Sample - Function EKG_Disconnect	SEKGSMP1
EKG61201	same	C Sample - Function EKG_Checkpoint	SEKGSMP1
EKG61202	same	C Sample - Function EKG_Stop	SEKGSMP1
EKG61302	same	C Sample - Function EKG_CreateClass	SEKGSMP1
EKG61303	same	C Sample - Function EKG_DeleteClass	SEKGSMP1
EKG61304	same	C Sample - Function EKG_CreateField	SEKGSMP1
EKG61305	same	C Sample - Function EKG_DeleteField	SEKGSMP1
EKG61306	same	C Sample - Function EKG_CreateSubfield	SEKGSMP1
EKG61307	same	C Sample - Function EKG_DeleteSubfield	SEKGSMP1
EKG61401	same	C Sample - Function EKG_ChangeField	SEKGSMP1
EKG61402	same	C Sample - Function EKG_SwapField	SEKGSMP1
EKG61403	same	C Sample - Function EKG_ChangeSubfield	SEKGSMP1
EKG61404	same	C Sample - Function EKG_SwapSubfield	SEKGSMP1
EKG61405	same	C Sample - Function EKG_LinkTrigger	SEKGSMP1
EKG61406	same	C Sample - Function EKG_LinkNoTrigger	SEKGSMP1
EKG61407	same	C Sample - Function EKG_UnLinkTrigger	SEKGSMP1
EKG61408	same	C Sample - Function EKG_UnLinkNoTrigger	SEKGSMP1
EKG61409	same	C Sample - Function EKG_CreateObject	SEKGSMP1
EKG61410	same	C Sample - Function EKG_DeleteObject	SEKGSMP1
EKG61411	same	C Sample - Function EKG_RevertToInherited	SEKGSMP1
EKG61412	same	C Sample - Function EKG_AddNotifySubscription	SEKGSMP1

Table 26. List of RODM Samples (continued)

Distributed As	Name	Description	Data Set Name
EKG61413	same	C Sample - Function EKG_DeleteNotifySubscription	SEKGSMP1
EKG61415	same	C Sample - Function EKG_TriggerNamedMethod	SEKGSMP1
EKG61416	same	C Sample - Function EKG_TriggerOIMethod	SEKGSMP1
EKG61417	same	C Sample - Function EKG_AddObjDelNotifySubs	SEKGSMP1
EKG61418	same	C Sample - Function EKG_DelObjDelNotifySubs	SEKGSMP1
EKG61419	same	C Sample - Function EKG_ChangeMultipleFields	SEKGSMP1
EKG61501	same	C Sample - Function EKG_QueryField	SEKGSMP1
EKG61502	same	C Sample - Function EKG_QuerySubfield	SEKGSMP1
EKG61503	same	C Sample - Function EKG_QueryEntityStructure	SEKGSMP1
EKG61504	same	C Sample - Function EKG_QueryFieldStructure	SEKGSMP1
EKG61505	same	C Sample - Function EKG_QueryFieldID	SEKGSMP1
EKG61506	same	C Sample - Function EKG_QueryFieldName	SEKGSMP1
EKG61507	same	C Sample - Function EKG_QueryNotifyQueue	SEKGSMP1
EKG61508	same	C Sample - Function EKG_QueryMultipleSubfields	SEKGSMP1
EKG61509	same	C Sample - Function EKG_Locate	SEKGSMP1
EKG61510	same	C Sample - Function EKG_QueryResponseBlockOverflow	SEKGSMP1
EKG61600	same	C Sample - Function EKG_ExecuteFunctionList	SEKGSMP1
EKG62001	same	C Sample - Function EKG_QueryFunctionBlockContents	SEKGSMP1
EKG62002	same	C Sample - Function EKG_LockObjectList	SEKGSMP1
EKG62003	same	C Sample - Function EKG_UnlockAll	SEKGSMP1
EKG62004	same	C Sample - Function EKG_ResponseBlock	SEKGSMP1
EKG62005	same	C Sample - Function EKG_SendNotification	SEKGSMP1
EKG62006	same	C Sample - Function EKG_SetReturnCode	SEKGSMP1

Table 26. List of RODM Samples (continued)

Distributed As	Name	Description	Data Set Name
EKG62007	same	C Sample - Function EKG_WhereAmI	SEKGSMP1
EKG62008	same	C Sample - Function EKG_OutputToLog	SEKGSMP1
EKG62009	same	C Sample - Function EKG_MessageTriggeredAction	SEKGSMP1
EKG62011	same	C Sample - Function EKG_QueryObjectName	SEKGSMP1
EKG6VDCL	same	C sample that declares RODM data types	SEKGSMP1
EKG6WAIT	same	C sample to invoke RODM EKGWAIT	SEKGSMP1
EKGCMIMV	same	C sample that increments the value of a specified field	SEKGSMP1
EKGCPPI	same	Sample change method to invoke EKGSPPI	SEKGSMP1
EKGCTABL	same	Sample loader control file	SEKGLUTB
EKGCTIM	same	Sample change method source	SEKGSMP1
EKGCUST	same	Sample RODM customization file	SEKGSMP1
EKGCVER	same	C sample verification routine source	SEKGSMP1
EKGIN1	same	Sample RODM structure load input	SEKGSMP1
EKGIN3	same	Sample RODM object load input	SEKGSMP1
EKGINIT	same	Sample initialization method	SEKGSMP1
EKGINMTB	same	Sample loader install method table	SEKGCAS1
EKGIVER	same	PL/I sample verification routine source	SEKGSMP1
EKGKUCDS	same	Sample that allocates output data sets used by RODMUNLD to store RODM loader statements	SEKGSMP1
EKGKUJCL	same	Sample that invokes the RODM unload function	SEKGSMP1
EKGLG000	same	Sample log formatter procedure	SEKGSMP1
EKGLLINK	same	Sample JCL to link edit user modules that load structure and object definitions to RODM	SEKGSMP1
EKGLLOAD	same	Sample RODM JCL procedure to load RODM data cache	SEKGSMP1
EKGLOADP	same	Sample RODM JCL procedure to load RODM data cache	PROCLIB
EKGLUSER	same	Sample JCL for structure or object load of RODM through a user module	SEKGSMP1
EKGMIMV	same	Sample named method source	SEKGSMP1
EKGNEQL	same	Sample notification method 4	SEKGSMP1
EKGNLST	same	Sample notification method 3	SEKGSMP1
EKGNOTF	same	Sample notification method 1	SEKGSMP1
EKGNTHD	same	Sample notification method 2	SEKGSMP1

Table 26. List of RODM Samples (continued)

Distributed As	Name	Description	Data Set Name
EKGOPPI	same	Sample object independent method to invoke EKGSPPI	SEKGSMP1
EKGPTENU	same	Sample load parameter mapping table	SEKGLUTB
EKGRDUMP	same	Sample JCL to run the RODM dump utility	SEKGSMP1
EKGRLOG	same	Sample JCL to submit log formatter procedure	SEKGSMP1
EKGSCKPT	same	Sample JCL to allocate VSAM checkpoint data sets for a second instance of RODM	SEKGSMP1
EKGSI101	same	Allocates the RODM log VSAM data bases	SEKGSMP1
EKGSI102	same	Allocates the RODM log VSAM data bases for a second instance of RODM	SEKGSMP1
EKGSI201	same	Allocates the RODM master, transaction, and check point VSAM data bases	SEKGSMP1
EKGSI202	same	Allocates the RODM master, transaction, and check point VSAM data bases for a second instance of RODM	SEKGSMP1
EKGSID01	same	Deletes RODM VSAM databases	SEKGSMP1
EKGSID02	same	Deletes the RODM log VSAM data bases for a second instance of RODM	SEKGSMP1
EKGSID03	same	Deletes the RODM master, transaction, and check point VSAM data bases for a second instance of RODM	SEKGSMP1
EKGSJ004	same	Allocates logs and VSAM databases for RODM	SEKGSMP1
EKGSLOG	same	Sample JCL to allocate VSAM data sets used for logging for a second instance of RODM	SEKGSMP1
EKGSVER	same	Sample JCL to run the installation verification routine	SEKGSMP1
EKGXRODM	same	Sample JCL to start RODM	PROCLIB

FLCARODM Samples

The following samples are specific to FLCARODM which provides an external REXX interface for the manipulation of RODM objects.

Table 27. List of FLCARODM Samples

Distributed As	Name	Description	Data Set Name
FLCSBX1	same	Sample REXX example for FLCARODM	CNMSAMP
FLCSBX2	same	Sample REXX example for FLCARODM	CNMSAMP
FLCSBX3	same	Sample REXX example for FLCARODM	CNMSAMP
FLCSBX4	same	Sample REXX example for FLCARODM	CNMSAMP
FLCSBX5	same	Sample REXX example for FLCARODM	CNMSAMP
FLCSBX6	same	Sample REXX example for FLCARODM	CNMSAMP
FLCSBX7	same	Sample REXX example for FLCARODM	CNMSAMP

Table 27. List of FLCARODM Samples (continued)

Distributed As	Name	Description	Data Set Name
FLCSBX8	same	Sample REXX example for FLCARODM	CNMSAMP
FLCSDX1	same	Sample REXX example for FLCARODM	CNMSAMP
FLCSDX2	same	Sample REXX example for FLCARODM	CNMSAMP
FLCSDX3	same	Sample REXX example for FLCARODM	CNMSAMP
FLCSPX1	same	Sample REXX example for FLCARODM	CNMSAMP
FLCSPX2	same	Sample REXX example for FLCARODM	CNMSAMP
FLCSQX1	same	Sample REXX example for FLCARODM	CNMSAMP
FLCSQX2	same	Sample REXX example for FLCARODM	CNMSAMP
FLCSQX3	same	Sample REXX example for FLCARODM	CNMSAMP
FLCSQX4	same	Sample REXX example for FLCARODM	CNMSAMP
FLCSQX5	same	Sample REXX example for FLCARODM	CNMSAMP
FLCSSTEM	same	Sample REXX example for FLCARODM	CNMSAMP
FLCSX10	same	Sample REXX example for FLCARODM	CNMSAMP
FLCSX11	same	Sample REXX example for FLCARODM	CNMSAMP
FLCSX12	same	Sample REXX example for FLCARODM	CNMSAMP
FLCSX13	same	Sample REXX example for FLCARODM	CNMSAMP
FLCSX14	same	Sample REXX example for FLCARODM	CNMSAMP
FLCSX15	same	Sample REXX example for FLCARODM	CNMSAMP
FLCSX16	same	Sample REXX example for FLCARODM	CNMSAMP
FLCSX17	same	Sample REXX example for FLCARODM	CNMSAMP
FLCSX18	same	Sample REXX example for FLCARODM	CNMSAMP
FLCSX19	same	Sample REXX example for FLCARODM	CNMSAMP
FLCSX1	same	Sample REXX example for FLCARODM	CNMSAMP
FLCSX20	same	Sample REXX example for FLCARODM	CNMSAMP
FLCSX21	same	Sample REXX example for FLCARODM	CNMSAMP
FLCSX22	same	Sample REXX example for FLCARODM	CNMSAMP
FLCSX2	same	Sample REXX example for FLCARODM	CNMSAMP
FLCSX3	same	Sample REXX example for FLCARODM	CNMSAMP
FLCSX4	same	Sample REXX example for FLCARODM	CNMSAMP
FLCSX5	same	Sample REXX example for FLCARODM	CNMSAMP
FLCSX6	same	Sample REXX example for FLCARODM	CNMSAMP
FLCSX7	same	Sample REXX example for FLCARODM	CNMSAMP
FLCSX8	same	Sample REXX example for FLCARODM	CNMSAMP
FLCSX9	same	Sample REXX example for FLCARODM	CNMSAMP
FLCSXF1	same	Sample REXX example for FLCARODM	CNMSAMP
FLCSXL01	same	Sample REXX example for FLCARODM	CNMSAMP
FLCSXL02	same	Sample REXX example for FLCARODM	CNMSAMP
FLCSXQ1	same	Sample REXX example for FLCARODM	CNMSAMP
FLCSXQ2	same	Sample REXX example for FLCARODM	CNMSAMP

Table 27. List of FLCARODM Samples (continued)

Distributed As	Name	Description	Data Set Name
FLCSXQ3	same	Sample REXX example for FLCARODM	CNMSAMP
FLCSXS01	same	Sample REXX example for FLCARODM	CNMSAMP
FLCSXS02	same	Sample REXX example for FLCARODM	CNMSAMP
FLCSXS03	same	Sample REXX example for FLCARODM	CNMSAMP
FLCSXS04	same	Sample REXX example for FLCARODM	CNMSAMP
FLCSXS05	same	Sample REXX example for FLCARODM	CNMSAMP
FLCSXS06	same	Sample REXX example for FLCARODM	CNMSAMP
FLCSXS07	same	Sample REXX example for FLCARODM	CNMSAMP
FLCSXS08	same	Sample REXX example for FLCARODM	CNMSAMP

SNA Topology Manager and APPN Accounting Manager Samples

Table 28 lists the samples and includes a brief description of each SNA Topology Manager and APPN Accounting Manager sample and the data set name where each member resides once installation is complete.

Table 28. List of SNA Topology Manager and APPN Accounting Manager Samples

Distributed As	Name	Description	Data Set Name
CNMSTYLE	same	This sample defines many of the NetView initialization parameters. Refer to CNMSTYLE for more information. If you make changes to this sample while the NetView program is running, recycle the NetView program to merge these changes.	DSIPARM
CNMSXENT	same	Sample used to show that the SNA topology manager and APPN accounting manager are installed	CNMSAMP
DSICMENT	same	NetView command model statements for the APPN accounting manager and the SNA topology manager.	DSIPARM
FLBAUT	same	Automation Table Entries for SNA topology and APPN accounting managers	DSIPARM
FLBEXV	same	SNA topology manager Exception View Table	DSIPARM
FLBEXVU	same	SNA topology manager Exception View Table user entry. This is a sample INCLUDE file that is included from SNA topology manager initialization file FLBEXV.	DSIPARM
FLBGMMPR	same	NetView operator profile definition for APPN accounting manager	DSIPRF
FLBOSIDS	same	SNA topology manager OSI/Display Status Table. Maps the OSI status values to DisplayStatus values.	DSIPARM

Table 28. List of SNA Topology Manager and APPN Accounting Manager Samples (continued)

Distributed As	Name	Description	Data Set Name
FLBOSIDU	same	SNA topology manager OSI/Display Status Table user entry. This is a sample INCLUDE file that is included from SNA topology manager initialization file FLBOSIDS.	DSIPARM
FLBS4210	same	Sample PL/I Link Re-Activate Command Processor	CNMSAMP
FLBS4211	same	Sample PL/I Link Re-Activate Method	CNMSAMP
FLBS8001	same	Sample REXX exec named REFRESHC. This exec executes a TOPOSNA CRITICAL command against a list of resources specified in a member of the DSIOPEN dataset.	CNMSAMP
FLBS8002	same	Sample used by the FLBS8001 sample to provide a table of LUs for the REFRESHC command.	CNMSAMP
FLBSRT	same	SNA topology manager Status Resolution Table. This table contains the hierarchy of OSI status entries used to resolve the status of a multiply-owned resource.	DSIPARM
FLBSRTU	same	SNA topology manager Status Resolution Table user entry. This is a sample INCLUDE file that is included from SNA topology manager initialization file FLBSRT.	DSIPARM
FLBSYSDA	same	APPN accounting manager initialization file	DSIPARM
FLBSYSD	same	SNA topology manager initialization file	DSIPARM
FLBTPROF	same	Sample operator profile statements for NetView SNA topology manager Function	DSIPRF
FLBTRDM1	same	SNA Topology Data Model loader file that creates the generic managed object classes	CNMSAMP
FLBTRDM2	same	SNA Topology Data Model loader file that creates the APPN and Subarea managed object classes	CNMSAMP
FLBTRDM3	same	SNA Topology Data Model loader file that creates a SNA topology manager specific class for operational defaults	CNMSAMP
FLBTRDM4	same	SNA Topology Data Model loader file that creates RODM method objects	CNMSAMP
FLBTRDM5	same	SNA Topology Data Model loader file that sets default values for the generic managed object classes	CNMSAMP
FLBTRDM6	same	SNA Topology Data Model loader file that sets default values for the APPN and Subarea object classes	CNMSAMP
FLBTRDM7	same	SNA Topology Data Model loader file that sets GMFHS-navigation values	CNMSAMP
FLBTRDM8	same	SNA Topology Data Model loader file that creates View_Information_Reference_Class objects	CNMSAMP

Table 28. List of SNA Topology Manager and APPN Accounting Manager Samples (continued)

Distributed As	Name	Description	Data Set Name
FLBTRDM9	same	SNA Topology Data Model loader file that is reserved for Tivoli use	CNMSAMP
FLBTRDMA	same	SNA Topology Data Model loader file for Predefined Network_View_Class view object	CNMSAMP
FLBTRDMB	same	SNA Topology Data Model loader file for View Information Object Class Instances	CNMSAMP
FLBTRDMC	same	SNA Topology Data Model loader that sets links between VIROs and DRTs	CNMSAMP
FLBTRDMD	same	SNA Topology Data Model loader file that creates Exception Views	CNMSAMP
FLBTRDME	same	View Notification Granularity loader file that installs notification methods	CNMSAMP
FLBTRDMG	same	SNA Topology Data Model loader file that takes the netid from DisplayResourceName	CNMSAMP
FLBTRDMH	same	SNA Topology Data Model loader file that creates a SNA_Backbone_View	CNMSAMP
FLBTRDMI	same	SNA Topology Data Model loader file that produces a grid layout for More Detail views	CNMSAMP
FLBTRDMJ	same	SNA Topology Data Model loader file that creates port aggregate objects and suppresses logical links	CNMSAMP
FLBTRDMZ	same	SNA Topology Data Model loader file that creates the Topology_Manager class	CNMSAMP
FLBTREU	same	A method that is a sample exception view customization method that can be invoked from the SNA topology manager method exception view table is refreshed.	CNMSAMP
FLBTRSC1	same	SNATM APPN Sample Network JCL for the initial scene	CNMSAMP
FLBTRSC2	same	SNATM APPN Sample Network JCL for scene 2	CNMSAMP
FLBTRSC3	same	SNATM APPN Sample Network JCL for scene 3	CNMSAMP
FLBTRSC4	same	SNATM APPN Sample Network JCL for scene 4	CNMSAMP
FLBTRSC5	same	SNATM APPN Sample Network JCL for scene 5	CNMSAMP
FLBTRSC6	same	SNATM APPN Sample Network JCL for scene 6	CNMSAMP
FLBTRSN1	same	SNATM APPN Sample Network loader file for the initial scene	CNMSAMP
FLBTRSN2	same	SNATM APPN Sample Network loader file for scene 2	CNMSAMP
FLBTRSN3	same	SNATM APPN Sample Network loader file for scene 3	CNMSAMP

Table 28. List of SNA Topology Manager and APPN Accounting Manager Samples (continued)

Distributed As	Name	Description	Data Set Name
FLBTRSN4	same	SNATM APPN Sample Network loader file for scene 4	CNMSAMP
FLBTRSN5	same	SNATM APPN Sample Network loader file for scene 5	CNMSAMP
FLBTRSN6	same	SNATM APPN Sample Network loader file for scene 6	CNMSAMP
FLBTRUM	same	A method that is a sample status customization method that can be invoked from the SNA topology manager method FLBTRST when the manager is processing a status update for an object in RODM	CNMSAMP

Index

D

E

F

G

H

I

J

L

M

N

O

P

R

S

T

U

V

W

X

File Number: S370/4300/30XX-50
Program Number: 5697-B82 (MVS/ESA)

Printed in the United States of America
on recycled paper containing 10%
recovered post-consumer fiber.

SC31-8770-00